TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Haris Ahmed IVEM184575

Low Power IoT Sensor System for Underground Drainage Monitoring

Master's Thesis

Supervisor: Alar Kuusik

Ph.D.

Tallinn 2021

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Haris Ahmed IVEM184575

Energiasäästlik IoT sensorsüsteem maa-aluse drenaaži monitoorimiseks
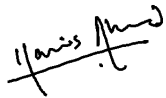
Magistritöö

Juhendaja: Alar Kuusik

Ph.D.

Tallinn 2021

Author's declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Haris Ahmed

04.01.2021

Abstract

Real time monitoring and control of Urban Drainage Systems is considered as the most feasible solution for preventing urban flooding events caused by increased rainfall due to climate change. This work focuses on the design and development of a low powered Internet of Things (IoT) based sensor system which can monitor water levels inside an underground drainage system in real time. This thesis presents literature overview of different water level sensing systems and Low Powered Wide Area Networks (LPWAN), feasibility analysis of two LPWAN technologies i.e. LoRa and NB-IoT for manhole communication, and comparison of different NB-IoT cellular development platforms. Moreover, hardware setup, software development and optimizations, and cloud infrastructure implementation details for development of a prototype are discussed, and the results of the sensing capabilities and power consumption of the system are summarized. The developed prototype is capable of making water level measurements up to 1 meter with ± 2 cm accuracy while consuming 69.2 µA average current and can operate for more than 6.5 years on a 4000 mAh battery.

This thesis is written in English language and is 72 pages long, including 7 chapters, 31 figures and 12 tables.

Annotatsioon

Energiasäästlik IoT sensorsüsteem maa-aluse drenaaži monitoorimiseks

Reaalajalist sadeveedrenaaži jälgimist ja juhtimist peetakse üheks parimaks lahenduseks kliima soojenemisest tulenevate äkkvihmade põhjustatud üleujutuste vältimisel linnaruumis. Käesolev töö käsitleb energiasäästliku IoT sensorsüsteemi disaini ja arendust, mis võimaldab reaalajas jälgida veetaset maa-alustes drenaažitorustikes. Lõputöö sisaldab kirjanduse ülevaadet erinevatest veemõõdusüsteemidest ja LPWAN energiasäästlikest kaugsidevõrkudest; analüüsi kahe sidetehnoloogia – LoRa ja NB-IoT kasutatavusest kanalisatsioonikaevudes ja erinevate NB-IoT arendusplatvormide võrdlust. Prototüübi riistvaralahenduse, tarkvaraarenduse ja optimeerimise, pilvelahenduse realiseerimise detailid ning süsteemi mõõtmiste ja energiatarbe parameetrid on töös samuti kokku võetud. On näidatud, et prototüüp mõõdab kuni 1m kauguselt veetaset täpsusega ± 2 cm tarbides seejuures keskmiselt 69,2 µA toitevoolu ning töötades enam kui 6,5 aastat 4000 mAh liitiumpatareist..

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 72 leheküljel, 7 peatükki, 31 joonist, 12 tabelit.

List of abbreviations and terms

UDS	Urban Drainage System
SUDS	Smart Urban Drainage System
RTMC	Real Time Monitoring and Control
LPWAN	Low Power Wide Area Networks
LoRa	Long Range Wide Area Network
LTE	Long Term Evolution
SiP	System-in-Package
ISM	Industrial, Scientific and Medical
bps	Bits per second
CSS	Chirp Spread Spectrum
SF	Spreading Factor
LoRaWAN	Long Range Wide Area Network
QoS	Quality of Service
TDOA	Time Difference Of Arrival
3GPP	3rd Generation Partnership Program
LTE	Long-Term Evolution
GSM	Global System for Mobile Communications
OFDMA	Orthogonal Frequency Division Multiple Access
FDMA	Frequency Division Multiple Access
PSM	Power Saving Mode
eDRX	Extended Discontinuous Reception
DRX	Discontinuous Reception
VoLTE	Voice over LTE
M2M	Machine-to-machine
IP	Internet Protocol
NIDD	Non-IP Data Delivery
LED	Light Emitting Diode
LDRs	Light Dependent Resistors

OSA	Optical Spectrum Analyser
FBG	Fibre Bragg Grating
RTOS	Real Time Operating System
AT	ATtention
SINR	Signal to Interference and Noise Ratio
RSSI	Received Signal Strength Indication
NB-RSRP	Narrow Band Reference Signal Received Power
SNR	Signal to Noise Ratio
EGPRS	Enhanced General Packet Radio Service
AWS	Amazon Web Services
MQTT	Message Queuing Telemetry Transport
GPIOTE	General Purpose Input Output Tasks and Events
UART	Universal Asynchronous Receiver Transmitter
I2C	Inter-Integrated Circuit
SPI	Serial Peripheral Interface
RTC	Real-Time Clock
PPI	Programmable Peripheral Interconnect
Li-Po	Lithium Polymer
N-MOS	N-type Metal Oxide Semiconductor
SDK	Software Development Kit
IDE	Integrated Development Environment
JSON	JavaScript Object Notation
API	Application Programming Interface
BSD	Berkeley Software Distribution
HAL	Hardware Abstraction Layer
FOTA	Firmware over-the-air
HTTP	Hypertext Transfer Protocol
TLS	Transport Layer Security
CSR	Certificate Signing Request
CA	Certification Authority
RSA	Rivest-Shamir-Adleman

Table of contents

List of figures

List of tables

1 Introduction

Urban Drainage Systems (UDSs) are considered to be the stepping stone of modern civilization as they have allowed for spatially dense cities while ensuring safe and hygienic living conditions for us. But these systems are prone to failure in extreme events when their design criteria are exceeded [1]. These extreme events can be triggered by different socio-economic and climatic factors. Socio-economic factors include ever increasing population density in urban areas resulting from demographic changes and immigration. Whereas, slow and steady rise in sea levels and increase of extreme rainfall events constitute the climatic factors.

By making use of different projection models for the analysis of meteorological data gathered during the past years, increase in extreme rainfall events is predicted across all of Europe. This is especially true for Northern Europe as the frequency of these extreme rainfall events is projected to increase more than thrice by 2030 [2]. Higher frequency of these events can result in flooding of both urban and rural areas. Various studies have been conducted which focus on taking measures which can minimize the effects of these flooding events. But most of these measures focus on post-event correction rather than pre-event prevention. Of the pre-event preventive measures, one of the most promising technique which can significantly reduce the risk of urban flooding is the reduction of peak water flows through UDSs during extreme rainfall events.

Typical UDS's design criteria makes them operate on approximately 2/3 water level elevation rate in the channel because of it being hydraulically most feasible. Additionally, free space in the pipes allow for the ventilation of air in the drainage system which prevents trapping and building of air pressure in the channels. But if the water level elevation rate increases from the aforementioned design criteria, then it will result in overflow as the water starts flowing in the reverse direction i.e. from manholes towards the ground level. This becomes the source of urban flooding due to extreme rainfall events, which can pose a threat to a city's infrastructure and can disturb the lives of its inhabitants. This situation can get even worse in case of the combined sewer

systems where the stormwater and wastewater are discharged together. It can result in extreme health hazard as a result of resurfacing of untreated wastewater which can come in contact with other water bodies. [3]

Tackling this problem by formulating UDS as a control system i.e. Smart Urban Drainage System (SUDS) with Real-Time Monitoring and Control (RTMC) is seen as the most feasible solution [4]. These SUDS employ different actuators such as gates, inlets, weir walls and curbs which can be used to control the upward flow of water in the system. Installing these actuators is relatively simple and thus require no major structural overhaul. Thus, water levels can be decreased before extreme rainfall predictions which significantly lowers the risk of flooding. These systems have found to be very effective for reducing the peak water level up to 50% [5].

1.1 Problem Statement

As discussed in the introduction, the goal is to minimize the risk of urban flooding events during heavy rainfalls by controlling peak flows through the UDS by adding RTMC mechanisms. For setting up such RTMC mechanisms, two major design considerations are:

1. Comparison and Selection of an optimal wireless connectivity solution, which can provide reliable and energy efficient data transfer from the underground manholes.

2. Development and Implementation of energy efficient and secure data acquisition system capable of sending sensor data to remote server.

Until recently, realization of SUDS had not been possible because of the limitations of communication technologies for establishing reliable link between the sensors and the cloud. What makes it difficult is that for SUDS application, sensors need to be placed underground and must be able to communicate to at least tens of hundreds of meters while operating at batteries. This poses a great challenge for establishing reliable communication as electromagnetic waves undergo massive attenuation trying to penetrate the soil. Also, non-line of sight connection can hamper the quality of the connection drastically.

With the recent developments in Low Powered Wide Area Network (LPWAN) technologies, technologies like Sigfox, LoRa, cellular NB-IoT, cellular CAT-M1, and NB-Fi have made the implementation of SUDS concept feasible. All of these technologies share similar characteristics i.e. narrow-band, long range, noise insensitive, and energy efficient, and they offer some distinct features which make them stand out for different use cases.

The main objectives of the thesis are summarized below:

1. Selection of a sensor which can be used to measure water-level in real time.

2. Comparison and selection of relevant wireless communication technologies which can be used for a feasible SUDS implementation.

3. Comparative testing of selected candidate LPWAN candidate technologies for SUDS application on the basis of link quality measurements, energy consumption, and cost.

4. Comparison of potential embedded hardware platforms for the selected LPWAN technology on the basis of functionality, pricing, and power efficiency.

5. Development of a manhole sensor prototype and its optimization for low powered usage.

6. Implementation of a secure backend service prototype to collect the measurement results of the sensor device.

7. Analysis and testing of the developed system prototype.

The developed prototype will be deployed and tested as a part of "PRG667: Decentralized real-time management of stormwater systems in climate-proof smart cities" research project funded by the Estonian Research Council.

1.2 Thesis Structure

The thesis work has been organized in 7 different sections:

Section 1 lists introduction and description of the problem which this work aims to solve. Section 2 includes the background information and literature review of different water level sensing and LPWAN technologies followed by the comparison and selection of the most suitable ones based on the application requirements. Section 3 presents the results of feasibility testing of LoRa and NB-IoT for underground communication in manholes. Section 4 introduces different NB-IoT cellular platforms and presents their comparative evaluation. Section 5 covers details of hardware integration, software development and optimisations, and cloud server setup and configuration, necessary for implementing working prototype of the system. Section 6 summarizes results of the sensor testing and power consumption analysis conducted on the developed prototype. Section 7 concludes the thesis work by highlighting key areas which require more work to improve the performance of the developed solution.

2 Background and Literature Review

This chapter will present background information and literature overview which are relevant to solving key challenges of this work. First and foremost, all the important system design and operational considerations are established which are important for this solution. Then an existing manhole monitoring system will be referenced along with its limitations. After that, a brief overview and comparison of different water level sensors will be presented and the communication requirements for the chosen sensor will be evaluated. Finally, the theoretical aspects of relevant LPWAN technologies and their suitability for low powered monitoring of UDS will be discussed.

2.1 System Design and Operational Specifications

Before moving forward with comparison and evaluation of different technologies, it is important to establish some design and operational specifications for this application. Such specifications will provide with a frame of reference to compare these technologies and to reach towards meaningful conclusions about them from an objective point of view. These design considerations boil down to customer requirements and can vary from one UDS system to another. For this work, the monitoring system's design and operational characteristics were agreed upon by collaborating with the Tallinn University of Technology's Department of Civil Engineering and Architecture.

A better suited water level measurement sensor for this use case will be the one which operates on non-contact based principle. This is essential for longer lifetime of the sensing system as the sensors will operate passively and their performance will not get degraded being in contact with contaminated water. It should be noise insensitive and needs to remain largely unaffected from variable environmental conditions. The sensor should provide continuous water level measurements preferably up to 2 m with the measurement accuracy within ± 1 cm and should be energy efficient.

Suitable communication technology's maximum payload size and throughput should be sufficient to send sensor values from end device to the cloud server. The communication

protocol should support extreme power saving features to extend battery life cycle of end device to three years or more. Communication solution should provide some level of security. Additionally, bidirectional communication can be useful in future to establish control over different actuators to minimise water elevation rate.

The frequency of sensor data acquisition and transmission is dependent on the water level variation and boundary conditions of the entire UDS [4]. So, this frequency can change from one system to another. Since during normal weather conditions, water level variation in manholes is usually not that extreme so for this work the operational frequency of one measurement per hour has been chosen. This frequency can be increased during extreme rain fall events or when there is an increased risk of flooding. Such operational cycle can allow for better power optimisations without adversely impacting the effectiveness of the system.

2.2 Overview of an existing solution

Some existing solutions have been developed and tested for water level monitoring in manholes. The most prominent one is Sensoneo's Remote Water Level Monitoring System [6] which is described here along with its limitations.

This remote water level monitoring system is based on Sensoneo's Octopus sensor and is capable of depth measurements of up to 25 meters. The Octopus sensor relies on contact-based water level indicators and can only detect up to three water level thresholds. Thus, the solution is not suitable for real time water level measurements and can only notify when the water reaches certain predefined levels. The measurements are event triggered and only one measurement is taken per day if no event gets triggered. This system relies on GSM and EGPRS as the wireless communication solution. In terms of power, the system makes use of AA batteries and the battery is rated to last for several years.

2.3 Overview of water level measurement methods

Here, a brief overview of different water level measurement methods along with their operational characteristics will be summarised. The advantages and limitations of these

different methods will be highlighted as well. After comparing the most suitable methodologies, one will be selected for low powered sensing in this application.

2.3.1 Float level sensors

Float level sensors rely on a buoyant object floating on the surface of the liquid for measuring water level. The floating object is usually connected through cable with mechanical objects such as pulleys, gears which sink to the bottom. As the float level rises and falls, the pulleys and gear turn, and the water level is estimated using different electrical or mechanical approaches.

The problem with float level sensors is that it is very difficult to translate the change in the level of floating object to accurate level measurement [7]. These are usually employed for discrete level measurements defined by certain thresholds. Another major drawback is that the floating mechanism could easily become entangled which can restrict the free movement of the object thus producing erroneous results. This is especially true if the mechanism is employed in narrow wells and manholes [8].

2.3.2 Capacitive sensors

Capacitive sensors usually consist of two parallel electrodes with insulating material in between them. While designing these types of sensors, variation of the capacitor's shape and material can be exploited to fine tune their stability, resolution and price [9].

The operating principle of these sensors is not that different from a standard capacitive component. Water level measurements are obtained by measuring the capacitance between the two electrodes. As capacitance is proportional to the dielectric constant, a change in water level will result in changing of the effective dielectric constant and the capacitance between the electrodes. Hence, water level rising or falling between the electrodes will increase or decrease the capacitance of the circuit as a function of water level height. [8]

A water level measurement sensor using capacitive principle was designed and tested in [10]. The designed sensor was used to measure water level in a container whose maximum depth was 30 cm. The capacitance of the sensor was shown to be proportional to the level of water inside the container.

Wide availability of different type of materials for capacitive sensor's construction makes them extremely flexible for different fluid level measurement applications [9]. Typically, capacitive sensors provide measurements in range of 10 cm – 100 cm with 0.1 mm - 1 mm accuracy and their power consumption mostly ranges between 4 – 20 mA [11]. Capacitive sensors are not practically viable where deep liquid channel needs to be monitored. This is because the length of the electrodes must be longer than the maximum attainable height of the liquid [8]. Another downside of capacitive type sensing mechanism is that it can be easily affected by parasitic capacitances originating from the corrosion and degradation of electrodes [9].

