Low-power IoT Structural Health Monitoring System with Optimized Data Acquisition

by

Andi Liu

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in
Software Engineering and Intelligent Systems

Department of Electrical and Computer Engineering
University of Alberta

Abstract

For large-scale metallic structures, such as aircraft, ships, and pipelines, their failure could be catastrophic. A structural health monitoring system (SHM) to detect failures in advance is essential. The Internet-of-Things (IoT) is a key technology applied for remote structural health monitoring. A challenging problem that arises in this domain is power consumption. Currently, most IoT systems are battery-powered and only have limited power capacity. Nevertheless, frequently replacing the battery is undesirable because some IoTs may be installed in a remote area. The objective of this research is to design and develop an IoT-based structural health monitoring system that includes an ultra-low-power miniaturized sensing node integrated with a three-dimensional (3D) stress and strain sensor. The system can monitor 3D stress and strain values and upload them to the cloud.

The developed IoT system consisted of a sensing node, a router, and a gateway, while the sensing node was composed of an acquisition unit and a 3D Micro-Electro-Mechanical System (MEMS) sensor to monitor 3D stress and strain of a structure. The acquired data was transmitted from a node to a router, then sent to the gateway, and finally uploaded to the cloud, which could be viewed remotely. Bluetooth Low Energy (BLE) was the major communication between node – router and router – gateway. Wi-Fi was used by the gateway to upload data to the cloud.

The acquisition unit contained a low-power microcontroller with a built-in BLE wireless communication function, a high precision 16-bit analog-to-digital converter (ADC) and a 32Mbytes flash memory. It interfaced with a compact 3D MEMS strain-gauge typed sensor (7mm x 7mm) developed by a team in mechanical engineering at the

University of Alberta. The sensor has high accuracy and stability, low power consumption, and small size. In order to optimize the power consumption of the sensing node, intelligent dynamic sampling rate, low power sleep mode and optimized transmission power level were implemented into the design. The dynamic sampling rate was based on the thresholding concept. When a number of consecutive detected stress/strains exceeded a pre-set threshold range, the sampling rate was changed from 5Hz to 20 Hz. Data sampling was changed from every 200ms to 50ms. The acquisition unit was put into sleep mode while not collecting or transmitting data to decrease the power consumption. A transmit power optimization algorithm was implemented to find the optimal transmit power level to conserve power in transmission mode.

Experiments were conducted to determine the sensitivity and crosstalk of the ADC, the accuracy of the sensing node, the communication distance range between sensing node-router-gateway, the current consumption of the sensing node at different operating modes. Experiments were also conducted to validate the entire system. The results showed good sensitivity (0.51Ω) and accuracy (0.58±0.34Ω) of the sensing node, allowing detecting a sub-ohm resistance change on a strain-gauge type sensor. The maximum transmission distance was over 35m. The physical dimensions and power consumption of the sensing node were 35mm x 38mm x 10 mm, 24mW in active mode, and 9.4mW in sleep mode, respectively. The experimental results also demonstrated that the average power consumption of the sensing node was 14.0mW per hour if an average of 10 triggered high-frequency samplings occurred over a day. The system could last for more than seven days with a 1200mAh battery.

Preface

Portions of the material in this thesis have been submitted in the paper: A. Liu, W. Moussa and E. Lou, “Low-power IoT Structural Health Monitoring Platform with Optimized Data Acquisition,” IEEE Transactions on Instrumentation and Measurement, 2021 (Submitted). I designed the hardware and software design of the system. Prototypes of the sensor board and acquisition board were built with the assistance of Dr. Moussa and Dr. Lou. After building the prototypes, I conducted experiments for functional validation and discussed the results. I composed the manuscript with the assistance of Dr. Lou.

Acknowledgements

First and foremost, I express my gratitude to my supervisor – Dr. Edmond Lou, for his valuable advice, support, and patience at every stage of the research project. Every step of the way, he kept me on track. Without his understanding and help in the past few years, it would be impossible for me to complete my study.

I would like to thank Syncrude Canada for providing the financial support as well as the initial inspiration for this project.

I would also like to thank Jonny Song and Brett Baker for their technical advice in programming. Discussions with them gave me so much help.

Finally, I wish to thank my family and friends for their moral support and patience. Thanks for their supports and always being there for me.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

3D: Three Dimensional

ADC: Analog-to-Digital

API: Application Programming Interface

BLE: Bluetooth Low Energy

CPU: Central Processing Unit

CS: Chip Select

FP: Fabry–Perot interferometric

FBG: Fibre Bragg-Grating

FPGA: Field Programmable Gate Array

FRP: Fiber-Reinforced Polymer

FSR: Full-Scale Range

GAP: Generic Access Profile

GATT: Generic Attribute Profile

GF: Gauge Factor

GPIO: General-purpose input/output pins

IC: Integrated Circuit

IDE: Integrated Development Environment

INL: Integral Nonlinearity

IoT: Internet of Things

ISM: Industrial, Scientific, and Medical band

LSB: Least Significant Bit

MCU: Microcontroller Unit

MEMS: Micro-Electro-Mechanical System

MTU: Maximum Transmission Unit

MUX: Multiplexer

NVM: Non-Volatile Memory

OSR: Oversampling Ratio

OQPSK: Offset Quadrature Phase-Shift Keying

PCB: Printed Circuit Board

PCM: Polymer Composite Material

PHY: Physical Layer

PVDF: Polymer Piezoelectric

PZT: Piezoelectric

RAM: Random Access Memory

SPI: Serial Peripheral Interface

SHF: Super High Frequency

SHM: Structural Health Monitoring

SoC: System-on-chip

S/N: Signal-to-noise

TX: Transmit

UHF: Ultra High Frequency

VSAT: very small aperture terminal

Wi-Fi: Wireless Fidelity

WSN: Wireless sensor network

Chapter 1. Introduction

Structural health monitoring (SHM) can be described as the process of monitoring engineering structures such as bridges and buildings and detecting damages of them. These damages refer to changes in the infrastructure's material or geometric properties, which may affect its current or future performance.

SHM can be used to investigate the structures' integrity. Due to this nature, SHM can be employed for rapid condition screening and generating reliable information regarding the integrity of the structure [1] after an extreme event (e.g., earthquake). Based on this generated information, experts can evaluate the residual strength of the structure and determine whether the reoccupation of a specific building is feasible. Therefore, SHM plays an essential role in securing public safety.

Moreover, SHM can be used as a precaution to prevent unexpected structural failures. Multiple minor damages may jointly and continuously deteriorate the structure's condition until the structure is no longer acceptable for the users. Continuous SHM makes it feasible to take measures before any loss occurs. Regarding large-scale metallic structures, such as aircraft, ships, and pipelines, their failure could be catastrophic. Their failure not only puts human life in danger but also results in significant financial losses. The leakage of hazardous materials may endanger the environment as well. On December 11, 2012, an internationally underground natural gas pipeline in West Virginia, USA, spilled and burnt with a diffusion area of approximately 8.88×10^4 m^2. This accident cost 8.68 million dollars [2]. Another example would be the leakage in the Donghuang oil pipeline in Qingdao, Shandong Province, China. The crude oil spilled into a municipal drainage culvert, where it accumulated with gas and eventually ignited by a spark and caused a catastrophic explosion. The incident claimed the lives of 62 individuals and wounded 136 others, resulting in a direct economic loss of approximately 751 million Chinese yuan [2]. In conclusion, a structural health monitoring system to detect failures in advance is essential. Such a system can protect human life, substantially reduce maintenance costs, and prevent catastrophic structural failures.