2.3.3 Ultrasonic sensors

Ultrasonic sensors conduct water level measurements without being in direct contact with the water body. They rely on the "time of flight" principle and speed of sound for calculating the distance between the transducer and the surface of water.
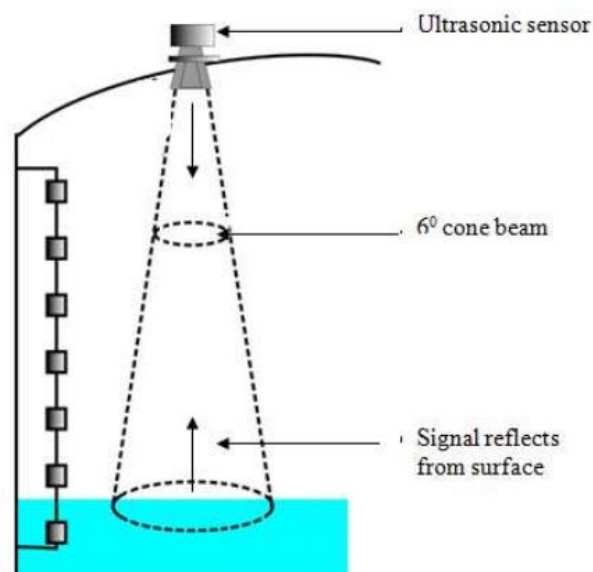
Figure 1. Working of ultrasonic sensor for water level measurement [8]

Ultrasonic sensors can emit high frequency pulses ranging from 20 kHz to 200 kHz. These pulses travel towards the water surface and are reflected back to the sensor which can now act as a receiver as shown in figure 1. The time duration between the emitted pulses and the reflected pulses can be used to calculate water level by using equation (1). Ultrasonic pulses with higher frequencies can be used for short level measurements, while lower frequency ones can be used for long level measurements.

$$\text{Distance} = \frac{v}{2t} \tag{1}$$

where 'v' is the speed of sound in the air, and 't' is the time delay between the transmitted and reflected pulse. [8]

A prototype of a water level measurement system was developed and tested using an ultrasonic ping sensor in [12]. Testing results showed that water levels were differentiable with 1 cm resolution and the system can be upscaled to create a river flow management system. Another model based on ultrasonic sensors was developed and tested for estimation of multiple levels of water in a tank in [13]. This model employed a wavelet transformation technique for signal filtering and detection in noisy environments. HC-SR04M ultrasonic sensor has been used in [14] to develop an Artificial Neural Network based water level measurement system in range of 2 cm – 5 m within ± 1 cm measurement accuracy.

Ultrasonic sensors are advantageous because of their non-contact sensing mechanism as their performance does not degrade over time. Different ultrasonic sensor can have different measurement range, accuracy and power consumption. Commonly available, AJ-SR04M have a range of 6 m with distance accuracy of 1 cm, whereas its current consumption is rated at 8 mA [15]. Recently released, TDK's chirp CH-201 is an ultra-low power sensor and it can measure distance in range of 2 cm – 5 m while consuming only 13.5 µA current for one sample [16]. Their biggest limitation is that measurements from these sensors get affected by temperature variations as the speed of sound changes, so they cannot be used in places where extreme temperature variations are anticipated [8]. Another limitation of ultrasonic sensors is that small obstructions in line of sight of sensor can result in erroneous measurements.

2.3.4 Microwave sensors

Microwave sensors share similar working principles as ultrasonic sensors with the only difference being the frequencies of the emitted pulses. Microwave sensor measurements are based on the "time of flight" principle as well but it emits electromagnetic waves with speeds much higher than that of sound.

In [17], a millimetre wave sensor has been developed and used for water level surface and sub-surface sensing. This sensor has also been used for continuously measuring water level in a tank with less than .1 cm error.

Variety of microwave sensors are available in the market with different performance characteristics and power ratings. Typically, these sensors can have very high current consumption while taking measurements. E.g. SEN0192 have measuring range of 2 m - 16 m with measuring accuracy of 3 cm and its maximum current consumption can be up to 60 mA [18]. Water level measurements with microwave sensors involves much more sophisticated circuits as compared to ultrasonic sensors. So, the cost of the system becomes considerably higher.

2.3.5 Laser sensors

Like ultrasonic and microwave sensors, laser sensors operate on "time of flight" principle, but here the speed of light is used to do the measurements. This can be advantageous in some scenarios and disadvantageous in others.

Laser sensors work by transmitting short pulse of light down to the water surface and receiving the reflected light. A special timing circuit is used to measure the time of flight and distance is calculated from equation (1) by replacing speed of sound with the speed of light. A laser sensor setup for measuring liquid level in a tank is shown in figure 2.
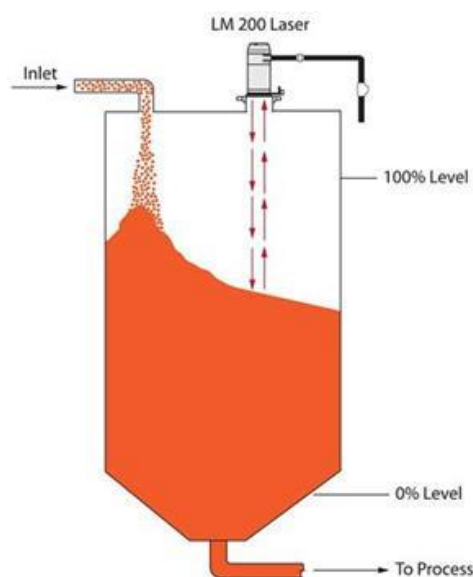
Figure 2. Laser sensor setup for liquid level measurements [7]

23

The biggest advantage of laser over ultrasonic and microwave sensors is that laser beam have minimal beam spread (0.2 degree beam divergence) and they can be directed through very small space [7]. They can also find their application in very high-temperature and high-pressure applications in conjunction with a glass window which can isolate the sensor from the extreme conditions while still allowing for reasonable measurements. Commercially, different laser sensors are available with different ratings, but lasers have typically very high operating currents. Laser sensor in [19] can measure distance in range of 40 cm - 40 m with accuracy of 2 mm, and the operating current during measurements can be as high as 180 mA.

2.3.6 Comparison of water level sensors

After considering different water level sensors which have been utilised in the literature for different applications, following conclusions can be made:

Capacitive sensors are not ideal for UDS monitoring because of their contact based operating principle. Although, they provide good accuracy and resolution, their performance will deteriorate by being in contact with drainage water over long period of time. Similarly, float level sensors are not suitable because they can only provide discrete measurements for certain thresholds. Also, there is high chance of their failure in narrow manholes because of the entanglement of the measuring mechanism.

Ultrasonic, microwave, and laser sensors are all contactless sensors and look promising for UDS monitoring. The performance characteristics of previously referenced sensors from each category are summarised in table 1.

Table 1. Comparison of Ultrasonic, Microwave, and Laser sensors

Sensors	Range	Accuracy	Current Rating
Ultrasonic Ping	2 cm – 6 m	1 cm	8 mA
Ultrasonic Chirp	20 cm – 5 m	5 mm	13.5 µA
Microwave	200 cm – 16 m	3 cm	60 mA
Laser	4 cm – 40 m	2 mm	180 mA

From the comparison in table 1, laser sensor provides the maximum accuracy and range, but it also has the highest current consumption. This makes it not so suitable for low powered application. The microwave sensor has far worse accuracy as compared to laser but consumes significantly less current. This makes it a better fit over the laser sensor but not ideal for low powered usage. Both ultrasonic sensors have low current consumptions i.e. < 10 mA and they are capable of making measurements up to 4 – 5 m with ± 1 cm accuracies which is sufficient for this application. For this work, ultrasonic ping sensor (AJ-SR04M) has been chosen for water level sensing as it has already been tested for water level measurements in 2 cm – 5 m range with ± 1 cm measurement accuracy [14]. Although, the new ultrasonic chirp sensor is more power efficient, but it is not considered here because of its limited availability and minimal support for application development.

The measurement results of AJ-SR04M can be obtained from equation (1) and the resulting payload size can be of 8 bytes maximum. This is because for getting distance values in cm range at least double precision variables are required which are of 8 bytes.

2.4 Overview of LPWAN technologies

Recent developments in LPWAN technologies have filled the gap of the networking requirements in Internet-of-Things (IoT) landscape. These technologies allow for long range connectivity while keeping power consumption and cost characteristics minimal. Typically, their communication ranges are up to 10 - 40 km in rural areas and 1-5 km in urban areas [20]. Their low power consumption can allow for battery lifetime of 1-10 years depending upon the application [21]. This makes them ideal for outdoor and indoor use cases for a wide array of IoT applications. Among these IoT technologies, Sigfox, LoRa, NB-IoT, CAT-M1 and NB-Fi are the most sought after.

2.4.1 Sigfox

Sigfox is a French LPWAN network operator which provides a solution for LPWAN applications supporting network coverage range in kilometres while consuming very little power. Its infrastructure is based on the company's patented technology and it relies on deployment of proprietary base stations wherever it needs to be supported.

Connection between end devices and base stations make use of sub-GHz unlicensed ISM (Industrial, Scientific and Medical) bands. Frequency band usage is limited to only 100 Hz thus making it very noise insensitive and extremely power efficient. Initially, Sigfox only supported uplink communication i.e., from end device to the base station, and support for bidirectional communication was introduced later. But downlink communication i.e., from base stations to the end devices can follow only after an uplink communication. [22]

Sigfox only allows up to 140 uplink messages per day and the maximum payload length for these messages is constrained to 12 bytes. Moreover, only 4 downlink messages are allowed per day with the maximum payload length limited to 8 bytes. Also, the maximum throughput of the network cannot exceed 100 bps. Sigfox's maximum supported ranges are up to 10 km in urban areas and up to 40 km in rural areas. [22]

2.4.2 LoRa

LoRa is a physical layer spread spectrum modulation technology, innovated by Semtech, which is based on Chirp Spread Spectrum (CSS) modulation scheme. LoRa emerged as a LPWAN operating in the unlicensed ISM frequency bands i.e. 868 MHz in Europe, 433 MHz in Asia, and 915 MHz in North America.

LoRa implements bidirectional communication and it spreads a narrow-band signal to a much wider bandwidth which makes the signal extremely resilient to noise and interference. This also reduces the risk of detection and jamming of the output signal [23]. LoRa provides six spreading factors ranging from SF7 to SF12 which allows users to make suitable trade-off between its range and data rate according to the application requirements. Spreading factor is directly proportional to the range and inversely proportional to the data rate. This means that SF12 supports the longest range with the lowest data rate and SF7 supports the shortest range with the highest data rate. Overall, LoRa data rate's lower bound is 300 bps and upper bound is 50 kbps. Maximum payload allowed for a LoRa packet is 243 bytes. [22]

While LoRa is the physical layer modulation scheme, LoRaWAN is a full fledge communication protocol built on top of LoRa which was first standardized in 2015 under LoRa-Alliance. A typical LoRaWAN network consists of star-of-stars topology in which gateways act as a middle ground between LoRa end-devices and a Network

Server which transmits the packets from the end devices to an Application Server. LoRaWAN works on top of LoRa so, the battery lifetime of any node, the Quality of Service (QoS), the network capacity, and the available security features depend on both the protocol and the network architecture. This protocol and architecture along with its several layers is shown in figure 3.

In LoRaWAN end-devices are not associated with a specific gateway. This means that an end-device can send data packet to multiple gateways. Then each gateway will route this data packet to the network server via some sort of legacy cellular, ethernet or Wi-Fi connection. This network server is designed intelligently so that it can handle redundant data packets, send acknowledgements through optimal gateway, deal with adaptive data rate, and can do some security checks before handling data to the application server. [24]

Figure 3. LoRa and LoRaWAN layers [24]
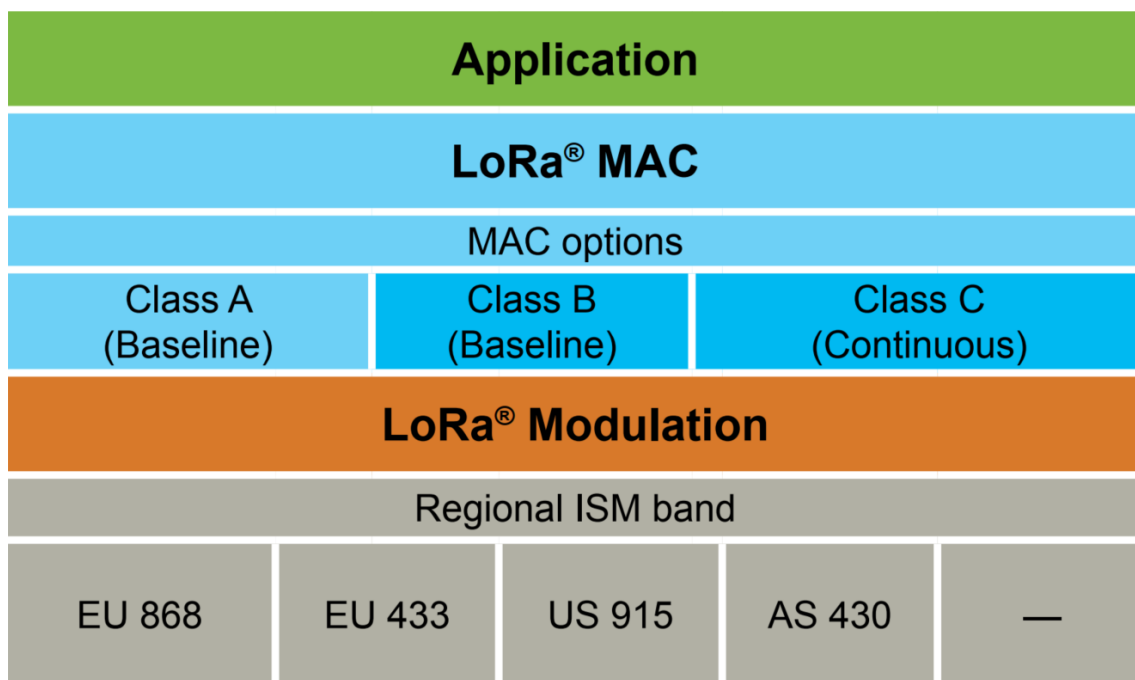
Another important aspect of LoRaWAN is that it provides different classes for end-devices. These classes can help optimize end-device's battery life by differentiating network downlink communication latency.

(Class A) Bi-directional end-devices: End-devices from Class A support bi-directional communication in a way that after each device's uplink transmission there is a follow up

of two short downlink receive windows. This is the most power efficient operating mode for LoRa end-devices as the downlink communication can happen only for a short period of time after an uplink transmission thus increasing the latency.

(Class B) Bi-directional end-devices with scheduled receive slots: At scheduled times, Class B end-devices can allow for extra receive windows for downlink communication in addition to the Class A receive windows. The opening of these scheduled receive windows are controlled by a beacon from the gateway, so that the network server knows when the end-device is ready for the downlink communication.

(Class C) Bi-directional end-devices with maximal receive slots: End-devices from this class have open receive windows all the time unless when the device is transmitting uplink messages. Here the latency is minimised at the expense of much lower battery lifetime of the end-device.

Unlike Sigfox, LoRaWAN does not have any limitations on the maximum amount of uplink or downlink messages [24]. Typically, LoRaWAN network coverage range is up to 5 km in urban areas and up to 20 km in rural areas [22].

2.4.3 NB-IoT

NB-IoT emerged in Release 13 of the 3rd Generation Partnership Program (3GPP) in June 2016 as a cellular Narrow Band technology designed to fulfil the IoT application requirements. NB-IoT is designed in such a way that it can coexist with other cellular standards such as Long-Term Evolution (LTE) and Global System for Mobile Communications (GSM) in licensed frequency bands.