A common mode of failure of large metallic structures is cracking. It is mainly caused by long-term damage accumulation due to the inevitable aging of the structure and degradation resulting from the operational environment. A straightforward approach to avoid the accumulation of minor cracks is to replace the fractured portion. However, for a large and complicated structure, this solution is costly and challenging to manage. An alternative approach is to apply repaired patches over the crack. However, these bonded patches lose their elastic modulus slowly and suffer from high shear stress. Eventually, a patch debonding may occur. SHM can be used to monitor the stress and strain applied to the metallic structure, consequently detecting potential debonding of the adhesively bonded repair [3]. This research was initiated by Syncrude Canada to monitor 3D stress and strain on an oil-sand structure; the design must meet the specifications provided by Syncrude Canada.

In general, SHM requires continuous observation using periodically sampled response measurements from a series of sensors. Data captured from these sensors, including the applied stress and strain or an environmental condition such as temperature, humidity, and vibrations, can determine the structures' condition. Among many different types of sensors, stress and strain sensors are commonly used to detect the stress-concentrated regions that are prone to induce structural damage [4]. A fair amount of research has been conducted for data interpretation [5-9] to detect damage and determine damage locations.

Due to the large size of an engineering structure and the limited sensing region of a single sensor, multiple sensors are usually necessary to monitor the whole structure. Although traditional wired SHM systems have been developed and used for many years [1], they have a significant drawback, which is the costly and time-consuming installation process as it requires intensive cabling works. As the technology becomes more advanced, employing the Internet of Things (IoT) for SHM is a valuable alternative. The definition of an IoT is a system of interrelated devices and the ability to transmit data wirelessly with no human-to-human or human-to-computer interaction [10]. The cloud stores large volumes of data. IoT makes it feasible to integrate SHM with the Internet to track data regardless of time and location. An IoT SHM avoids the cost of massive cabling and cable installation. The communication protocol is the major component affecting the data transmission, which includes the rules, syntax, semantics, and synchronization of communication and covers the authentication, error detection, and recovery methods. Many wireless communication protocols can be utilized in IoTs, which include Wi-Fi, ZigBee, Bluetooth, and Bluetooth Low Energy (BLE). Currently, most studies on IoT-based SHM platforms utilize Wi-Fi and ZigBee.

Since most IoT systems aim to be battery-powered, a challenge that arises in IoT-based SHM systems is the power consumption and limited power capacity. With a fixed battery capacity, a sensor node with lower current consumption can run for a longer period, reducing the need for battery replacement. Frequently replacing the battery is undesirable because some IoTs may be installed in a remote area. In other words, a system with low power consumption is required. Based on the review of the existing platforms, the power consumption of the IoT-based SHM systems is still one of the challenges.

1.1 Objectives

The objectives of this research are:

- To design and develop an IoT-based SHM system that can monitor three-dimensional (3D) stress and strain values and transmit them wirelessly,
- To design and develop a miniaturized low-power IoT acquisition platform that is integrated with a 3D stress-strain Microelectromechanical Systems (MEMS) sensor to have high sensitivity and detect small changes on that sensor,
- To measure and transmit data wirelessly and data can be downloaded from Google Drive for processing,
- To design and implement an intelligent data acquisition approach so that low power consumption can be achieved.

1.2 Thesis Organization

This thesis consists of six chapters. It starts with an introduction of structural health monitoring and the IoT in Chapter 1. The objectives of this research are also listed.

Chapter 2 provides stress-strain sensor background and a literature review about different types of stress-strain sensors. It also includes details about several different wireless communication protocols, such as Wi-Fi, ZigBee, Bluetooth, and Bluetooth Low Energy (BLE), and a literature review of SHM systems that integrate them.

Chapter 3 discusses the hardware design details, the rationale of the components' selection of the sensing node and router, and the selection of the gateway to form an IoT system.

Chapter 4 discusses the software designs of the system, including the firmware design of the acquisition board, router, and gateway. Details about an Android user interface application to interface with the node and router platforms for function validation are also included.

Chapter 5 describes the printed circuit board (PCB) design of the acquisition board, the prototypes of the IoT system. It also reports different experiments and results to validate the accuracy, current consumption, and communication range of the individual platform and the entire integrated system.

Chapter 6 summarizes the thesis work, concluding remarks about the research, and then some future recommendations.

Chapter 2. Literature Review

2.1 Overview

Generally, a structural health monitoring system consists of three components to provide and report the status of the structure: a) sensor, b) data acquisition system, and c) data processing and analysis. This chapter provides a literature review on both the stress-strain sensors and data acquisition systems. Section 2.2 presents a few common types of sensors that have been used for structural health monitoring. Section 2.3 discusses the data acquisition systems implementing different communication protocols. The pros and cons of the most common data communication protocols are also described.

2.2 Stress and Strain Sensors

Stress is a physical quantity that describes the force applied per unit area; strain is a measure of deformation in terms of relative displacement of particles in the body that excludes rigid-body motions. When forces are applied to an object in the direction that causes its elongation, it is called tensile stress. On the contrary, when forces cause compression of an object, it is called compressive stress. If a deformation exists, a strain may arise, and the quantity of it describes the deformation. Equation (2-1) describes the relationship between stress and strain.

$$Stress = Elastic\ modulus \times Strain \tag{2-1}$$

Many different types of stress and strain sensors have been developed to monitor the health of the structure. The following three sections describe the commonly used stress and sensors that have been reported in the literature, including a) optical fiber sensors (Section 2.2.1), b) piezoelectric sensors (Section 2.2.2), and c) strain gauge type sensors (Section 2.2.3).

2.2.1 Optical Fiber Sensors

Optical fibers are long, flexible, transparent fibers made by pure glass or plastic with a diameter of about human hair (~ 0.25 mm). They are arranged in bundles and can be used to transmit lights and sounds between the ends of the fiber. For the structural health monitoring application, lights are usually transmitted in the fiber. The applied stress and strain can be determined by measuring the change in the transmitted light intensity, polarization, phase, wavelength, or time of light in the fiber. Compared with other sensors, optical fiber sensors are small in physical dimensions and do not require electrical power at remote locations. Fabry–Perot (FP) interferometric sensor and Fibre Bragg-Grating (FBG) sensor are the two common optical fiber sensors [11].

2.2.1.1 Fabry–Perot Interferometric Sensor

Figure 2-1 shows an extrinsic Fabry-Perot interferometer sensor, which is composed of a capillary silica tube containing two cleaved optical fibers facing each other with an air gap of a few micrometers between them. The incident light beam traveling along the fiber is reflected at the interfaces of the glass-to-air and air-to-glass and recombined in the fiber core, causing interference at the output. By analyzing the output signals, the changes in the fiber spacing can be reconstructed using coherent or low-coherent techniques. Then, the strain applied onto the fiber can be obtained since it corresponds to the gap change while the strain sensitivity depends on both the FP cavity length and the cavity shape. Liu et al. [12] studied the fiber devices with three different cavity shapes: elliptic, D- and crescent-shaped FP sensor cavities, as shown in Figure 2-2, and demonstrated the crescent-shaped FP cavity for ultra-sensitive strain measurement. The test results showed that sensitivity of 9.67pm/με was achieved for a cavity length of 9μm. However, the FP sensor has a restriction that only one sensor can be integrated per fiber.

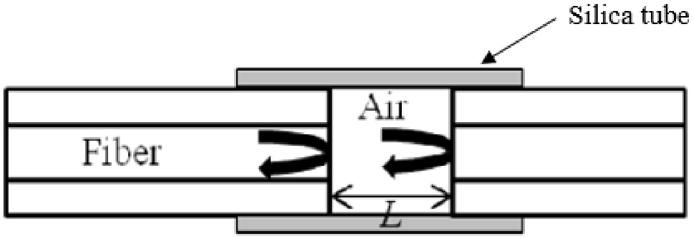


Figure 2-1 Structure of a Fabry-Perot interferometric sensor [13]

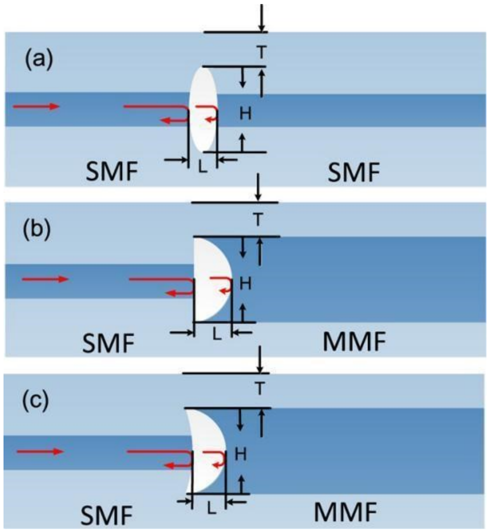


Figure 2-2 (**a**) Elliptic FP cavity; (**b**) D-shaped FP cavity; (**c**) crescent-shaped FP cavity [12]

2.2.1.2 Fibre Bragg-Grating Sensor

The refractive index is a physical property that determines how much the path of light is bent or refracted when entering a material. Fiber Bragg grating (FBG) sensor is a sensor whose fiber core has a periodic change of the refractive index, n_{eff}, causing a reflection of light with the Bragg wavelength λ_B, as shown in Figure 2-3. FBG strain sensors measure stress and strain based on the change in wavelength of the reflected light beam when exposed to structural deformation. When a broadband light source is injected into the optical fiber with grating, a light source with a specific wavelength corresponding to the grating is reflected. All other light sources with different frequencies pass through the grating undisturbed. The specific wavelength of the reflected light, which is called the Bragg wavelength $\lambda_{B,}$ can be calculated as follows:

$$\lambda_B = 2n_{eff}\Lambda \tag{2-2}$$

Where n_{eff} is the effective refractive index of the grating in the fiber core, and Λ is the grating period.

When the optical fiber is subjected to a uniaxial mechanical strain at a constant temperature, the relationship between mechanical strain and the change of the Bragg wavelength can be derived as follows:

$$\frac{\Delta\lambda_B}{\lambda_{B0}} = (1 - p_{eff})\varepsilon_{mech} \tag{2-3}$$

Where $\Delta\lambda_B$ is the change of Bragg wavelength, λ_{B0} is the unstrained Bragg wavelength, P_{eff} is the first order strain-optic coefficient, and ε_{mech} is the mechanical strain applied to the fiber.

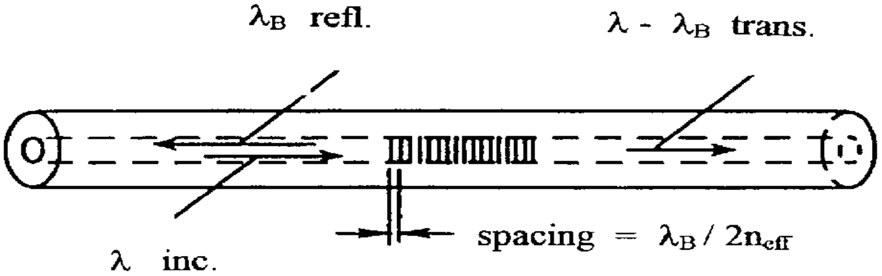

λ_B refl.
λ - λ_B trans.
λ inc.
spacing = $\lambda_B / 2n_{eff}$

Figure 2-3 Fiber Bragg Grating sensor [14]

However, when the temperature is not constant, it affects the material properties of the fiber optic, which then affects the measurement of strain. Hence, FBG-based sensors are simultaneously sensitive to both strain and temperature. Kumar et al. [15] designed a temperature-insensitive strain measurement sensor based on two fiber Bragg grating for the structural health monitoring system to compensate for the temperature effects. The first FBG sensor only acted as a temperature sensor, while the other one was used for both temperature and strain measurements. Both FBGs were located at the same sensing point with a 2nm separation of central wavelength to avoid signal overlapping. By subtracting the wavelength shift of the first FBG due to temperature in the second FBG due to both strain and temperature, the wavelength change corresponding to the applied strain could be obtained. However, when the FBG sensors with a bandwidth of 0.44nm were used, the test results showed that the wavelength shifts observed at the Optical Spectrum Analyzer overlapped with each other and could not extract the wavelength shift of each sensor. When both sensors operated at 0.12nm bandwidth, there was a negligible chance of overlapping. A 0.05nm wavelength shift was observed for 50µstrain, and a 0.14nm wavelength shift was observed for a 10°C increase in temperature.

Apart from the temperature compensation problem, packaging techniques for bare FBG sensors are also of significance. A suitable packaging can protect the inside optical fiber and is less likely to be damaged by external infringement, increasing service life. Zhi Zhou et al. [16] developed an optical fiber Bragg grating strain sensor with a package made of an advanced composite material, fiber-reinforced polymer (FRP). The durability was significantly increased due to the FPR materials without sacrificing the sensing performance of the bare FBG sensor. Compared with metal packaged optical fiber strain sensors, FRP-packaged strain sensors showed superior corrosion resistance making them suitable for long-term structural health monitoring. These FRP-packaged FBG strain sensors have been successfully applied in the SHM system of Aizhai Bridge.