NB-IoT is based on the LTE communication protocol and it borrows some LTE functionalities and makes necessary adjustments to it for IoT application use cases. Hence, we can visualise NB-IoT as a new protocol built specifically for LPWAN applications, but it has a solid structural foundation in the form of LTE. NB-IoT can support up to 100K end devices per base station. The data rate is limited to 20 kbps for the uplink transmission and 200 kbps for the downlink transmission. In terms of the payload size, the maximum payload size allowed for a single message is 1600 bytes. Similar to LoRaWAN, NB-IoT does not have any limit on the transmission of uplink

and downlink messages per day. In terms of range, NB-IoT can have coverage of up to 1 km in urban areas and 10 km in rural areas. [22]

Apart from the normal configuration of NB-IoT in which the end device's modem has to continuously maintain a link with the LTE base station, NB-IoT also supports two different configurations i.e. Power Saving Mode (PSM) and Extended Discontinuous Reception (eDRX).

PSM Configuration:

PSM configuration makes it easier to minimise NB-IoT device's power consumption and can help achieve 10-year battery life. Before PSM, it was possible to switch off a device's radio module to conserve battery, but the device had to reattach to the LTE network on the next boot up. Reattaching to the network can take considerable energy and the cumulative effect of all the reconnections throughout the lifecycle of an application can be huge. PSM configuration eliminates the need for reconnections and still allows for the complete shutdown of the radio module to conserve battery.

PSM can be requested by providing two timers (T3324 and T3412) which the network may accept as it is or with modifications. These values are then retained by the network and the device remains registered on the network. This registration is valid as long as the device wakes up and sends some data to the network before expiration of these timers. A typical PSM cycle can be seen in figure 4.
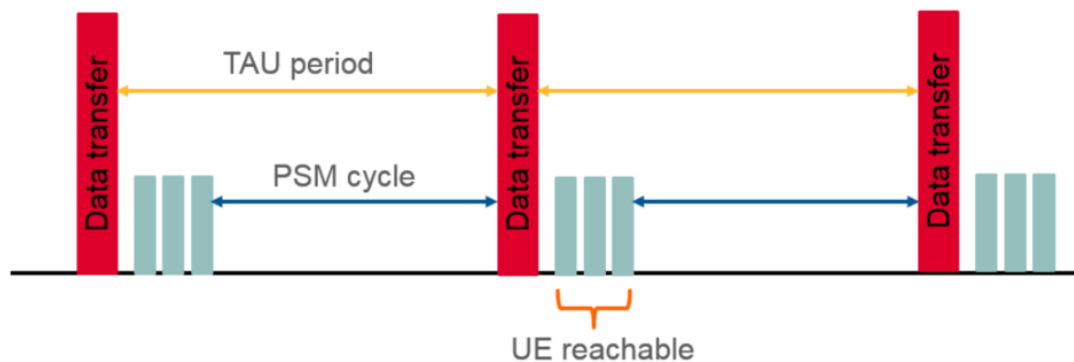
Figure 4. Tracking Area Updating (TAU) period and PSM cycle [25]

29

In PSM configuration, the maximum limit for a device to sleep is 413 days (set by T3412) and the maximum limit for a device to be reachable is 186 minutes (set by T3324). [25]

eDRX configuration:

eDRX is the extension of an LTE feature Discontinuous Reception (DRX) for IoT devices to help reduce power consumption. DRX allows smartphones to momentarily switch off the receiver of their radio module to extend battery life. eDRX can greatly extend the time interval for which NB-IoT device is not reachable by the network. eDRX is not as effective as PSM to reduce power consumption but it offers a balance between the device reachability and the power consumption. Typical eDRX cycle can be seen in figure 5.
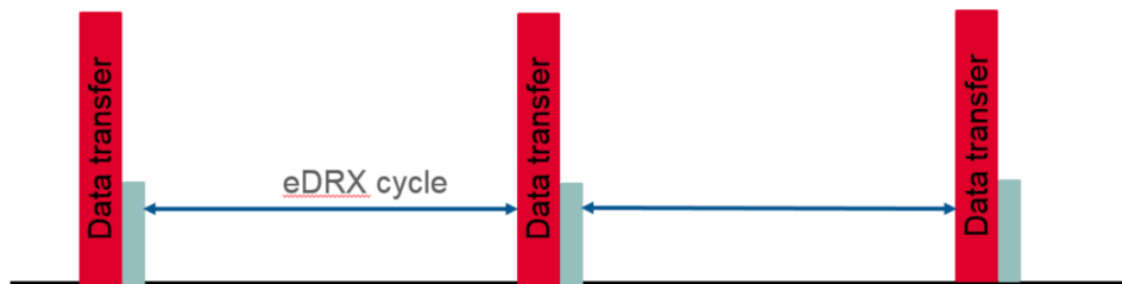
Figure 5. An eDRX cycle [25]

In eDRX configuration, minimum and maximum allowed cycle lengths are 20.48 seconds and 10485.76 seconds (~175 minutes) respectively.

2.4.4 CAT-M1

CAT-M1 was introduced in the Release 13 of the 3GPP and it targeted LPWAN applications as well. Like NB-IoT, CAT-M1 was built to coexist with LTE and GSM protocols in licensed frequency band. Network providers can provide support for CAT-M1 on existing LTE infrastructure though a software update.

CAT-M1 communication utilises 1.40 MHz bandwidth and can provide data rates of up to 1 Mbps in both directions. Compared to NB-IoT, CAT-M1 is ideal for fast moving devices as it can efficiently handle switching between the cell towers much like LTE. Additionally, it includes support for voice over LTE (VoLTE) as well [26].

CAT-M1 specifications fall somewhere between LTE and NB-IoT and it is targeting IoT devices which are more complex as compared to NB-IoT which is ideal for simpler IoT applications.

2.4.5 NB-Fi

NB-Fi is a bidirectional LPWAN communication protocol developed by WAVIoT for IoT and Machine-to-Machine (M2M) applications. NB-Fi protocol can support up to 4.3 million devices in a single network while using 32-bit ID to keep track of each device. NB-Fi does not rely on Internet Protocol (IP) addressing to optimise the payload. By using Non-IP Data Delivery (NIDD) payload size can be decreased by 20 bytes or more as this is the minimum size of an IP header. This helps reduce the complexity and price of the NB-Fi devices.

NB-Fi operates in unlicensed ISM bands and the minimum width of its frequency channel can be 50 Hz. Its maximum payload length is limited to 240 bytes and it can sustain maximum data rate of 25 kbps. Just like Sigfox, NB-Fi limits the maximum number of uplink messages to 3 million and downlink messages to 100k. Maximum range of the NB-Fi network is 10 km in urban areas and 40 km in rural areas [27].

2.4.6 Comparison of LPWAN technologies

Table 2 summarises the important technical specifications of the LPWAN technologies discussed above.

For this application, Sigfox is not ideal because of its two major limitations. First being the uplink message cap at 140 messages per day, and second being the restriction of uplink packet size to 12 bytes [22]. Although, 12 bytes is more than the ultrasonic sensor payload of 8 bytes, it does not leave any overhead for packet headers and metadata. LoRa seems to be a good solution as the payload size and range supported by it are sufficient for UDS monitoring application. Also, it can be optimised for low power usage by switching between different device classes if necessary. In LTE technologies, NB-IoT seems to be ideal with significantly high payload size and decent range in addition to supporting extremely low powered configurations. CAT-M1, on the other hand, cannot be considered for prototyping and testing because its deployment is limited in Estonia as of now. NB-Fi has sufficient payload size and the limit on the

uplink and downlink messages is rather generous, but it does not support additional configurations to tailor the solution for low powered usage.

2.5 Conclusion from background and literature overview

An overview and comparison of different water level sensing mechanisms was presented in section 2.3. After the comparison, ultrasonic ping sensor i.e. AJ-SR04M has been selected for this application, and it will be utilised for the development of prototype. Section 2.4 covered different LPWAN communication protocols which can be feasible for underground communication from manhole. After the comparative evaluation, LoRa and NB-IoT have been chosen as the candidate technologies for the communication part. Feasibility of both of these technologies for manhole communication will be tested in the next section.

Table 2. Comparison of LPWAN technologies

Technology	Frequency	Range	Max. Payload	Data Rate
Sigfox	Unlicensed ISM band 433 MHz, 868 MHz, 915 MHz	10 km (urban) 40 km (rual)	Uplink: 12 bytes Downlink: 8 bytes	100 bps
LoRaWAN	Unlicensed ISM band 433 MHz, 868 MHz, 915 MHz	5 km (urban) 10 km (rural)	243 bytes	50 kbps
NB-IoT	Licensed LTE frequency bands	1 km (urban) 10 km (rural)	1600 bytes	200 kbps
CAT-M1	Licensed LTE frequency bands	1 km (urban) 10 km (rural)	Unknown	1 Mbps
NB-Fi	Unlicensed ISM band 433 MHz, 868 MHz, 915 MHz	10 km (urban) 40 km (rual)	240 bytes	25 kbps

3 Feasibility testing of LoRa and NB-IoT

For low powered communication requirements in UDS, LoRa and NB-IoT came out as the better suited technologies from literature in section 2. Before making a choice between one or the other, feasibility tests were performed at a local manhole for getting a better idea about their performance characteristics in real life. The hardware setup for testing, testing procedure, and results of the testing are presented in this section.

3.1 Hardware setup for NB-IoT testing

The end device for NB-IoT testing was setup using an Avnet Silica NB-IoT BG96 shield [28] with a NB-IoT SIM card from Elisa as shown in figure 6. This shield utilised Quectel BG96 as a cellular modem which can be communicated to via AT commands over serial interface. While testing, different AT commands were issued to the modem for querying Received Signal Strength Indication (RSSI), Signal to Interference and Noise Ratio (SINR), and Narrow Band Reference Signal Received Power (NB-RSRP) which are network connectivity parameters that can be used to evaluate the coverage and signal strength of the network.

Figure 6. Avnet Silica NB-IoT BG96 (Quectel) shield with an NB-IoT SIM

3.2 Hardware setup for LoRa testing

Unlike NB-IoT, LoRa testing requires an end device as well as a gateway which connects LoRa end devices to the internet. LoRa gateway was setup using MultiTech Conduit MTCDT 246L shown in figure 7. This is a mLinux programmable LoRa gateway supporting EU ISM communication frequencies at 868 MHz [29].

Figure 7. MultiTech Conduit MTCDT 246L LoRa gateway

On the other hand, Pycom LoPy 1.0 with Expansion board 3.0, shown in figure 8, acted as a LoRa end device. This end device was programmed to send random packets to the gateway which published these packets on a cloud server. LoRa gateway also embeds the Signal to Noise Ratio (SNR) and RSSI of the received message into the incoming LoRa message before publishing it to the cloud. These parameters can be used to evaluate the network connection between the gateway and the end device.
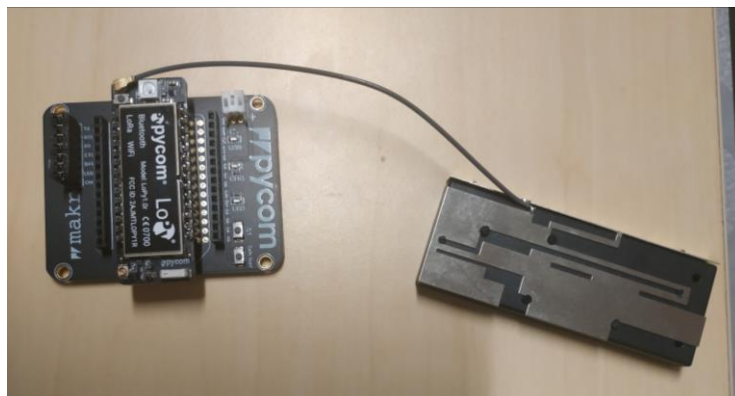

Figure 8. Pycom LoPy 1.0 with Expansion board 3.0

3.3 Testing procedure

Tests were conducted in a similar way for both NB-IoT and LoRa. First, the network connectivity parameters were collected at the ground level and 100 cm above the ground to establish a baseline for these values. Next, end devices were mounted on a stick and placed deeper into the manhole with increments of 20 cm while keeping the lid of the manhole open and the connectivity parameters were noted. Finally, the previous step was repeated with the manhole lid closed off this time.

3.4 Results

3.4.1 NB-IoT results

After NB-IoT testing with Elisa SIM, the results for SINR, RSSI, and NB-RSRP are visualized in figure 9, 10, and 11 respectively. Table 3 summarizes the reference values of these parameters according to 3GPP standard.

Table 3. Reference values of NB-IoT according to 3GPP standard [30], [31], [32]

SINR (dB)	RSSI (dBm)	NB-RSRP (dBm)	Strength
> 12.5	> -65	> -84	Excellent
10 to 12.5	-65 to -75	-85 to -102	Good
7 to 10	-75 to -85	-103 to -111	Fair
< 7	< -85	< -112	Poor

It is evident from the SINR graph that moving further down the ground level makes the SINR values worse even with the manhole open. Moreover, closing the lid makes them even worse by a small factor. The worst SINR value observed was 13.8 which is still excellent as per reference values.

RSSI values came out to be poor even on ground level and they suffer more deterioration moving deeper into the open manhole. Same is true with the manhole closed but putting the cover makes the values significantly worse. When the lid is closed all RSSI values are below −100 dBm which is not ideal, but still reliable communication can be established as receiver sensitivity of this NB-IoT modem is −114

dBm [33]. Thus, there is more room for signal to get even worse before any dropouts can occur.
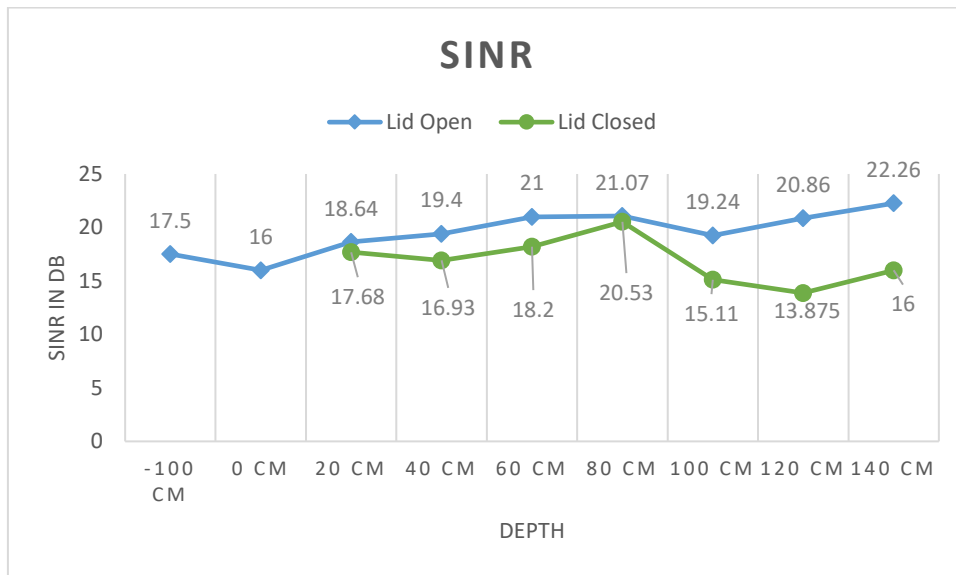
Figure 9. SINR values from NB-IoT testing
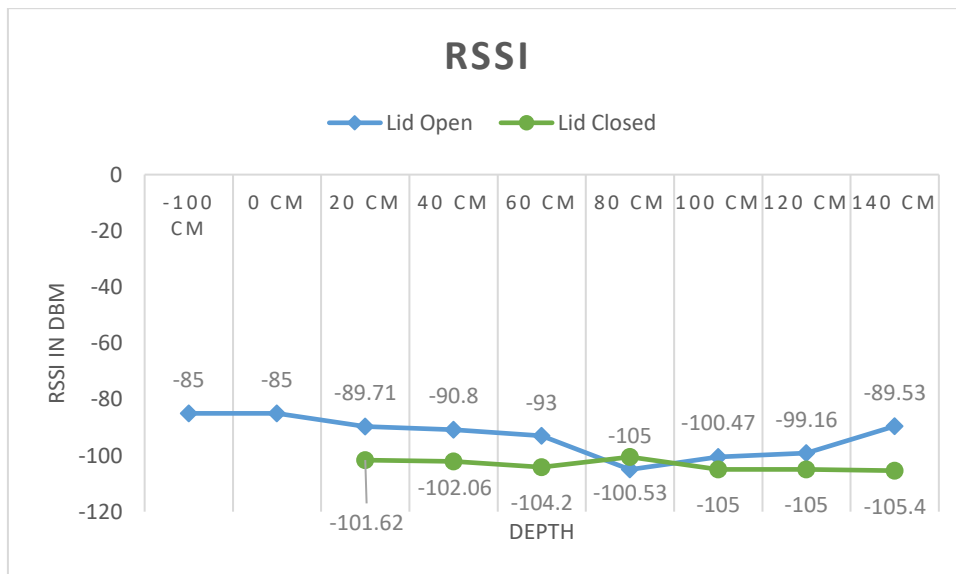
Figure 10. RSSI values from NB-IoT testing