Murukeshan et al. [17] demonstrated the use of embedded FBG sensors to evaluate strains in composite specimens. Intracore Bragg grating fibers of different wavelengths were embedded within 10-layered carbon-fiber-reinforced specimens and glass-fiber-reinforced specimens. The result showed that the force and the Bragg-reflected wavelengths exhibit highly linear behavior under the same temperature. Bragg wavelength of a specimen under three-point bending for several temperatures and forces were tested to eliminate the effect of temperature. A linear relationship was discovered between the wavelength under the same force and different temperatures, and an equation was derived to compute the wavelength. However, their experiments only tested on a few points. The slope of the strain-temperature curve was an approximation that might not be accurate for a wide range of temperatures. Matveenko et al. [18] studied the use of FBG sensors embedded into polymer composite material (PCM). The strain was measured under compression and tension and compared with the results of numerical modeling based on the finite element method. The differences between the experimental and numerical results in the tensile and compression zones were less than 7.5% and 6%, respectively. The numerical results also demonstrated that the sensing performance of the FBG sensor was not affected by the embedment.

2.2.1.3 Summary

Optical fiber sensors can be used as strain sensors for structural health monitoring and sometimes are preferred because of their small size, electromagnetic interference immunity, and no electrical power required at the remote location. Fabry-Perot sensor is a type of interferometric sensor that can compute strain accurately due to its high-resolution quality. However, only one sensor can be used per fiber. On the contrary, multiple FBG sensors can be engraved on the same fiber at different positions because of multiplexing. Moreover, it is capable of measuring non-uniform strain field measurement. It is reliable for long-term measurements since it is frequency coded without drifting by aging and can be embedded into composite materials to measure strain under both tension and compression. However, employing Fiber Bragg Grating Sensors for structural health monitoring in composite materials [19] has certain obstacles. One of them is the connection interface between the embedded sensors and the surroundings, which is essential for the industrialization of the process. Furthermore, when compared to other sensors, optical fiber sensors are more difficult to install and may need material embedment, which is not practical for existing structures. It is more expensive than other types of sensors and cannot be mass-produced like micro-electro-mechanical system sensors.

2.2.2 Piezoelectric Sensors

Other than the optical fiber sensor, the piezoelectric (PZT) sensor utilizes the piezoelectric effect to measure stress and strain. The piezoelectric effect is the characteristics of certain materials to generate an electric charge in response to applied mechanical stress. When mechanical stress is applied to a piezoelectric material, the asymmetrical lattice of molecules distorts, and a reorientation of electric charges within the material occurs, causing a relative charge displacement in the material. The charge displacement induces surface charges on the material of opposite polarity between the two sides. As shown in Figure 2-4, the induced surface charges are opposite when a tensile or compressive force is applied to the material. By implanting electrodes into the top and bottom surface of the material, these surface charges can be measured as an output voltage.

Hence, the changes in pressure, strain, or force can be measured by converting them to an electrical charge. Equations (2-4) and (2-5) describe piezoelectricity.

$$D_m = d_{m\beta}\sigma_\beta + \varepsilon^{\sigma}_{mk}E_k \qquad (2\text{-}4)$$

$$S_\alpha = s^{E}_{\alpha\beta}\sigma_\beta + d_{\beta m}E_m \qquad (2\text{-}5)$$

Where the indexes α, β can be any value from 1 to 6 and m, k can be 1, 2, or 3, referring to different directions in Figure 2-5. D is the electric displacement (C/m^2), d is the piezoelectric constant (m/V), σ is the stress (N/m^2), ε is the dielectric permittivity (F/m), E is the electric field (V/m), S is the strain (m/m), and s is the compliance coefficient (m^2/N). For instance, the coefficient d$_{31}$ is the induced strain in direction 1 per unit electric field applied in direction 3; d$_{15}$ is the induced shear strain about direction 2 per unit electric field applied in direction 1. The positive polarization (P) is the same as the positive z-direction.

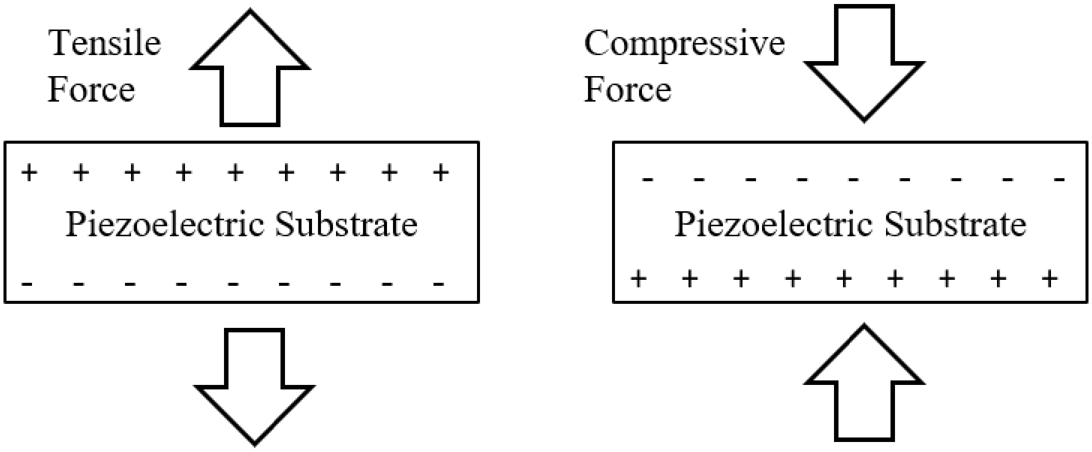


Figure 2-4 Mechanical tension or compression on a piezoelectric induces a current depending on the type of stress

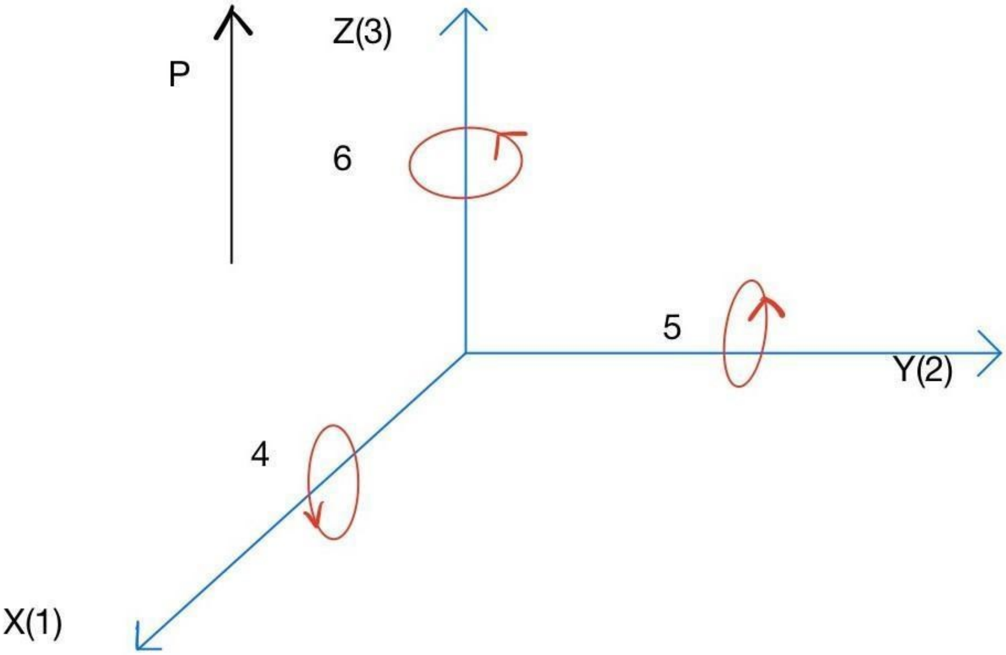


Figure 2-5 Directions of the electric displacement, piezoelectric constant, stress, dielectric permittivity, electric field, strain, and compliance coefficient. 1 indicates the positive x-axis direction, and 4 indicates the direction in which it rotates around the x-axis.

For a piezoelectric film, the poling direction is usually along with the thickness, denoted as the 3-axis, while the 1-axis and 2-axis are in the plane of the film sheet. Equation (2-4) becomes Equation (2-6) when the applied external electric field is zero.

$$\begin{bmatrix} D_1 \\ D_2 \\ D_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & d_{15} & 0 \\ 0 & 0 & 0 & d_{24} & 0 & 0 \\ d_{31} & d_{32} & d_{33} & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{bmatrix} \tag{2-6}$$

By taking the integral of the electric displacement D with respect to the area A, the generated charge q can be calculated using Equation (2-7):

$$q = \iint [D_1 \quad D_2 \quad D_3] \begin{bmatrix} dA_1 \\ dA_2 \\ dA_3 \end{bmatrix} \tag{2-7}$$

Many piezoelectric materials can be used for piezoelectric sensors, and piezoceramics (PZT) and polymer piezoelectric (PVDF) are the two most common types. Table 2-1 compares the typical characteristics of PZT and PVDF. The Young's modulus of PVDF, which describes the materials' stiffness, is only 4-6 GPa, which is approximately 1/12th of that of PZT. PVDF is therefore much more suitable for sensor applications because it is less influenced by the dynamics of the host structure and less stiff. It is also easy to adapt to any shape, even though their lower piezoelectric coefficients (approximately 1/10th of PZT).

Table 2-1 Typical properties of PZT and PVDF at 25°C [20]

	PZT-5H	PVDF
Young's Modulus (GPa)	71	4-6
d_{31} (pC/N)	-274	18-24
d_{32} (pC/N)	-274	2.5-3
d_{33} (pC/N)	593	-33
e_{33} (nF/m)	30.1	0.106

Note: pC/N is pico-Coulombs per Newton, nF/m is nano-Farad per m

Kobayashi et al. [21] developed an ultra-thin piezoelectric strain sensor based on PZT/Si material to measure two-dimensional dynamic strain. The sensor consisted of 25 ultra-thin plates (5x5 array) on a flexible printed circuit with Cu wiring for structural health monitoring. However, the fabricated sensor only showed 0.16 mV/με of sensitivity, similar to conventional strain gauges.

Instead of PZT strain sensors, Song et al. [22] studied the piezoelectric sensing of PVDF film and developed a method for low-frequency vibration measurement through dynamic strain measurement based on piezoelectric sensing. The dynamic strain was transferred to the electric signal by a piezoelectric sensor. Both theoretical analysis and experimental research on the response of sensors under different exciting frequencies (from 0.1Hz to 40Hz) were included. The experimental results indicated that piezoelectric sensors had the characteristic of high sensitivity, wide frequency-response ranges, and good linearity. The output voltage sensitivity of the PVDF sensing element was 1.75 and 3.555mV/με (cm)^2 under an exciting frequency of 0.1Hz and 40Hz, respectively.

However, the response of PZT sensors may be affected by temperature. Baptista et al. [23] presented an experimental study of the effect of temperature on the electrical impedance of the PZT sensors. The results showed that both the amplitude and frequency of impedance signatures were affected by the temperature. Hence, the temperature effect is critical for structural health monitoring, especially for small damage detection. Surakit Roya et al. [24] studied load monitoring for structural health monitoring using piezoelectric transducers. They concluded that a temperature compensation model needed to be developed to address the coupling effect of temperatures.

2.2.2.1 Summary

Overall, piezoelectric sensors can also be used as strain sensors for structural health monitoring. It is small in size and light in weight. Adhesive tapes can be used to bond piezoelectric sensors to the host structure. Sensors can be mass-produced using MEMS technology which consequently reduces the cost of each sensor. Among the two common materials for piezoelectric sensors: PZT and PVDF, PZT has higher piezoelectric coefficients, whereas PVDF has a lower Young’s modulus. However, piezoelectric sensors are not suitable for static measurements because the imperfect insulating materials and reduction in internal sensor resistance affect the output signals over time. Lastly, all the piezoelectric sensors mentioned above are either one-dimensional or two-dimensional only.

2.2.3 Strain Gauges

The electrical-resistance strain gauges are most widely used because of their relatively low cost, high accuracy, and repeatability [25]. It is a sensor that can measure the applied force based on the change in its resistance. When a strain gauge is attached to an object by a dedicated adhesive, the geometry of the strain gauge is fixed. However, when stress is applied, the object is either stretched or compressed, which causes a change in resistance of the strain gauge. This phenomenon takes advantage of the physical property of electrical conductance and its dependence on the conductor's geometry. Normally, the Wheatstone bridge circuit is used to detect the change of the resistance, which can be used to calculate the applied stress. Based on Equation (2-1), the strain can also be obtained. On the other hand, the gauge factor (GF) of a strain gauge is correlated to the applied strain in the following equation:

$$\mathrm{GF} = \frac{\Delta R/R_G}{\varepsilon} = 1 + 2v + \frac{1}{\varepsilon_l}\frac{\Delta\rho}{\rho} \tag{2-8}$$

Where △R is the change in resistance caused by applied stress or strain, R_G is the unstrained resistance of the strain gauge, $\boldsymbol{\varepsilon}$ is the applied strain, ρ is the resistivity, Δρ is the change of the resistivity, and υ is the Poisson ratio, which is the diametral strain per longitudinal strain, as shown in Equation (2-9).

$$v = \frac{\varepsilon_d}{\varepsilon_l} \tag{2-9}$$

Where the diametral strain ε_d is the variation in diameter per unit diameter and the longitudinal strain ε_l is the variation in length per unit length.