NB-RSRP values go down slowly when end device is situated at more depth in the manhole in both the cases i.e. open and closed. Considering the modem receiver sensitivity of the device, values as low as −114 dBm should pose no difficulty in receiving the packets but in this case values are going further down as well when the lid is closed. Thus, values at more than 80 cm depth with lid closed are lower than −114 dBm indicating the increased probability of packet dropouts. Thus, for establishing

37

reliable communication the device should be placed closer to the ground level preferably above 80 cm.
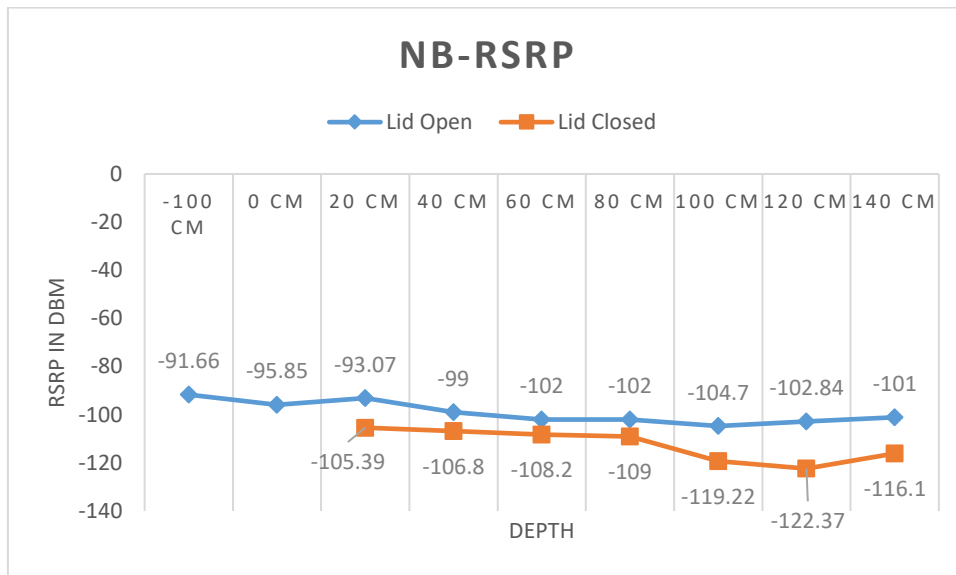
Figure 11. NB-RSRP values from NB-IoT testing

3.4.2 LoRa test results

LoRa SNR and RSSI values obtained from testing are summarized in figure 12 and 13 respectively. Reference values of SNR and RSSI from LoRa documentation are presented in table 4.

Table 4. Reference values of LoRa from its documentation [34]

SNR (dB)	RSSI (dBm)	Strength
10	-30	Strong
-20	-120	Weak

Similar to NB-IoT, the SNR values for LoRa dropped while placing the end device in more depth in both the cases. With the lid closed the values seem to be near the noise floor and at one instance it becomes negative as well i.e. goes below the noise floor. The lowest value observed was −0.8 dB which is way above the −20 dB mark on which LoRa communication is rated to work fine without any major issues.
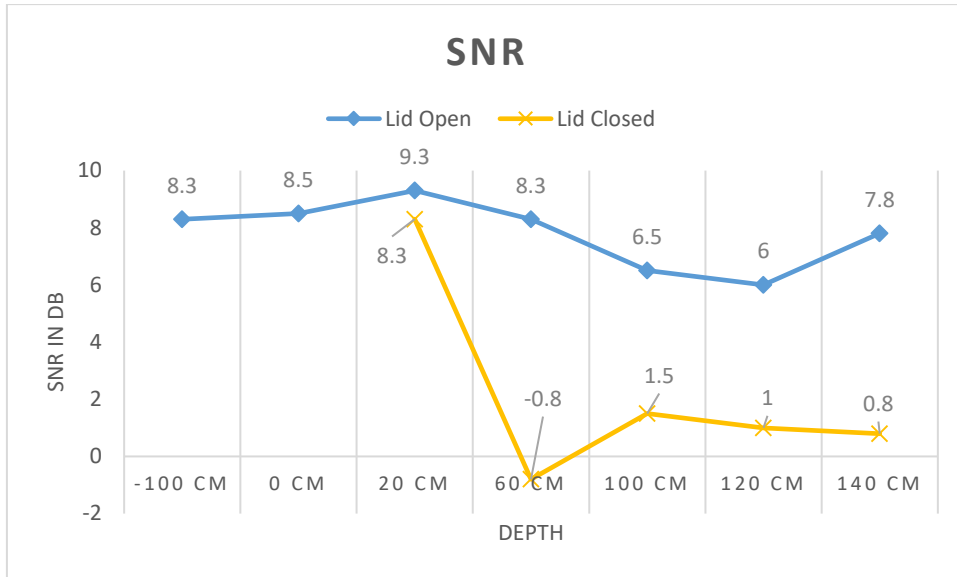
Figure 12. SNR values after LoRa

With regards to RSSI, slight drop in the signal strength can be observed moving down the manhole. Having the lid closed the lowest value observed was −110 dBm which is feasible for communication as lowest threshold for LoRa communication is −120 dBm.
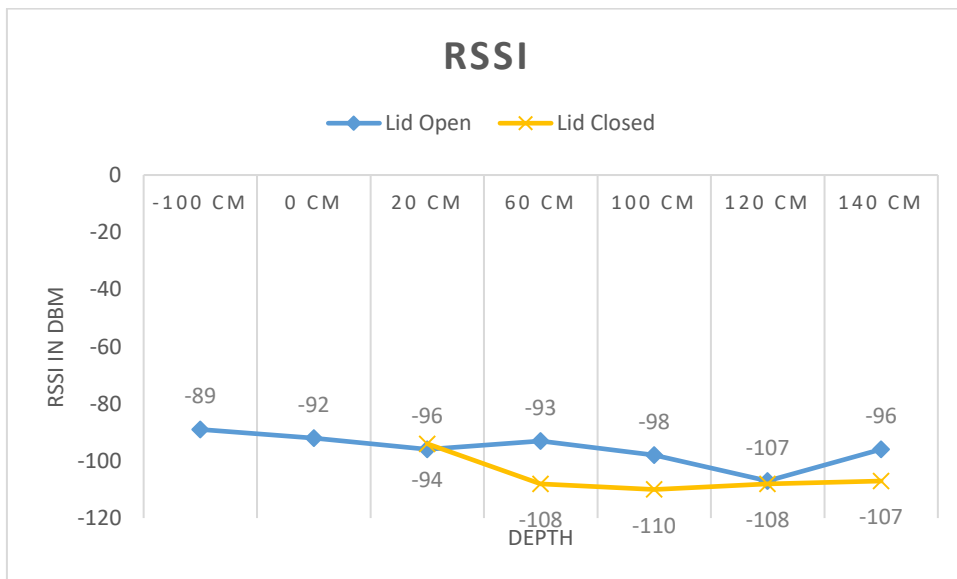
Figure 13. RSSI values from LoRa testing

3.5 Conclusion from feasibility testing

After conducting these tests, both LoRa and NB-IoT turned out to be viable communication solutions for this application. In terms of signal propagation, LoRa holds an edge over NB-IoT as the network connectivity parameters for it turned out to

39

be excellent at even extreme depths inside the manhole. This is to be expected because of the requirement of dedicated gateway for LoRa communication which can provide better results but with additional development costs as a trade-off. On the other hand, NB-IoT signal strength parameters values are weak as indicated by the reference values but still within the modem sensitivity limit of -114 dBm at depth of 80 cm or less. For UDS monitoring, the device will be placed very close to the surface of manhole within $10 - 20$ cm so this does not seem to be an issue.

From usability and commercial point of view, NB-IoT is much simpler because it doesn't involve any technical debt solving of installing, configuring, and maintaining of gateways. This is the motivation behind choosing NB-IoT as a communication solution for this work. In the next section, a comparison of different NB-IoT development platforms will be presented before moving on with the prototype development.

4 NB-IoT development platforms

For NB-IoT development, different platforms are available, and they can be differentiated based on their hardware configuration, form factor, software development experience, power consumption, and pricing. It is worth taking a look at different platforms available in the market before moving on with the prototype development. In this section, a brief overview of three platforms utilising different chipsets will be presented, and their suitability is evaluated based on the application requirements. Similar performance characteristics can be expected from dozens of other development platforms built on the same chipsets as these ones. Thus, this comparison provides an overview of variety of different platforms available in the market for NB-IoT development.

4.1 Quectel BG96

Quectel's BG96 is a widely used LTE module supporting NB-IoT communication along with CAT-M1 and EGPRS. It comes with Land Grid Array (LGA) packaging and has a compact form factor (26.5 mm x 22.5 mm x 2.3 mm) which makes it extremely flexible and versatile for different application use cases. Since, Quectel has a lot of experience in shipping various cellular products, their hardware designs, operating manuals, and application notes are very comprehensive. Also, as being the major shareholder in the IoT landscape, they have big a development community surrounding their products.

BG96 is optimised for ultra-low power consumption and supports NB-IoT power saving configurations i.e. eDRX and PSM. According to its data sheet, its typical current consumption on PSM is 10 µA [35]. One drawback of working with BG96 is that it can only be used by interfacing it with some other controller over USB, UART, I2C etc. so the power consumption of that device needs to be factored in as well.

4.2 uBlox SARA-R4

SARA-R4 is a series of NB-IoT, CAT-M1, and EGPRS modules developed by uBlox supporting different LTE bands for different geographical regions. These boards come in 96 pin LGA packaging with the 16.0 mm x 26.0 mm x 2.4 mm physical dimensions which allows hardware designers to integrate them with different chipsets in small form factors. An example of such integration is Arduino MKR NB 1500; combining Arm Cortex-M0 32-bit with uBlox SARA-R410M-02B [36]. These uBlox boards have excellent documentation and usage guidelines and are widely used in the community after being adapted by Arduino for LTE solutions.

SARA-R4 series are designed while keeping power conservation in mind and support NB-IoT power saving configurations such as eDRX and PSM. Their typical current consumption on PSM mode is rated at 8 µA in the datasheet [37]. Like BG96, these modules can be used in tandem with some other microcontroller over UART, USB, I2C which increases the power consumption significantly.

4.3 Nordic Semiconductor nRF-9160

Nordic Semiconductor's nRF-9160 is a highly integrated SiP supporting NB-IoT / CAT-M1 networks. This module combines an Arm Cortex-M33 application processor with a Nordic's inhouse LTE modem and includes a power management system. The result is a miniature LGA package with physical dimensions of 10 mm x 16 mm x 1 mm capable of running application code and LTE communication. The hardware is supplemented with decent documentation and user guides. The application development is supported through a Real Timer Operating System (RTOS) which gives excellent control over hardware peripherals and allows for heavy optimizations to conserve power. But since the platform is new there is little to no availability of sensor libraries and third-party software which can make the development extremely challenging and time consuming.

nRF-9160 is excellent for low powered LTE applications as the current consumption on NB-IoT eDRX and PSM are extremely low. According to its product specification, PSM floor current values for modem are rated at 2.7 µA whereas application processor

on idle consumes 1.8 µA; making total of 4.5 µA [33]. The biggest advantage of nRF-9160 is that it does not require some external controller and it can work on its own.

4.4 Comparison of the development platforms

Some of the important specifications and features of the NB-IoT development platforms discussed are summarized in table 5.

Table 5. Comparison of BG96, SARA-R4, and nRF-9160 development platforms

Specs and Features	Quectel BG96	uBlox SARA-R4	Nordic nRF-9160
Packaging	LGA	LGA	LGA
Form Factor	26.5 mm x 22.5 mm x 2.3 mm	16.0 mm x 26.0 mm x 2.4 mm	10 mm x 16 mm x 1 mm
External Application Processor Required	Yes	Yes	No
PSM Current	10 µA (Modem only)	8 µA (Modem Only)	2.7 µA (Modem) 1.8 µA (Application Processor on Idle) Total: 4.5 µA
Price (per unit)	16.09 Euros	22.10 Euros	19.73 Euros

4.5 Conclusion from comparison of development platforms

It is evident from the comparison of these NB-IoT development platforms that Nordic nRF-9160 offers significant advantages over its counterparts at a competitive price. Its highly integrated design eliminates the need for external microcontroller thus making it extremely power efficient. Thus, its application processor and NB-IoT modem's PSM current consumption is lower than the PSM current consumption of BG96 and SARA-R4. That is without considering the current consumption of the external microcontroller required to work with BG96 and SARA-R4. Considering these factors, nRF-9160 is chosen as the NB-IoT development platform for this work.

5 Prototype Design and Development

The system prototype designed in this work for low powered underground drainage monitoring is shown in figure 14. The end device is made up of Nordic Thingy 91 which is interfaced with an ultrasonic ping sensor AJ-SR04M. It is programmed to take water level measurements and communicate them securely with the Amazon Web Services (AWS) IoT MQTT server via NB-IoT network after specific intervals of time. During these intervals, the device will switch to low power mode and conserve as much energy as possible to maximize its battery life.
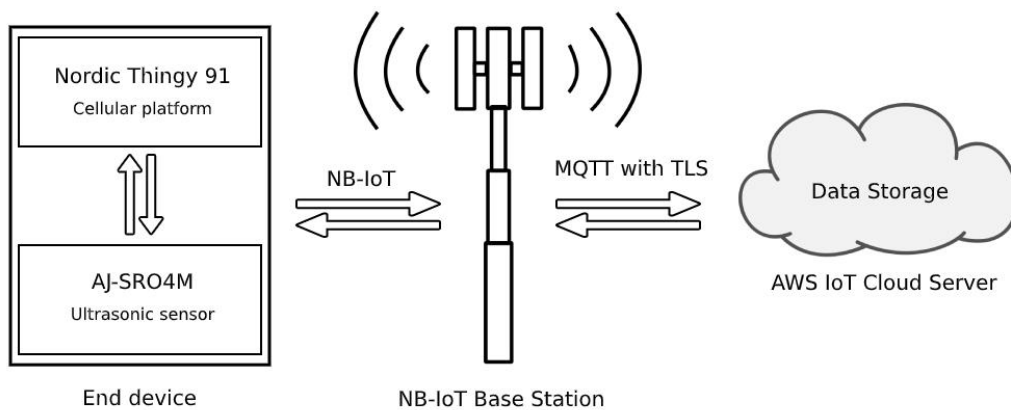
Figure 14. Block diagram of the developed prototype

5.1 Hardware description

5.1.1 Nordic Thingy 91

Nordic Thingy 91, shown in figure 15, is a NB-IoT prototyping platform equipped with Nordic's nRF-9160 SiP which is well optimized for low powered applications supporting NB-IoT PSM and eDRX modes. It provides with different peripherals and interfaces such as GPIOTE, UART, I2C, SPI, RTC, and PPI etc. for all sorts of hardware component integration and application development. Additionally, the device has a small physical footprint which makes it perfect for remote monitoring and asset tracking applications.