The Wheatstone bridge configuration with a voltage or current excitation source is commonly used to measure resistance changes. Wheatstone Bridge is a network composed of four resistors with an excitation source. It is the electrical equivalent of two parallel voltage divider circuits with R_1 and R_2 as one and R3 and R4 as the other. Figure 2-6 shows a Wheatstone bridge circuit with a voltage excitation source $V_{ex.}$ When the bridge is balanced, R1/R2 = R4/R3, and the output voltage Vo is zero. Equation (2-10) shows the calculation of V_o based on the resistors R_1– R_4.

$$V_o = V_{ex}(\frac{R_3}{R_3+R_4} - \frac{R_2}{R_1+R_2}) \tag{2-10}$$

If R_4 is the strain gauge with the resistance value R, the same value as R_1, $R_{2,}$ and R_3, a minor change in its resistance ΔR affects the output voltage V_o. Equation (2-11) shows the calculation of V_o.

$$V_o = V_{ex}\left(\frac{\Delta R}{4R(1+\frac{\Delta R}{2R})}\right) \approx V_{ex}\left(\frac{\Delta R}{4R}\right) \tag{2-11}$$

If the voltage source is replaced with a current source I_{ex}, and the output voltage becomes:

$$V_o = \left(\frac{I_{ex}}{R_1+R_2+R_3+R_4}\right)\left(R_3(R_1+R_2) - R_2(R_3+R_4)\right) \tag{2-12}$$

$$V_o = I_{ex}\left(\frac{\Delta R}{4\left(1+\frac{\Delta R}{4R}\right)}\right) \approx I_{ex}\left(\frac{\Delta R}{4R}\right) \tag{2-13}$$

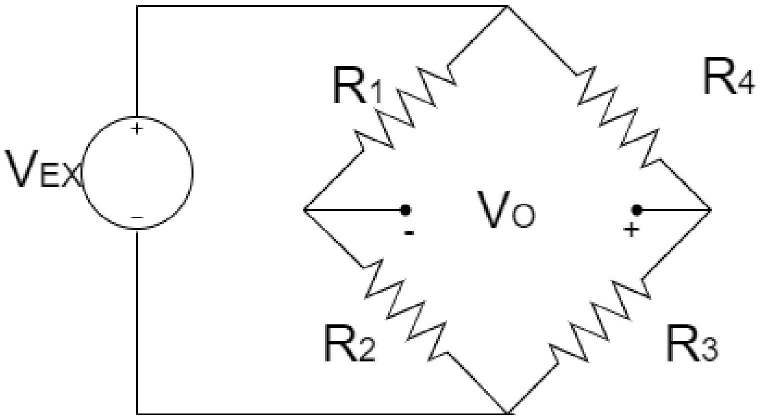


Figure 2-6 Wheatstone bridge circuit

There are mainly two types of electrical-resistance strain gauge sensors, metal-foil- and semiconductor-based.

2.2.3.1 The Metal-foil Strain Gauge

The metal-foil-based strain gauges consist of a grid of resistance wire or metal foils cut into a zigzag pattern. Metal-foil strain gauges utilizing printed circuit technology were first developed by Peter Jackson in the United Kingdom in 1952 [26]. Since then, they have shown considerable potential for industrial use. The gauge factor of a metal-foil-based strain gauge is usually around 2 to 5 [27]. It is mainly determined by the material gauge factor. Table 2-2 shows the gauge factors of the metal-foil strain gauge with some different materials. In metals, the changes in length and cross-sectional area dominate the change in resistance, while the change in resistivity is negligible.

Table 2-2 Gauge factors of metal-foil strain gauges using different metal materials [28]

Metal	Gauge factor
Platinum (Pt 100%)	6.1
Platinum-Iridium (Pt 95%, Ir 5%)	5.1
Platinum-Tungsten (Pt 92%, W 8%)	4.0
Isoelastic (Fe 55.5%, Ni 36%, Cr 8%, Mn 0.5%)	3.6
Nichrome V (Ni 80%, Cr 20%)	2.1
Karma (Ni 74%, Cr 20%, Al 3%, Fe 3%)	2.0
Monel (Ni 67%, Cu 33%)	1.9

It is important to note that both the temperature and the strain can affect the change in resistance. In that case, a temperature change may influence the strain measurement. To solve this problem, J.K. Sell et al. [29] proposed a concept for measuring strain with temperature compensation using two nested, meander-shaped strain gauges consisting of different materials. Since the two sensors were placed approximately at the same position, they were assumed to be affected by the same strain and temperature. Strain and temperature could then be solved based on the resistances of the two strain gauges, which were given by a linear system. Even though the temperature dependence of the carbon-based strain gauge was nonlinear, a reasonable linear approximation in the temperate region between 20°C and 60°C was found. Moreover, the results showed that the strain estimation was significantly improved compared to the gauge without temperature compensation techniques.

Instead of focusing on the temperature effect, D.Gräbner et al. [30] designed a new flexible, foil-based strain gauge with smaller dimensions (1.8μm in width). The result showed that it could be integrated into elastomer gaskets causing minimum damage in the host material matrix, and could be used to measure strain locally in very small areas. Nanogranular Platinum (ng-Pt) was used as a strain sensing material. However, the average gauge factor was 1.88, which was comparably low and indicated a low strain sensitivity. Furthermore, both the metal-foil strain gauges in [29] and [30] are one-dimensional.

2.2.3.2 Semiconductor Strain Gauge

Around 1970, the first semiconductor strain gauges were developed for use in the automotive industry. Semiconductor strain gauge, also known as the piezoresistive sensor, is another type of strain gauge sensor. Compared with a metal-foil-based strain gauge, it usually has a larger gauge factor, making it more sensitive. Unlike metal-foil-based strain gauges that use metallic materials as sensitive elements, the semiconductor-based strain gauges use semiconducting materials as sensitive elements. It is made from semiconductor material in which a p-type region has been diffused into an n-type base. In semiconductor materials like silicon or germanium, the change in resistivity dominates the change in resistance. Consequently, the change in resistivity dominates the change in resistance, which provides a gauge factor of around 50 times that of metal-foil strain gauges. When it is stretched or compressed, its resistance changes according to the piezoresistive effect. It has high sensitivity, low hysteresis, and integration with chip electronics. T. Toriyama et al. [31] developed a 100nm square single crystal silicon strain gauge that could measure shear stress and strain. H. Kuo et al. [32] studied the high-performance piezoresistive strain micro sensor for field applications. The relationship between temperature stability and doping level was investigated. The experimental results showed that heavily doped sensors were less sensitive to temperature variation. However, the gauge factor of the heavily doped sensor was lower. Instead of increasing doping level to minimize temperature variation, Stephen P. Olson et al. [33] integrated a temperature sensor onto the piezoresistive strain gauge sensor. Temperature data were recorded with the stress and strain data to allow the signal to be temperature compensated. The test result showed that the fabricated strain gauges could respond to load starting from unloaded with a step of 10με while dissipating only 14μW of power. Y.Kim et al. [34] proposed a thin polysilicon strain gauge for the measurement of strain in structural elements, and the strain gauge element was separated from the silicon wafer. The sensitivity was relatively high since the gauge factor was approximately 34.0, which was considerably higher than a conventional metal-foil-based strain gauge. Moreover, the temperature coefficient of resistance was significantly lower than that of mono-silicon strain gauges.

However, all the above sensors can only monitor uniaxial strain. They cannot measure three-dimensional stress and strain, which play an important role in stress analysis. A stress sensor capable of monitoring the 3D stress state can be a valuable alternative for detecting bonded structures' debonding compared to the conventional sensors [3].