Figure 15. Nordic Thingy 91 development kit

5.1.2 AJ-SR04M Ultrasonic sensor

AJ-SR04M is an ultrasonic sensor which provides support for 20 cm – 6 m non-contact distance measurement with 1 cm of measurement accuracy [15]. The sensor module integrates an ultrasonic transceiver with a control circuit. Some of its distinctive features are; small form factor, low power consumption, very high measurement resolution, and waterproofing of transceiver probe. All these features make it a perfect fit for this work. Important specifications of this sensor are summarised in table 6.

Table 6. Specifications of AJ-SR04M quoted from datasheet [15]

Specifications	Ratings
Operating voltage	3.0 V – 5.5 V
Current consumption	< 8 mA
Maximum range	6 m
Minimum range	20 cm
Measurement accuracy	± 1 cm
Resolution	1 mm
Operating temperature range	− 20 to 70 degree Celsius

This sensor can be interfaced either through GPIO or UART and supports different modes of operation. For this prototype, GPIO interface was used and the mode of operation is summarized below:

1. Sensor is triggered by sending a 10 us high pulse to the input pin labelled as 'Trig' or TX.

2. Module gets activated and transmits eight 40 kHz square waves through the ultrasound transceiver and immediately starts listening for the reflected pulses.

3. Sensor's response on the 'Echo' pin becomes a high signal for the duration of time it takes by the reflected signal to come back. Distance can then be calculated by using equation (1).

4. If the transmitted signal does not come back the high response from the sensor automatically becomes low after 38 ms, thus declaring the end of measurement.

The ultrasonic timing diagram in this mode is shown in figure 16.
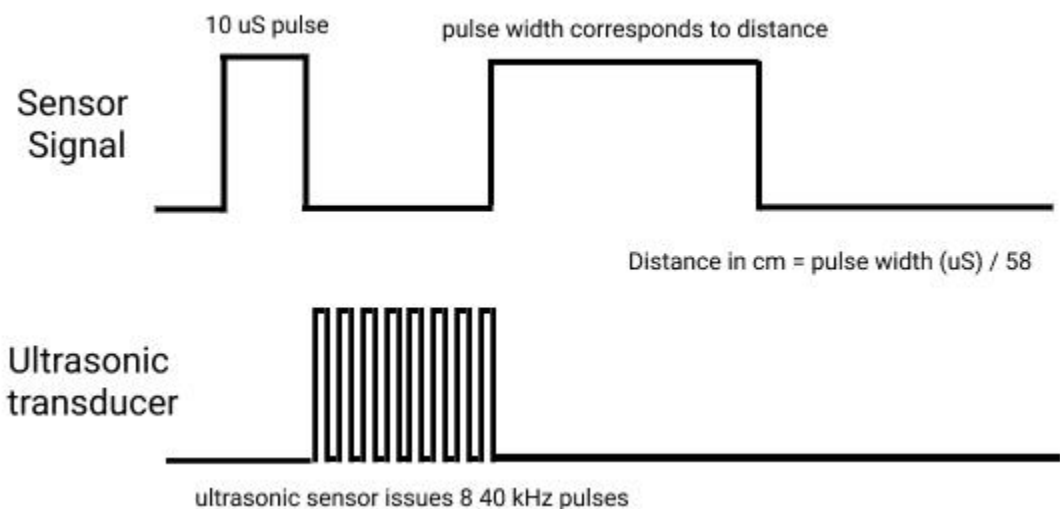
Figure 16. Timing diagram of AJ-SR04M in utilised mode

5.1.3 Integrating AJ-SR04M sensor with Thingy 91

As discussed in the previous section, AJ-SR04M was integrated with the Thingy 91 board over GPIO pins as shown in figure 17. The block diagram highlighting relevant connections between AJ-SR04M and Thingy 91 is shown in figure 18.

46
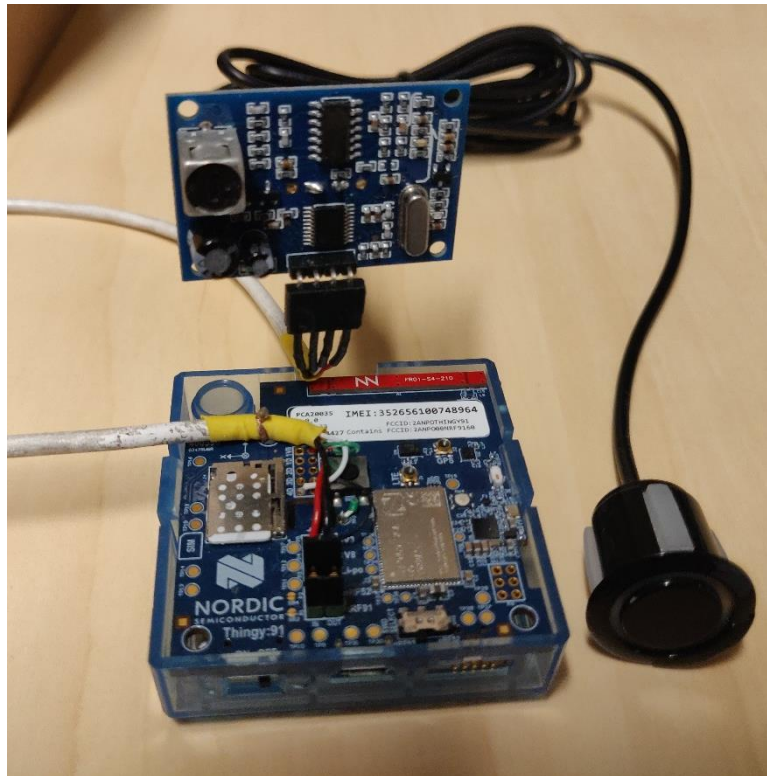
Figure 17. End device: Thingy 91 development kit interfaced with AJ-SR04M sensor
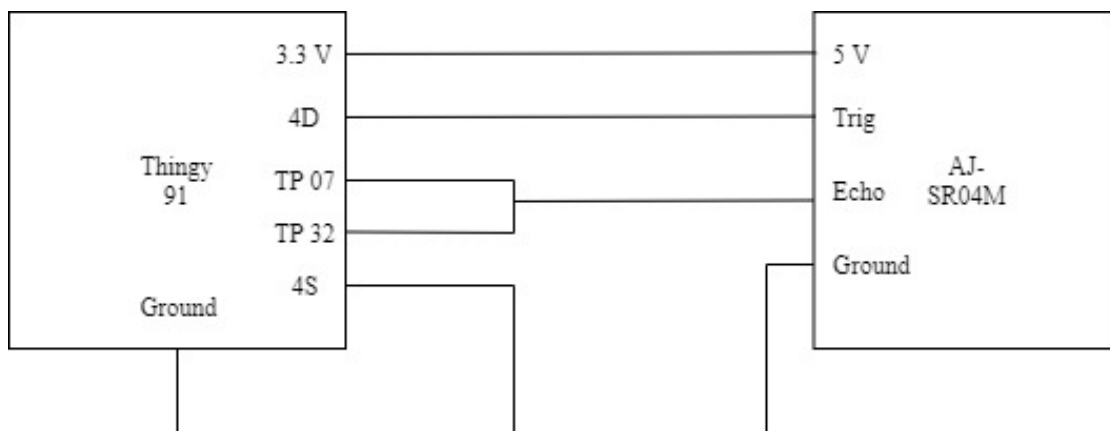
Figure 18. Schematic diagram of connections between Thingy 91 and AJ-SR04M

Input to the sensor over 'Trig' pin requires operating voltage of at least 3.0 V but nRF-9160 GPIO pins can output only 1.8 V. To overcome this problem, Thingy 91 provides N-MOS transistors with drain and source available on the external connectors while the gate is connected to nRF-9160 GPIO pins. So, for triggering the sensor an N-MOS transistor was used which was toggled by a GPIO output pin to get the required voltage waveform for the sensor. Output from the sensor over 'Echo' pin can be received

directly over a GPIO input pin. But the output was received over two GPIO input pins as this makes the delay calculation easier and precise.

5.2 Software development

The application development for this prototype mostly involved working with nRF Connect SDK which is an SDK provided by Nordic Semiconductor for developing software for nRF91, nRF52, and nRF53 series devices. This SDK comes with cellular IoT, Bluetooth, Thread, and Zigbee libraries, different application samples, and drivers for Nordic Semiconductor's devices.

nRF Connect SDK is based on Zephyr RTOS which is an open source, small, scalable, and secure operating system developed for connected devices. Along with Zephyr, it comes with nrfxlib which is a repository containing RTOS-independent drivers and libraries for Nordic Semiconductor's devices. Additionally, it makes use of MCUboot as well which is an open source secure bootloader for 32-bit microcontrollers.

Apart from nRF Connect SDK, Nordic provides nRF command line tools, nRF programmer, and SEGGER Embedded Studio Integrated Development Environment (IDE), and a tool chain manager for making the development process smoother.

5.2.1 Tools

The application developed for this prototype is based on nRF Connect SDK v1.4.0 which is the latest stable release at this time. Zephyr, nrxflib, and MCUboot release tags associated with this build are 2.4.0, 1.4.0, and 1.6.99, respectively. SEGGER Embedded Studio for ARM (Nordic Edition) v5.10d (64-bit) was used for compiling and building the application code.

5.2.2 Application description

The developed monitoring application sends ultrasonic distance measurements to the cloud server after specified interval of time. The sleep interval in normal operating mode is set to 60 minutes. The application is heavily optimized for low powered usage and turns off all the unnecessary peripherals, modem, SIM when they are not being used. NB-IoT's PSM is utilized to avoid re-registrations on the network even after

restarting of modem and SIM. The flow of the developed application can be summarized in the following steps:

1. The application first configures the LTE modem, establishes connection with the NB-IoT network, and requests for PSM from the network. This is done only once, and it can consume a lot of power.

2. Then, the connection to the chosen cloud server is established over secured MQTT protocol.

3. After the cloud connection is ready, all the peripherals required for sensor measurement i.e. GPIO, GPIOTE, GPIOTE Interrupts, and Timer, are activated.

4. The sensor is triggered and the response from the sensor is noted. Distance value is calculated from the sensor response.

5. The calculated distance value is embedded into a JSON object and sent to the cloud server. All the peripherals needed for sensor measurement are then turned off to save power.

6. The device is disconnected from the cloud and the device waits for its radio resources to go idle.

7. Once radio resources are free, low power mode is activated for a specified interval of time. In the low power mode, modem and SIM card are turned off and application processor is on idle to conserve power.

8. Finally, device wakes up from low power mode once the specified time interval is complete and steps 2 – 8 are repeated forever.

For this application, Zephyr kernel services and its core APIs are used for flow control, sleep modes, and power management aspects, whereas networking, cloud communication, and sensor integration are handled by nRF Connect SDK libraries and nrfxlib drivers.

5.2.3 Zephyr RTOS

Some of the important Zephyr kernel objects and APIs utilised for the application development are described below.

System Threads:

These are the main threads which are spawned by the Zephyr kernel during system initialization. Two of the system threads which are spawned in every application are Main thread and Idle thread. Main thread is assigned the highest priority of 0 by default, and it is responsible for initialization of the kernel along with different peripherals. Once the initialization is successful, Main thread calls the main () function defined in the application code. Idle thread, on the other hand, is assigned the lowest configurable priority and is executed when there is no other work for system to do. This thread is responsible for dealing with the power management subsystems as well. Both of these threads are system critical and can cause the system to crash if any one of them is aborted. Since it is not recommended to handle additional application related tasks using these threads, Workqueue threads were used for processing sensor, and cloud communication related tasks.

Workqueue Threads:

A Workqueue is a first in first out based kernel object that spawns a dedicated thread to process its work items. Each work item is a memory address associated with a function which is called on the execution of that work item. A delayed work item can also be assigned to a workqueue thus allowing for scheduling of work items. Workqueues are flexible as any number of work items can be assigned to them with only limitation being the available memory on a particular device.

In this application code, a workqueue with a size of 2048 bytes was configured, and two delayed workqueue items were initialised. One establishes connection to the cloud and the other one handles the sensor data acquisition and cloud communication. This way these tasks can be scheduled with the delay equivalent to sleep time and upon completion Idle system thread is triggered for power saving and switching to sleep mode.

Power Management subsystem and Sleep states:

Zephyr's power management subsystem supports several sleep states with different CPU, memory, and peripheral configurations. The power management features utilised for this application are categorised in System Power Management.

In System Power Management, when an idle thread is triggered the power management subsystem can put the system in either Sleep state or Deep Sleep state depending upon the duration of the idle time or selected power management policy. In Sleep state, CPU is halted but its execution context stays intact and some of the peripherals can be power gated. Waking up latency in this state are relatively low. On the other hand, in Deep Sleep state CPU is power gated and it loses its execution context. On wake-up, OS start up code must be executed again but memory can be selectively retained. More power can be saved in this state at the expense of higher wake up latencies.

These configurations are especially relevant for this application because for using NB-IoT PSM mode, execution context of the application processor and memory needs to be retained. Also, losing the execution context means running the OS start-up code from the beginning and initialising kernel services and peripherals on every wake up which can consume a lot of power. Thus, Sleep state with the application processor on Idle thread is the optimal way to go. While on Idle thread SIM, modem, and other unused peripherals used for communicating with the sensor can be power gated thus minimising the power consumption.

5.2.4 nRF Connect SDK libraries

nRF Connect SDK libraries critical for working with the modem, cloud, and networking sockets are presented below.

LTE link controller:

This is the library for interacting with the modem of nRF-9160 SiP. It provides different variables and functions definitions to have control over LTE configuration and connection. This library was used to configure LTE modem for NB-IoT connection, enabling and requesting PSM, and shutting down and waking up the modem. The relevant settings and PSM parameters requested for this application are specified in figure 19. These PSM parameters translate to 60 minute of TAU period while Active time is 0. Thus, device won't have to reregister to the network if it reconnects within a 60 minute window after every disconnection.

```
39   # LTE link control
40   CONFIG_LTE_LINK_CONTROL=y
41   # CONFIG_LTE_NETWORK_MODE_LTE_M=y
42   CONFIG_LTE_NETWORK_MODE_NBIOT=y
43   CONFIG_LTE_AUTO_INIT_AND_CONNECT=n
44
45   # PSM
46   CONFIG_LTE_PSM_ENABLE=y
47   CONFIG_LTE_PSM_REQ_RPTAU="00100001"
48   CONFIG_LTE_PSM_REQ_RAT="00000000"
49
```

Figure 19. LTE link control and PSM settings for the developed application

Cloud API:

Cloud API establishes the requirements that a cloud driver must implement in order to establish connection and communication with a cloud backend. Cloud API was used with "AWS_IOT" backend to establish secure communication with (AWS) IoT MQTT server.

Socket:

Socket library is a part of Networking API in nRF Connect SDK which provides different socket implementations compatible with Berkeley Software Distribution (BSD) socket API. This library was used to open MQTT sockets which were sent to the cloud server.