Miura et al. [35] designed the first piezoresistive stress sensing rosette, in which additional sensors were used to provide strain measurements in multiple directions. The designed sensing rosette was capable of extracting four of six stress components: the three in-plane stress components and the out-of-plane normal stress component. Bittle et al. [36] proposed the first comprehensive three-dimensional stress sensing rosette utilizing piezoresistive elements. It was later modified by Suhling et al. [37], who considered temperature compensation. The developed sensor could measure the complete three-dimensional stress state and provide reduced equation forms for some stress components and better stress localization. However, temperature compensation was not provided to all the output stress.

To solve this problem, a three-dimensional sensing rosette, as shown in Figure 2-7, was designed by Gharib [3, 38-39]. It added two more sensing elements to this 8-element sensing rosette. As a result, a total of 10 elements made up the sensing rosette capable of extracting all stress components with temperature compensation. Also, the MEMS technology was employed to provide smaller dimensional sizes, higher sensitivity, and resolutions. Compared with conventional strain sensors, MEMS sensors have higher performance and lower power consumption. Furthermore, this 3D strain sensor could be either dual- or single-polarity sensing elements through utilizing the unique behavior of the shear piezoresistive coefficient in n-Si with impurity concentration. Ten sensing elements were divided into three groups where each group had independent piezoresistive coefficients and temperature coefficients of resistance, which generated linearly independent equations to extract the six stresses with partial temperature compensation. In [3], the 3D stress sensing rosette was designed, micro-fabricated, and calibrated using the uniaxial, thermal, and hydrostatic loads. It was also tested by performing a four-point bending of a chip-on-beam specimen at room temperature.

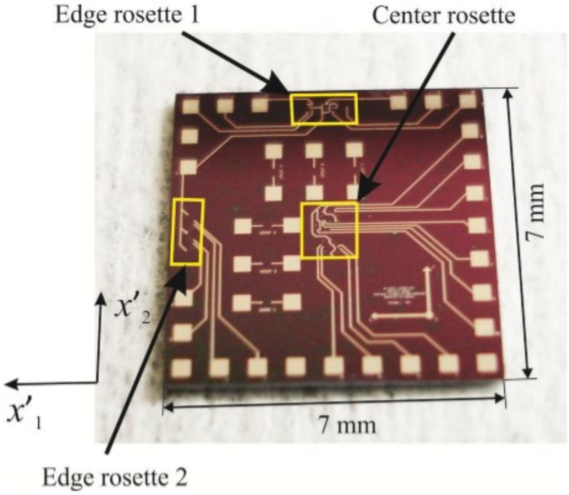


Figure 2-7 Three-dimensional piezoresistive strain gauge [3]

2.2.3.3 Summary

Overall, strain gauge sensors are widely used as strain sensors for structural health monitoring. They are small in size and light in weight. They can be bonded to the host structures easily. They only require minimum maintenance and have a long operating life. Recently, strain gauge sensors can be packaged using MEMS technology. The MEMS sensors are preferred now because of the higher accuracy and stability, smaller size, and lower power consumption [40].

2.2.4 Sensors Summary

Variety types of stress and strain sensors could be used for structural health monitoring; three most commonly used sensors have been reported in the literature: optical fiber, piezoelectric, and strain gauge. Table 2-3 provides a comparison of their advantages and disadvantages. Because of the relatively higher price and greater installation difficulty of optical fiber sensors and the deficiency of piezoelectric sensors in the area of truly static stress and strain measurement, strain gauge sensors were preferred for structural health monitoring. Furthermore, semiconductor strain gauges have a higher gauge factor, high sensitivity, low hysteresis, and integration with chip electronics compared with metal-foil strain gauges. The three-dimensional piezoresistive sensing rosette designed and fabricated by the mechanical department of the University of Alberta was chosen to be used [3]. It is capable of extracting all six stress components with temperature compensation.

Table 2-3 Comparison of different types of sensors

	Optical fiber	Piezoelectric	Strain gauge
Install complexity	High	Low	Low
Maintenance difficulty	High	Low	Low
Mass production	No	Yes	Yes
Cost	High	Low	Low
3D	No	No	Yes
Temperature effect	Yes	Yes	Yes
Static measurement	Yes	No	Yes

2.3 Data Acquisition System

A data acquisition system is a unit that can acquire sampled responses from sensors and convert the resulting samples into digital values that the microcontroller can process. If data is required to transmit wirelessly, a communication protocol, a system of rules that create consistency and universality for information transmission, is required to exchange information in or between network systems. The protocol defines the rules, syntax, semantics, and synchronization of communication. It also covers authentication, error detection, and recovery methods. Many wireless communication protocols can be implemented in hardware products for the Internet of Things (IoT), such as Wireless Fidelity (Wi-Fi), Z-Wave, ANT, ZigBee, Bluetooth, and Bluetooth Low Energy. Four typical communication protocols, including Wi-Fi (Section 2.3.1), ZigBee (Section 2.3.2), Bluetooth (Section 2.3.3), and Bluetooth Low Energy (Section 2.3.4), and the data acquisition systems based on them are discussed in Section 2.3. Section 2.3.5 provides a summary of the data acquisition systems.

2.3.1 Wi-Fi-Based Data Acquisition System

Wireless Fidelity, also known as Wi-Fi, is a type of wireless communication protocol that has been widely used in reality. It is based on the IEEE 802.11 family of standards, the most commonly used wireless computer networking standard. Instead of using Internet cables to transmit data through an Ethernet network, Wi-Fi uses radio waves for data transmission, eliminating the physical connection between devices. In this process, modulation and demodulation of the carrier wave were done. In most cases, 2.4 GHz ultra-high frequency (UHF) and 5GHz super high frequency (SHF) Industrial, Scientific, and Medical band (ISM) radio bands are used. They are subdivided into multiple channels, which can be shared between networks. However, only one transmitter can locally transmit data on a channel at one time. Besides Wi-Fi, ZigBee, Bluetooth, and BLE are the most common ways of transmitting a small amount of data over a limited range. Wi-Fi is popular for a long communication range and high data rate, as well as instant access to the cloud. Nowadays, it is readily available in most areas. Hence, developing a topology to extend the transmission coverage further is not necessary.

Researchers have developed Wi-Fi-based IoT platforms for different sensors [41-44]. A low-cost, stand-alone, Arduino-based health monitoring system for the bridge was developed [41], in which an Adafruit Wi-Fi module was used to transmit vibration data measured from a 3-axis digital accelerometer to the local Internet. It used a web-based API to run as a GET request link from Arduino; the Google script was run to acquire data, which was quite novel. Also, since the accelerometer used digital signals to communicate, the use of ADC was eliminated. In [42], Malik et al. also proposed a low-cost, reliable, and secure IoT-based structural health monitoring platform with Arduino. Multiple sensors for different parameters such as vibration, tilt, shock, strain, temperature, and humidity were integrated. An external Wi-Fi module transmitted sensor readings to an open-source cloud platform ThingSpeak for processing, storage, and analysis. Mahmud, Md Anam et al. [43] also developed a complete IoT platform for SHM using Wi-Fi for data transmission.