5.2.5 nrfxlib drivers

nrfxlib provides a HAL and different drivers for interacting with various peripherals of Nordic Semiconductor's device chipsets. These were used for sensor related configuration and measurements because they can interact with the peripherals directly without having any involvement from the kernel. Thus, sensor measurements can be made with excellent precision and minimum delays.

GPIO HAL:

GPIO HAL implements various data structures and functions for configuring different GPIO pins of the nRF-9160. GPIO HAL was used to configure the GPIO pin for driving input waveform to the sensor.

GPIOTE HAL and drivers:

GPIOTE is a hardware peripheral that can be used to assign tasks and track events on different GPIO pins of the nRF-9160. GPIOTE HAL and drivers provide software infrastructure to make use of this peripheral on different channels.

GPIOTE events were assigned to two different GPIO pins for monitoring response from the sensor's output. High Event marks the beginning of the sensor response and Low Event marks the end. This way of generating interrupts allowed for more precise measurements from the sensor with less time delays.

Timer HAL and drivers:

Timer HAL and drivers make the handling of the hardware timers possible without needing any support from the RTOS. The resolution of the sensor measurements rely on the accuracy of the measured timing events from the sensor response. So, Timer HAL and drivers were used to configure a timer with 32-bit width and 16 MHz clock which is the maximum supported by the Timer peripheral of nRF-9160. This allowed for timing sensor response in nano second accuracy which is ideal for measurements from the sensor.

5.3 Cloud platform

In this prototype, AWS IoT Hub has been chosen as a cloud platform and the monitoring information is being sent to AWS IoT MQTT server. AWS has been chosen because of the seamless integration of its different web computing services which can be helpful for UDS monitoring applications such as AWS Serverless or AWS Lambda. Moreover, AWS Firmware Over-the-Air (FOTA) can be used to remotely update the firmware of the devices over HTTP or MQTT TLS. Security considerations have also been a deciding factor for choosing AWS as the cloud platform. AWS makes it really simple to assign security policies and certificates to different devices and these features have been utilised for the development of this prototype.

5.3.1 AWS IoT MQTT server

In the developed prototype, communication with the cloud server is handled over AWS IoT MQTT server. MQTT is a lightweight messaging transmission protocol based on publish / subscribe principle developed for various IoT applications. MQTT protocol is ideal for IoT use cases because it is designed to work on resource constraint devices

while consuming very little power. Moreover, it is capable of delivering bi-directional messages on unreliable networks with very little network bandwidth utilisation. Also, MQTT can operate beneath TLS security layer for enabling secure communication with minimal performance overhead. All of these factors make MQTT ideal communication protocol for this application. Currently, AWS IoT offers major support for MQTT v3.1.1 specifications and features with only minor differences.

5.3.2 AWS IoT policies

AWS IoT policies help limit the access to AWS IoT core by allowing only legal device actions and denying illegal ones as determined by the device's policy document. The policy document is a JSON object containing one or more policy statements. Each statement must contain:

- Effect: which determines if the action is allowed or denied.

- Action: which determines the action targeted by the policy.

- Resource: which specifies the resource for which the action is allowed or denied.

Example of some of the basic actions can be: iot:Connect, iot:Publish, iot:Subscribe, iot:Receive etc. This approach is very flexible and policies can be updated easily if the application requirement changes. The policy document defined for this prototype device is shown in figure 20.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iot:*",
      "Resource": "*"
    }
  ]
}
```

Figure 20. Policy document for end device

5.3.3 AWS IoT Encryption

AWS IoT supports encrypted communication over TLS for server and client authentication. Server authentication helps ensure that the device is communicating with

a valid AWS IoT endpoint. Client authentication means that the end device needs to prove its authentication for communicating with the AWS IoT server.

AWS IoT relies on public key cryptography or asymmetric cryptography for authentication purposes. Public key cryptography makes use of a pair of keys for transmission of information securely. Sender encrypts a message using a public key and transmits it to the receiver who can only decrypt the message using a private key. Public and private keys can also be used to verify that the original message has not been altered with. Anyone with a public key can check the encrypted payload and can tell if the message originated from a sender with a valid private key or not. This is a fool proof way of sharing information securely as long as the private key is secured.

On AWS IoT, X.509 certificates are used to manage the generation and ownership of the public key. X.509 certificate can be obtained by making a Certificate Signing Request (CSR) from a Certificate Authority (CA). A CSR contains public key and some other identification information. The CA validates the requested CSR and signs it with a private key. Now, the generated certificate can be validated by anyone using the CA's public key. This mechanism of generating X.509 certificates has been shown in figure 21.
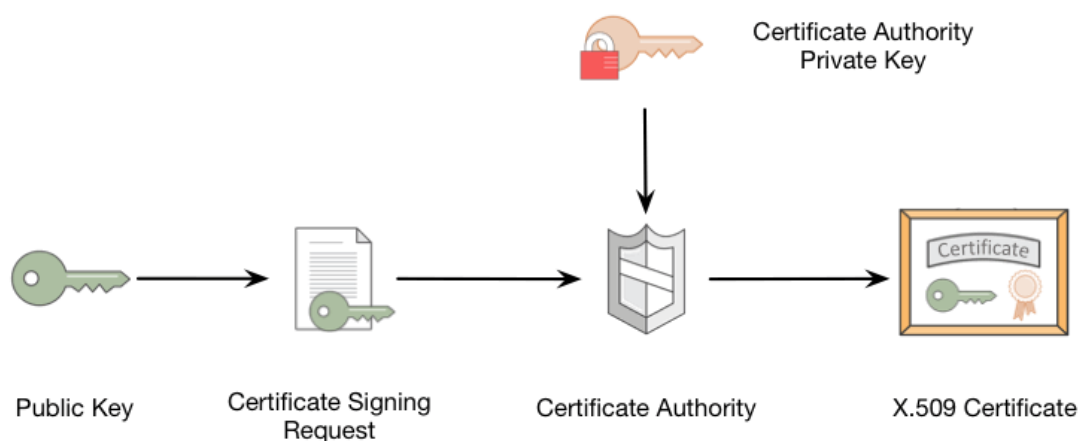

Figure 21. Mechanism of generating X.509 certificates [38]

Usage of AWS IoT certificates:

For establishing secure communication with the AWS IoT MQTT server and the end device, an X.509 certificate needs to be generated. This can be done by either using

Amazon one-click generation which makes use of AWS IoT CA, coming up with own CSR and signing it with AWS IoT CA, or using own certificate signed by some other trusted CA. For this application, AWS IoT certificates were generated by AWS IoT CA using one-click generation in the following manner:

- The CA certificate for server authentication was issued by Amazon Root CA 1 in RSA 2048-bit key format which is necessary for AWS IoT endpoint.

- Client certificate, public key, and private key were generated and downloaded from AWS IoT Hub.

These certificates were then assigned to this device and activated so that the end device can only establish connection after server and client authentication. Then, these certificates were uploaded to nRF-9160's memory. This can be done with the help of Certificate Manager in nRF Connect LTE Link Monitor application as shown in figure 22. CA certificate, Client certificate, and Private key values need to be copied in the respective fields and a security tag is assigned so that they can be referenced in the application code later.
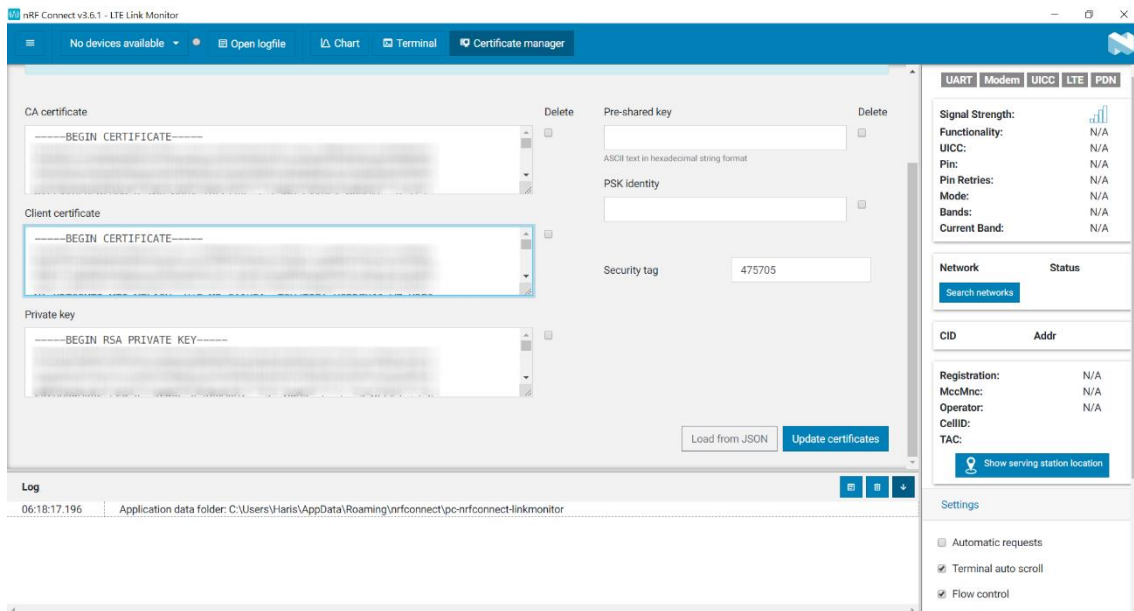

Figure 22. Uploading certificates to Thingy 91 via LTE Link monitor

Finally, the relevant application settings need to be modified as highlighted in figure 23. MQTT communication port needs to be changed to '8883' from '1883' as this is the right port for communication with TLS enabled. Also, the security tag chosen while

uploading certificates need to be specified so that the correct certificates can be used for establishing connection.

```
# AWS IoT
CONFIG_AWS_IOT=y
CONFIG_AWS_IOT_PORT=8883
CONFIG_AWS_IOT_CLIENT_ID_STATIC="nrf-          "
CONFIG_AWS_IOT_BROKER_HOST_NAME="                    .amazonaws.com"
CONFIG_AWS_IOT_SEC_TAG=475705
CONFIG_AWS_IOT_TOPIC_UPDATE_DELTA_SUBSCRIBE=y
CONFIG_AWS_IOT_CONNECTION_POLL_THREAD=y
```

Figure 23. Application settings for secured MQTT connection

5.4 Conclusion of prototype development

Section 5 presented an overview of hardware integration and configuration steps relevant to development of the prototype. In addition, details about the tooling, development environments, SDKs, and APIs necessary for the application development and power optimizations have also been discussed. Lastly, cloud communication protocol, security features, and corresponding application settings were highlighted as well.

6 Testing and Results

This section deals with the testing and analysis of the developed prototype for low powered underground drainage monitoring. In the first part, sensor testing procedure and setup is discussed and the results of the testing are presented. In the second part, the power consumption of the developed solution is analysed and battery life cycle estimations are made.

6.1 Ultrasonic distance sensor testing

For evaluating the quality of distance measurement results from the sensor, the prototype was tested in the lab environment at different distance levels and the measurement data was recorded. These sensor measurements were compared with the actual distance measurements taken from a measurement tape and error was calculated and corrected. Mean, Standard Deviation and Standard Error were calculated from recorded measurements for each distance level. The testing setup is shown in figure 24.

Figure 24. Ultrasonic distance sensor testing setup

Absolute Error for measurements can be calculated using equation (2),

$$\text{Absolute Error} = \text{Actual value} - \text{Measured value} \qquad (2)$$

Based on the magnitude of Absolute Error, Error Correction can be applied using equation (3),

$$\text{Error Correction} = \text{Absolute Error} - \text{Minimum Error Offset} \qquad (3)$$

Mean of the values can be calculated using equation (4),

$$\text{Mean} = \frac{\text{Sum of all values}}{\text{Total number of values}} \qquad (4)$$

Standard deviation of the measurements was calculated using equation (5),

$$\text{Standard Deviation} = \sqrt{\frac{\sum (x - x_{\text{Mean}})^2}{N}} \qquad (5)$$

where 'x' is the set of data values and 'x_{Mean}' is the mean of those values and 'N' is the total number of samples in that data. Standard Error can be calculated using equation (6),

$$\text{Standard Error} = \frac{\text{Standard Deviation}}{\sqrt{N}} \qquad (6)$$

where 'Standard Deviation' was calculated from equation (5) and 'N' is the total number of samples in the dataset.

6.1.1 Sensor testing results

Distance measurements were recorded at 30, 60, 70, 80, 100, and 130 cm. Mean, Absolute Error, and Error Correction calculations at each of these levels are summarised in table 7, whereas Standard Deviation and Standard Error of the measurements at different distances are plotted in figure 25.

Table 7. Mean, Absolute Error, and Corrected Error of measurements

Results	30 cm	60 cm	70 cm	80 cm	100 cm	130 cm
Mean	26.65 cm	58.16 cm	67.06 cm	76.87 cm	96.12 cm	123 cm
Absolute Error	3.35 cm	1.84 cm	2.94 cm	3.13 cm	3.88 cm	7 cm
Corrected Error	1.51 cm	0	1.1 cm	1.29 cm	2.04 cm	5.16 cm

From table 7, it can be interpreted that the magnitude of Absolute Error is varying from one distance level to another. Assuming the presence of static sensor offset, Error Correction is applied by subtracting the minimum error offset value of 1.84 cm from all the values. Now, the Corrected Error is a little above 2 cm for distance up to 100 cm and after that it jumps to 5.16 cm. Thus, measurement accuracy of sensor from these measurement samples can be regarded as ± 2 cm at distance in the range of 1 m and ± 6 cm for distance beyond that.

Standard Deviation of measurements from 30 cm up to 100 cm is approximately 0.5 cm and Standard Error is less than 0.07 cm. Thus, measurements seem to be precise with only ± 0.5 cm deviation up to a distance of 100 cm. But with the increase in distance these values got significantly worse and at 130 cm the standard error reaches 0.5 cm which can make the measurements very imprecise. Thus, this sensor seems to have precision of ± 0.5 cm up to 100 cm or 1 m and after that precision decreases to ± 4.5 cm.
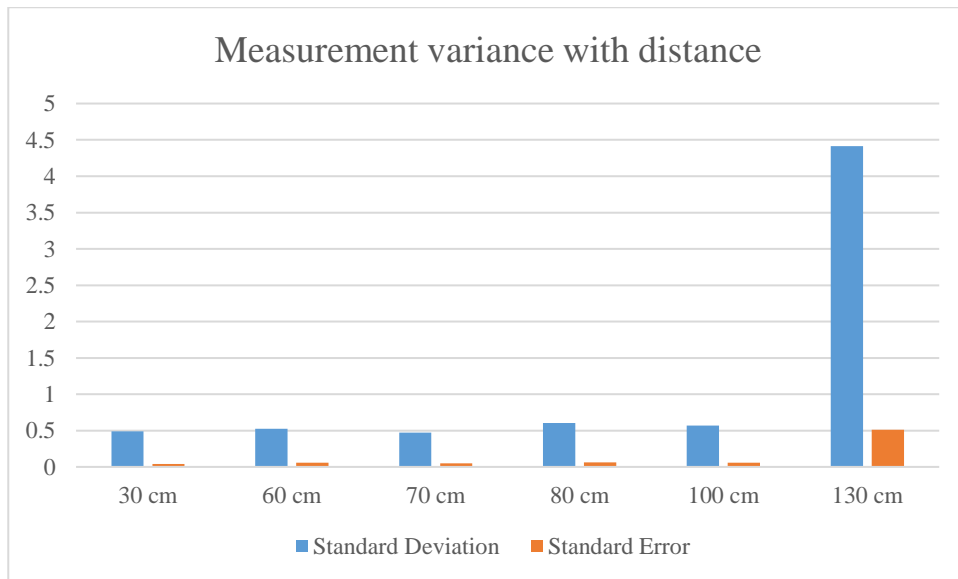
Figure 25. Standard Deviation and Standard Error of measurements

The sensor's performance seems to be significantly worse than the rated specs in the datasheet [15]. Also, huge performance drop can be observed in the measurement values moving past 100 cm distance range. Through further testing of sensor with other microcontroller boards, it was found out that the drop in sensor's performance is related to the lower voltage supply. Although the sensor's spec sheet specifies a range of 3.0 V

– 5 V as the operational voltage, the results become inaccurate and imprecise at voltages on the lower end of this range. Since nRF-9160 GPIO pins are limited to work on maximum voltage of 3.6 V [39], so the range of the sensor is compromised in this case.

6.2 Power Analysis

For approximating the expected battery life cycle of the system, current consumption of the device was measured, and consumed energy consumption was calculated for different states of the device operation. Testing procedure, recorded results, and battery life cycle estimations are described in this section.

6.2.1 Testing setup

Before starting with actual current measurements, Thingy 91 development kit needs to be prepared for measuring current consumption on nRF-9160 [40]. This is done by cutting the short on SB3 so that all the current is redirected to P1 terminals of the board. After this, an ampere meter, power profiler, or oscilloscope can be used to measure the current consumed by the nRF-9160. In this analysis, both ampere meter and power profiler were used to observe the current consumption trends of the device.

In terms of application software, the sleep state timer was reduced from 60 minutes to 5 minutes so that more data logs of multiple transmissions can be obtained in a short period of time.

6.2.2 Current measurements

Setup for noting current measurements during different states of the device through Agilent 34410A ammeter is shown in figure 26. More detailed result logs were obtained for analysis with the help of a DC Power Analyser. Keysight N6705C DC Power Analyser kit was used to power Thingy 91's nRF-9160 to obtain detailed current consumption logs. Thingy 91 was connected to channel 1 of the power profiler and logger was configured to record the results after every 1 ms. The testing setup is shown in figure 27.

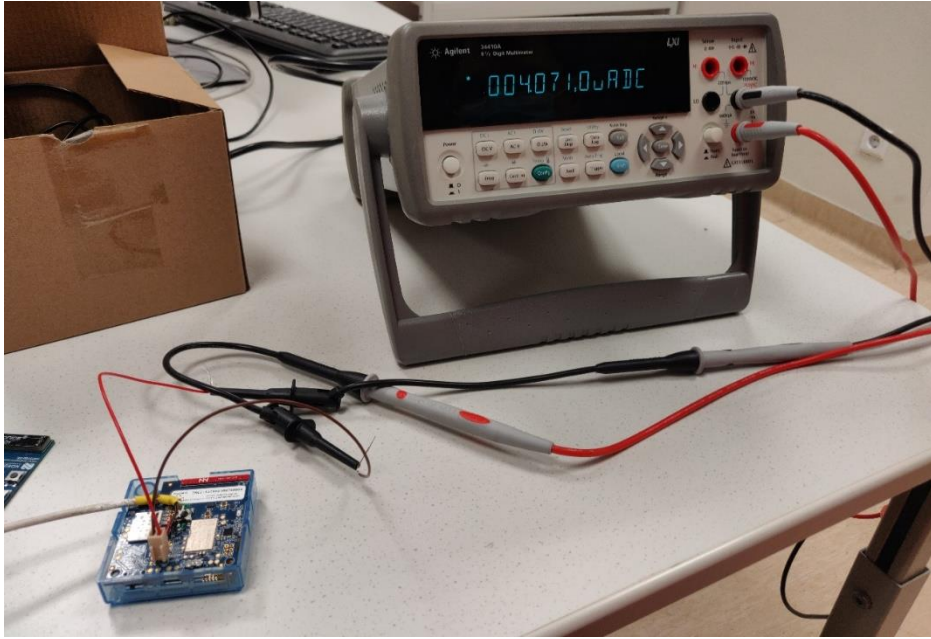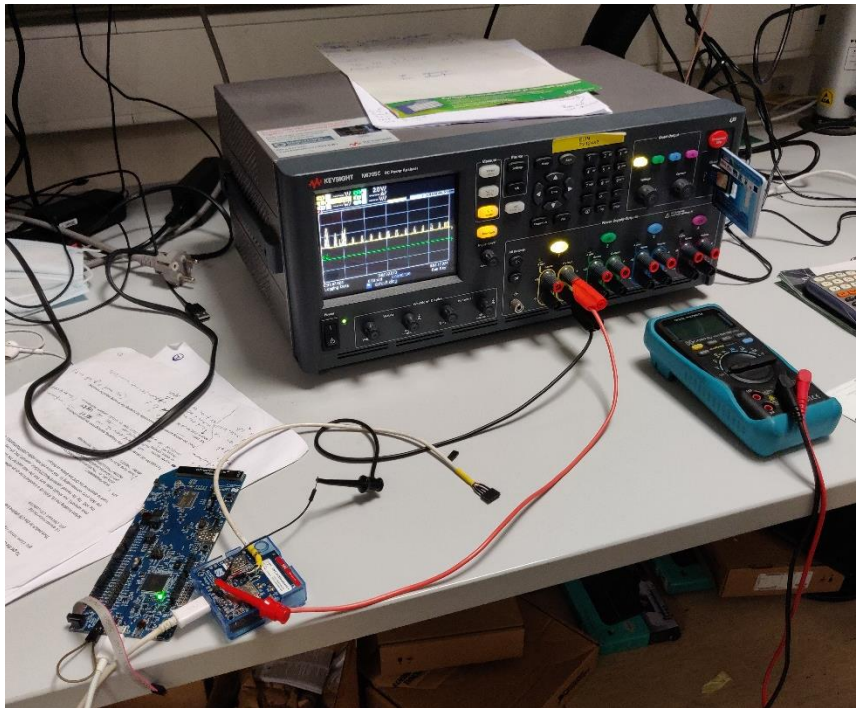Figure 26. Measuring Thingy 91 current using Agilent 34410A ammeter

Figure 27. Thingy 91 connected to Keysight N6705C DC Power Analyser kit

The current consumption log for the duration of 15 minutes is shown in figure 28. After the initial boot up and modem configuration, three data transmissions to the cloud can be observed in the form of spikes. In between each transmission is the current consumption during sleep mode.
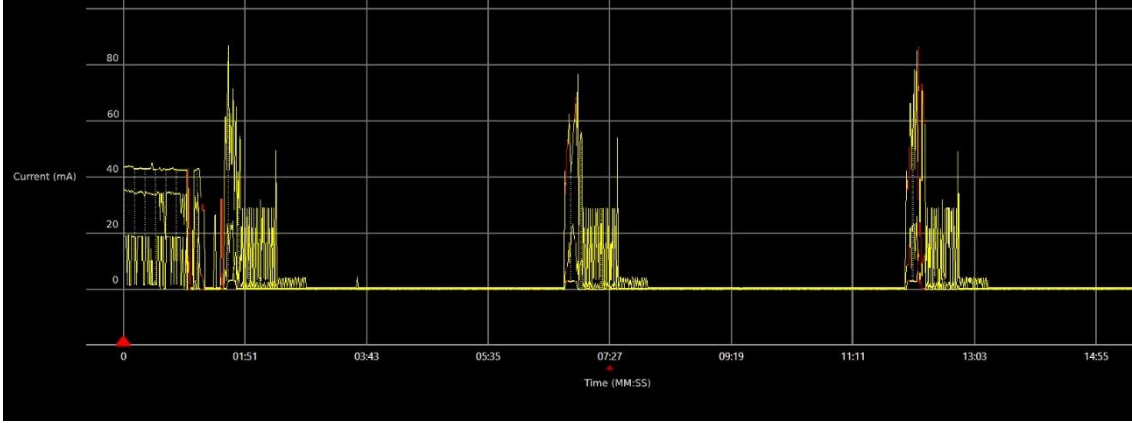
Figure 28. Current consumption of nRF-9160 for the duration of 15 minutes

This average current consumption during each state of the program's life cycle can be approximated by equation (7),

$$\text{Average Current} = I_{avg} = \frac{1}{T} \int I(t)\, dt \tag{7}$$

where 'I' is the value of current at time 't' and 'T' is the number of samples. In this case values were obtained from the power analyser kit. Also, current consumption per hour for each state has been calculated using equation (8),

$$Current\ consumption\ per\ hour = I_{avg} * T_h \tag{8}$$

where 'T_h' is the time duration in hour for which the device consumes 'I_{avg}'.

Initial bootup and modem configuration state:

This is the state when the device is turned ON from OFF state. In this state, application processor along with SIM and modem are turned on, modem is configured, and modem registers to the NB-IoT network. Since, PSM mode is requested from the network so this state is only triggered at the initial start-up of the device. This state can last from 1 – 2 minutes depending upon availability of the network. During the analysis, this state lasted for about 1 minute 13 seconds, and the current consumption during this state has been shown in figure 29. Average Current and Current Consumption per hour values for this state are summarized in table 8.
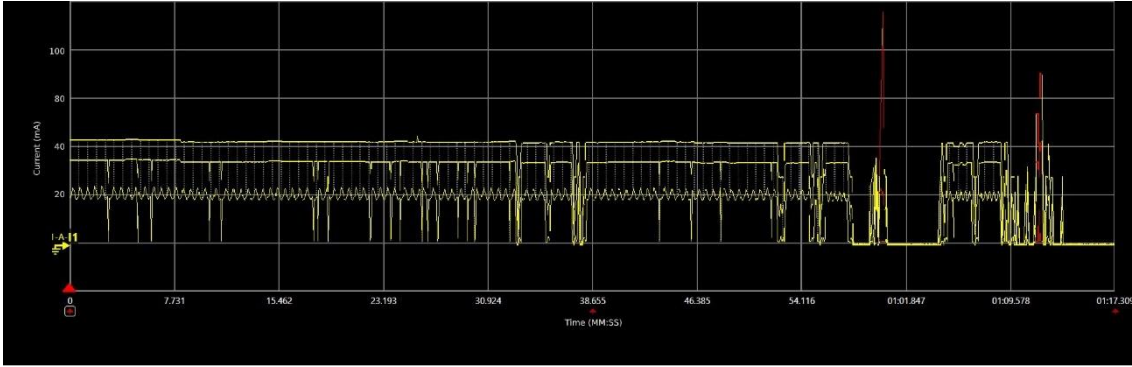
63

Figure 29. Current consumption during initial bootup and modem configuration state

Table 8. Current calculated values in initial bootup and modem configuration state

Time Elapsed	01 minute and 13 seconds
Average Current Consumption	28.7 mA
Current consumption per hour	581 µAh

Sensor data acquisition and communication state:

In this state, sensor is triggered, water level measurements are calculated, measurements are transmitted to the cloud, and SIM and modem are turned off. This is a recurring state which occurs after the specified sleep time interval which was 5 minutes for this analysis. This state lasts anywhere between 1 minute and 1 minute 30 seconds. The sensor data acquisition and cloud communication usually take very short time, but after that the cellular connection keeps the radio resource engaged for some time for downlink communication. Once the radio resources are released by the network the modem and SIM can be turned off and application can be put to sleep mode. This state lasted for around 1 minute and 15 seconds during the analysis and the corresponding current graph is shown in figure 30. Here the sensor data acquisition and communication part lasted for only 17 seconds i.e. from 6:45 to 07:02 in which most amount of current is drawn. Current consumption values calculated from this state are presented in table 9. It is to be noted that sensor current consumption is also included in these measurements and the current consumption for taking a distance measurement is 1.3 mA, as shown in table 10. This is considerably less than the rated power consumption of 8 mA or less from the sensor's datasheet [15].
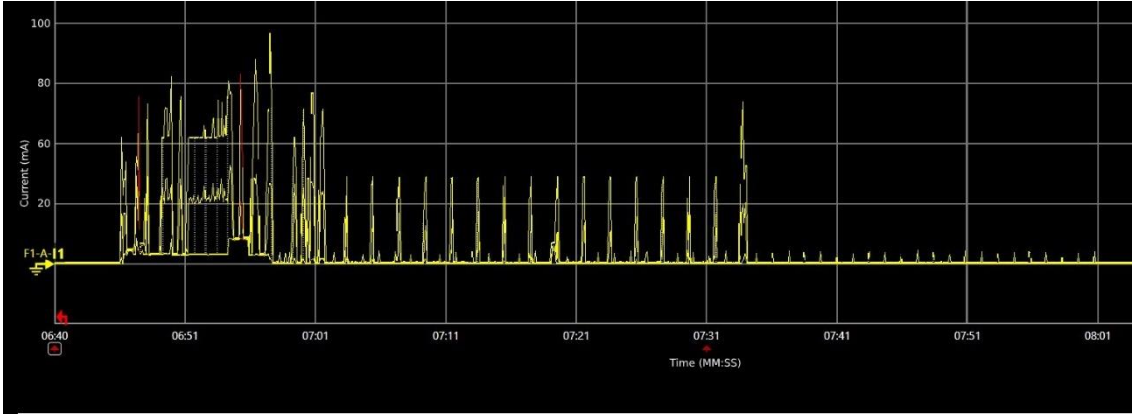
64

Figure 30. Current consumption during sensor data acquisition and communication state

Table 9. Current calculated values in sensor data acquisition and communication state

Time Elapsed (1 Transmission)	01 minute and 17 seconds
Average Current Consumption	3.02 mA
Current consumption per hour	64.5 µAh

Table 10. Current consumption of sensor for distance measurement

Current Consumption (1 measurement)	1.3 mA

Sleep state:

This is the most prevalent state of the device in which all the peripherals including modem and SIM are off and the application processor is in idle state. This state will be followed by the sensor data acquisition and transmission state after the sleep interval expires. The current consumption during this state is usually 4 – 5 µA visible as the straight lines between the transmission spikes in figure 31. The average current consumption came out to be 4.73 µA which is slightly higher than 4.5 µA which is the minimum achievable current on nRF-9160 in PSM mode as specified in the device's product specification [33]. Since this is the state in which the device will spend most of the time this makes the monitoring system extremely power efficient. Current consumption values estimated can be seen in table 11.
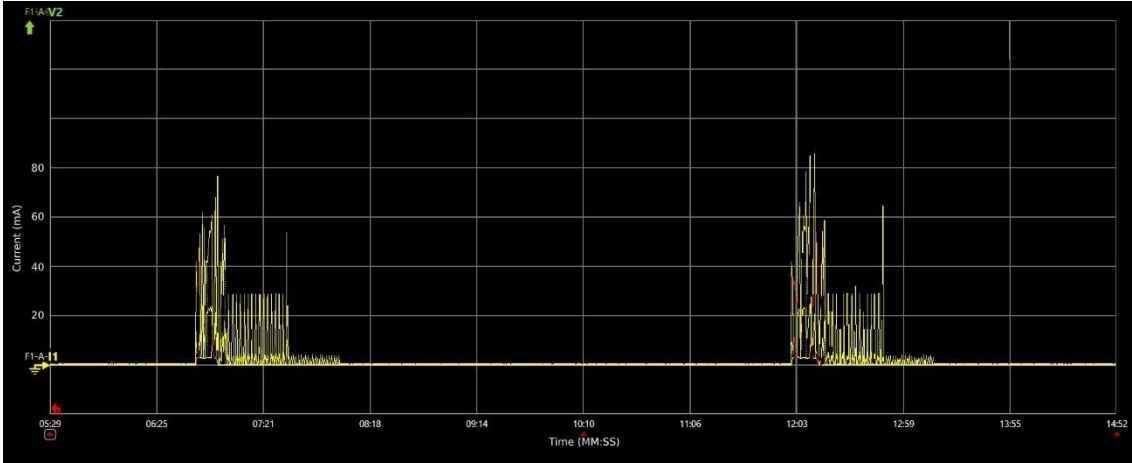
Figure 31. Current consumption during sleep state

Table 11. Current calculated values in sleep state

Time Elapsed	5 minutes
Average Current Consumption	4.738 µA
Current consumption per hour	348 nAh

6.2.3 Battery life cycle estimation

Now, that we have the current consumption measurements for different states of the device, battery life cycle of the device can be estimated in the normal operation i.e. with 60 minutes of sleep time between each transmission state. The battery life cycle of a device in hours can be estimated by using equation (9),

$$\text{Battery life cycle} = \frac{\text{Battery capacity}}{\text{Average current}} \qquad (9)$$

where 'Battery capacity' is the rated capacity of given battery in mAh and 'Average current' is the average current consumed by the device in the overall operation. For making the calculation simpler, the average current for the overall device operation is approximated using transmission state and sleep state as these are the recurring states in the overall operation. Initial bootup state current can be ignored because it is a one time expenditure at the initial start-up of the device. Average current can then be approximated using equation (10),

$$\text{Average current} = \frac{(I_{TX}*T_{TX})+(I_{SL}+T_{SL})}{T_{TX}+T_{SL}} \qquad (10)$$

66

where 'I_{TX}' is the average current in transmission state, 'I_{SL}' is the average current in sleep state, and 'T_{TX}' and 'T_{SL}' is the time spent in transmission state and sleep state respectively.

Considering a device operation of 1 hour with 1 minute 20 seconds of transmission duration and 58 minutes and 40 seconds of sleep duration the average current can be approximated from equation (9) at 69.2 µA. Pairing this with a 4000 mAh battery yields the overall battery life of approximately more than 57803 hours or 6.5 years from equation (8). These results are summarized in table 12.

Table 12. Battery lifecycle estimation values

Battery capacity	4000 mAh
Average current	69.2 µA
Estimated battery life	57803 hours or 6.5 years

6.3 Conclusion from prototype testing

From the sensor testing, it can be seen that after applying Error Correction measurements up to a distance of 1 m can be of ± 2 cm accuracy and after that the accuracy gets significantly worse i.e. ± 6 cm. In terms of precision, sensor values seem to hold up pretty well up to a distance of 1 m with variation of only ± 0.5 cm but after that the precision drops down to ± 4.5 cm. These poor measurement results can be because of low input voltage of 3.3 V to the sensor. Since, nRF-9160 GPIO pins cannot operate on voltage levels higher than 3.6 V this seems to be the limiting factor for this sensor's range. Range of the monitoring system can be improved by using a more power efficient ultrasonic sensor.

The detailed power analysis highlights the advantages of heavy power optimizations done for the application. It has been estimated that during the normal operation with one sensor measurement transmission every hour the average current consumed is approximately 69.2 µA. This results in approximately 6.5 years of battery life on a 4000 mAh battery which makes this solution extremely power efficient.

7 Conclusion and Future Work

This thesis covers various aspects of the design and development of a low powered IoT sensor system for underground drainage monitoring. This system is capable of making real time water level measurements which can be useful for implementation of a SUDS to prevent urban flooding events during extreme rain fall events. The developed prototype is capable of making water level measurements at a maximum distance of 1 m with ± 2 cm accuracy. These measurements are transmitted to AWS cloud with TSL encrypted MQTT via NB-IoT cellular network. The protype has been heavily optimised for low powered usage and can operate for more than 6.5 years on a 4000 mAh single cell Lithium battery with average current consumption of 69.2 µA.

More work is required for improving the range of the water level measurements by replacing the ultrasound sensor currently in use. The sensor has too short range for 3.3V operation and is rather sensitive to signal side reflections. TDK's ultra-low power chirp signal ultrasonic sensor seems to be a good choice as it can provide measurements of up to 5 m while consuming 15µA @ 1.8V during the measurements [41]. This is a new sensor with limited documentation for its application development so this couldn't be tested in this work due to significant development effort needed.

Another area for further developments is the comparison and analysis of manhole measurements data from different sensors in variable conditions. This will provide more details about the usage of different sensors for different applications in diverse use cases.

References

[1] P. Willems, 'Impacts of Climate Change on Rainfall Extremes and Urban Drainage Systems', *Water Intell. Online*, vol. 11, Aug. 2012, doi: 10.2166/9781780401263.

[2] L. Alfieri, L. Feyen, and G. Di Baldassarre, 'Increasing flood risk under climate change: a pan-European assessment of the benefits of four adaptation strategies', *Clim. Change*, vol. 136, no. 3–4, pp. 507–521, Jun. 2016, doi: 10.1007/s10584-016-1641-1.

[3] H. Malik, N. Kandler, M. M. Alam, I. Annus, Y. Le Moullec, and A. Kuusik, 'Evaluation of low power wide area network technologies for smart urban drainage systems', in *2018 IEEE International Conference on Environmental Engineering (EE)*, Mar. 2018, pp. 1–5, doi: 10.1109/EE1.2018.8385262.

[4] T. Beeneken *et al.*, 'Real time control (RTC) of urban drainage systems – A discussion of the additional efforts compared to conventionally operated systems', *Urban Water J.*, vol. 10, no. 5, pp. 293–299, Oct. 2013, doi: 10.1080/1573062X.2013.790980.

[5] L. Garcia, J. Barreiro-Gomez, E. Escobar, D. Tellez, N. Quijano, and C. Ocampo-Martinez, 'Modeling and real-time control of urban drainage systems: A review', *Adv. Water Resour.*, vol. 85, pp. 120–132, Nov. 2015, doi: 10.1016/j.advwatres.2015.08.007.

[6] 'Remote Monitoring of Rising Water Levels in Manholes', *Sensoneo*. https://sensoneo.com/product/manhole-monitoring-overflow-octopus-sensor/ (accessed Dec. 25, 2020).

[7] 'A dozen ways to measure fluid level and how they work - Level Measurement | Level Sensors | Level Transmitters | ABB'. https://new.abb.com/products/measurement-products/level/a-dozen-ways-to-measure-fluid-level (accessed Dec. 25, 2020).

[8] Engr. Ajayi Isaac Ayodele*1 & Engr. Dr. Iliya Tizhe Thuku2, 'A Review On Electrical Instrumentation Techniques Applied In Groundwater Level Determination', Sep. 2018, doi: 10.5281/ZENODO.1407686.

[9] K. Loizou and E. Koutroulis, 'Water level sensing: State of the art review and performance evaluation of a low-cost measurement system', *Measurement*, vol. 89, pp. 204–214, Jul. 2016, doi: 10.1016/j.measurement.2016.04.019.

[10] A. Qurthobi, R. F. Iskandar, A. Krisnatal, and Weldzikarvina, 'Design of capacitive sensor for water level measurement', *J. Phys. Conf. Ser.*, vol. 776, p. 012118, Nov. 2016, doi: 10.1088/1742-6596/776/1/012118.

[11] F. Ehiagwina, L. Afolabi, O. Kehinde, A. Olaoye, and A. Jibola, 'Overview of Liquid Level Detection Technologies with Performance Characteristics Assessment and Energy Cost Saving for Household Water Pumps', *World Wide J. Multidiscip. Res. Dev.*, vol. 3, pp. 287–294, Sep. 2017.

[12] S. Jatmiko, A. Mutiara, and M. Indriati, 'Prototype of water level detection system with wireless', *J. Theor. Appl. Inf. Technol.*, vol. 37, pp. 52–59, 2012.

[13] K. N. D. Grassi, R. C. S. Freire, and J. M. M. Villanueva, 'Evaluation of wavelet analysis performance in multiphase level measurement using ultrasonic sensors', in *2014 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*, May 2014, pp. 756–760, doi: 10.1109/I2MTC.2014.6860844.

[14] A. K. Sahoo and S. K. Udgata, 'A Novel ANN-Based Adaptive Ultrasonic Measurement System for Accurate Water Level Monitoring', *IEEE Trans. Instrum. Meas.*, vol. 69, no. 6, pp. 3359–3369, Jun. 2020, doi: 10.1109/TIM.2019.2939932.

[15] 'JSN-SR04T-Datasheet.pdf'. Accessed: Dec. 24, 2020. [Online]. Available: https://www.makerguides.com/wp-content/uploads/2019/02/JSN-SR04T-Datasheet.pdf.

[16] 'DS-000379-CH201-v1.1.pdf'. Accessed: Jan. 02, 2021. [Online]. Available: http://3cfeqx1hf82y3xcoull08ihx-wpengine.netdna-ssl.com/wp-content/uploads/2020/09/DS-000379-CH201-v1.1.pdf.

[17] Joongsuk Park and Cam Nguyen, 'A Ka-band stepped-frequency radar sensor for surface and subsurface sensing', in *2007 IEEE Antennas and Propagation Society International Symposium*, Jun. 2007, pp. 4921–4924, doi: 10.1109/APS.2007.4396648.

[18] 'SEN0192_Web.pdf'. Accessed: Dec. 24, 2020. [Online]. Available: https://media.digikey.com/pdf/Data%20Sheets/DFRobot%20PDFs/SEN0192_Web.pdf.

[19] 'US $36.95 20% OFF|High Precision Laser Measure Sensor Module Laser Distance Sensor For Arduino Infrared Ranging Sensor Laser Range Finder Module|Instrument Parts & Accessories| - AliExpress', *aliexpress.com*. //www.aliexpress.com/item/33035807395.html?src=ibdm_d03p0558e02r02&sk=&aff_platform=&aff_trace_key=&af=&cv=&cn=&dp= (accessed Dec. 25, 2020).

[20] M. Centenaro, L. Vangelista, A. Zanella, and M. Zorzi, 'Long-range communications in unlicensed bands: the rising stars in the IoT and smart city scenarios', *IEEE Wirel. Commun.*, vol. 23, no. 5, pp. 60–67, Oct. 2016, doi: 10.1109/MWC.2016.7721743.

[21] D. Patel and M. Won, 'Experimental Study on Low Power Wide Area Networks (LPWAN) for Mobile Internet of Things', in *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*, Jun. 2017, pp. 1–5, doi: 10.1109/VTCSpring.2017.8108501.

[22] K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, 'A comparative study of LPWAN technologies for large-scale IoT deployment', *ICT Express*, vol. 5, no. 1, pp. 1–7, Mar. 2019, doi: 10.1016/j.icte.2017.12.005.

[23] B. Reynders, W. Meert, and S. Pollin, 'Range and coexistence analysis of long range unlicensed communication', in *2016 23rd International Conference on Telecommunications (ICT)*, May 2016, pp. 1–6, doi: 10.1109/ICT.2016.7500415.

[24] LoRa Alliance Technical Marketing Workgroup, 'A technical overview of LoRa and LoRaWAN'. LoRa Alliance, Nov. 2015, [Online]. Available: https://lora-alliance.org/sites/default/files/2018-04/what-is-lorawan.pdf.

[25] NB-IoT forum and Internet of Things programme, 'NB-IoT Deployment Guide to Basic Feature Set Requirements'. GSMA, Apr. 2018, [Online]. Available: https://www.gsma.com/iot/wp-content/uploads/2018/04/NB-IoT_Deployment_Guide_v2_5Apr2018.pdf.

[26] J. Bardyn, T. Melly, O. Seller, and N. Sornin, 'IoT: The era of LPWAN is starting now', in *ESSCIRC Conference 2016: 42nd European Solid-State Circuits Conference*, Sep. 2016, pp. 25–30, doi: 10.1109/ESSCIRC.2016.7598235.

[27] 'NB-Fi specification'. https://waviot.com/technology/nb-fi-specification/ (accessed Dec. 04, 2020).

[28] 'Avnet Silica NB-IoT Sensor Shield | Avnet Silica'. https://www.avnet.com/wps/portal/silica/products/new-products/npi/2018/avnet-nb-iot-shield-sensor/ (accessed Jan. 02, 2021).

[29] 'MultiTech Conduit® Programmable Gateway for the Internet of Things EU868 for Europe', p. 4.

[30] 'Mobile Signal Strength Recommendations - Teltonika Networks Wiki'. https://wiki.teltonika-networks.com/view/Mobile_Signal_Strength_Recommendations (accessed Jan. 02, 2021).

[31] N. Poddar, S. Z. Khan, J. Mass, and S. N. Srirama, 'Coverage Analysis of NB-IoT and Sigfox: Two Estonian University Campuses as a Case Study', in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, Jun. 2020, pp. 1491–1497, doi: 10.1109/IWCMC48107.2020.9148570.

[32] S. Z. Khan, H. Malik, J. L. R. Sarmiento, M. M. Alam, and Y. L. Moullec, 'DORM: Narrowband IoT Development Platform and Indoor Deployment Coverage Analysis', *Procedia Comput. Sci.*, vol. 151, pp. 1084–1091, 2019, doi: https://doi.org/10.1016/j.procs.2019.04.154.

[33] 'nRF9160_PS_v2.0.pdf'. Accessed: Dec. 15, 2020. [Online]. Available: https://infocenter.nordicsemi.com/pdf/nRF9160_PS_v2.0.pdf.

[34] 'LoRa — LoRa documentation'. https://lora.readthedocs.io/en/latest/ (accessed Jan. 03, 2021).

[35] 'Quectel_BG96_LPWA_Specification_V1.8.pdf'. Accessed: Dec. 13, 2020. [Online]. Available: https://www.quectel.com/UploadFile/Product/Quectel_BG96_LPWA_Specification_V1.8.pdf.

[36] 'Arduino MKR NB 1500'. https://store.arduino.cc/arduino-mkr-nb-1500-1413 (accessed Jan. 03, 2021).

[37] 'SARA-R4_ProductSummary_(UBX-19049143).pdf'. https://www.u-blox.com/en/docs/UBX-19049143 (accessed Dec. 15, 2020).

[38] 'Understanding the AWS IoT Security Model', *Amazon Web Services*, May 11, 2017. https://aws.amazon.com/blogs/iot/understanding-the-aws-iot-security-model/ (accessed Dec. 29, 2020).

[39] 'Nordic Semiconductor Infocenter'. https://infocenter.nordicsemi.com/index.jsp?topic=%2Fps_nrf9160%2Frecommended_op_conditions.html (accessed Dec. 31, 2020).

[40] 'Thingy91_UG_v1.4.pdf'. Accessed: Jan. 04, 2021. [Online]. Available: https://infocenter.nordicsemi.com/pdf/Thingy91_UG_v1.4.pdf.

[41] 'CH201 | TDK'. https://invensense.tdk.com/products/ch201/ (accessed Jan. 04, 2021).

Appendix 1 – Non-exclusive licence for reproduction and publication of a graduation thesis[1]

I Haris Ahmed

1. Grant Tallinn University of Technology free licence (non-exclusive licence) for my thesis "Low Power IoT Sensor System for Underground Drainage Monitoring", supervised by Alar Kuusik

    1.1. to be reproduced for the purposes of preservation and electronic publication of the graduation thesis, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright;

    1.2. to be published via the web of Tallinn University of Technology, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright.

2. I am aware that the author also retains the rights specified in clause 1 of the non-exclusive licence.

3. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

04.01.2021

---

1 The non-exclusive licence is not valid during the validity of access restriction indicated in the student's application for restriction on access to the graduation thesis that has been signed by the school's dean, except in case of the university's right to reproduce the thesis for preservation purposes only. If a graduation thesis is based on the joint creative activity of two or more persons and the co-author(s) has/have not granted, by the set deadline, the student defending his/her graduation thesis consent to reproduce and publish the graduation thesis in compliance with clauses 1.1 and 1.2 of the non-exclusive licence, the non-exclusive license shall not be valid for the period.