In this platform, PZT sensors were implemented. The collected data were fed into an ADC and then read and stored by a Raspberry Pi. It has to be noted that the use of Raspberry Pi limits the dimensional size of the product. The Raspberry Pi used a Butterworth filter to remove noises and a mathematical model based on a combination of Pitch-catch and pulse-echo techniques to calculate the accurate location and size of the damage. Another low-cost distributed embedded system for structural health monitoring using Wi-Fi was presented in [44]. A Wi-Fi transceiver was used to transmit data to a NoSQL database in the cloud. Unlike the Arduino-based platform in [41] that used digital accelerometers, four analog accelerometers were used in [44], and the internal ADC of the MCU was implemented to provide data. On-board online filtering and fusion of the collected data were also provided. The synchronization algorithm was implemented, and the signal-to-noise ratio was reduced with a spatial redundancy of the sensors.

However, the Wi-Fi-based systems discussed above focused mainly on low-cost instead of low power consumption, a major limitation for Wi-Fi-based SHM systems. Compared with other wireless transmission protocols, Wi-Fi has a larger transmission range and draws more power at the same time. In particular, a trade-off between functionality and power consumption has to be taken into consideration. Currently, rechargeable batteries are the most common portable power source for wireless sensors. Only a finite amount of power is contained without charging; when batteries are exhausted, replacements can be difficult, especially when there are many sensors in remote areas.

Power consumption and battery lifetime were not mentioned at all in [41,43-44]. In [42], the current consumption of the proposed was tested at a sampling frequency of 1Hz. Roughly 38 mA and 117 mA currents were consumed without and with the Wi-Fi module running, respectively. The author stated that it could operate for about 2.5 days between recharging. However, this was doable on the basis of using a 10,400mAh Xiaomi power bank instead of a rechargeable battery. The use of a power bank increases both the cost and physical dimensions of the platform, which is not suitable for large-scale SHM. Moreover, a higher sampling frequency may increase the power consumption further.

In order to reduce power consumption, Lei Sun et al. [45] introduced the compressed sensing theory, which could improve the system's overall performance by compressing the sampled signals. Signals generated by accelerometer sensors were sampled and compressed simultaneously using a Field Programmable Gate Array (FPGA) chip (EP4CE6E22) and then transmitted using Wi-Fi. Finally, the accelerometer signals were reconstructed from the compressed signals in order to determine the structural state. Lab experiments validated the hardware platform that the reconstructed signal was good for SHM. However, it has not been validated for a long-term and large-scale SHM study.

Overall, Wi-Fi-based data acquisition systems can upload data to the cloud directly without considering the distance between two sensor nodes. Moreover, Wi-Fi-based platforms have a broad bandwidth to transmit a large amount of data. However, the high-power consumption of Wi-Fi makes battery life a concern.

2.3.2 ZigBee-Based Data Acquisition System

ZigBee is another popular wireless communication protocol that mainly operates in the 2.4 GHz frequency band but can also support the other frequency bands for ISM, such as 868, 915, and 920 MHz bands. The offset quadrature phase-shift keying (OQPSK) is used for signal modulation. Since the IEEE 802.15.4 standards specify the lower protocol layers for small devices with limited communication capabilities, it features low power consumption. However, the low power consumption limits the transmission distance to 10~100 meters depending on the power output and the conditions of the environment. It needs to be noted that ZigBee does not have direct access to the internet, unlike Wi-Fi, which can upload data directly to the Internet. Most smartphones and computers do not yet have direct support for ZigBee. In order to transmit data over a long distance, a mesh network of multiple intermediate ZigBee devices is needed to pass the data through and finally reach the more distant ones. A node discovery mechanism using broadcasting is used for network formation, and an Ad-hoc On-Demand Distance Vector is used as the default routing protocol for networks organized in a mesh topology. The data throughput can be up to 250kb/s but is typically used at much lower data throughput, which reduces the power consumption of sensing devices. In terms of security, a 128-bit symmetric encryption key is used. Overall, ZigBee is particularly suitable for scenarios where only infrequent data package exchange is required and power consumption is a priority. It is commonly used in the data acquisition system for structural health monitoring.

Gao et al. developed a high-performance wireless sensor platform using ZigBee for distributed SHM [46]. A set of PZT transducers were deployed at the structural surface. Some of them worked as the actuators generating the lamb-wave, while the others worked as sensors to locate the damages based on the receiving lamb waves. The sensing nodes used embedded distributed lamb-wave data processing for information extraction. A high sampling frequency was implemented, ensuring detection of some fast and valuable changes in the monitored parameters. A down-sampling algorithm was implemented to reduce the large amount of data collected with a high sampling frequency, consequently minimizing transmission time and memory requirement. Additionally, a dual-processor architecture was implemented because the errors in time synchronization could degrade both the accuracy of arrival time calculation and the resolution of damage localization. The main processor was used for computation-intensive data processing, while the second processor was used for wireless communication and time synchronization. However, this architecture increased both the product size and power consumption which were not mentioned. Furthermore, the transmission distance was not mentioned as well. Since data were sent from sensor nodes to the base station directly, the transmission distance was limited by the transceiver and was not expected to be adequate for large-scale structural health monitoring. Sadoudi et al. also proposed a ZigBee-based wireless sensor network for the health monitoring of rails [47]. Similar to the network proposed by Gao et al., the network consisted of multiple sensor nodes and a base station. However, the power consumption was also not mentioned. Besides that, instead of using PZT sensors, multiple acoustic sensors are coupled with ZigBee modules for health monitoring, and the detection principle was based on acoustic noise correlation techniques. Experiments have been conducted to validate the feasibility of the noise correlation techniques. Liu et al. also proposed a practical structural health monitoring platform called SenetSHM which also used ZigBee for data transmission [48]. Three-axis accelerometers and an external ADC were used to acquire data. Unlike other platforms, SenetSHM could be configured for two different SHM applications, short-term and long-term. In the short-term mode, SenetSHM was capable of achieving high frequency and synchronized sensing at the same time during the process of real-time onboard data storage. In long-term mode, a reliable wake-up strategy and in-network processing were implemented to save energy while it included a vibration-triggered wake-up and radio-triggered wake-up unit. Moreover, an effective wireless sensor network (WSN) tailored SHM algorithm was proposed. It has to be noted that the power consumption increased when the number of electrical components increased while the system was not in the sleep low-power mode.

Instead of using a single way of wireless transmission, a combination of ZigBee and a very small aperture terminal (VSAT) network was employed in a large-scale structural health monitoring system for long-span bridges [49]. The Zigbee was used in the local wireless network in order to achieve multi-hop communication, while the VSAT network connected the local network to the remote control center. Interestingly, a combination of clustering topology and hybrid technique in the routing protocol was used. The hybrid technique solved the problem of unbalanced power consumption caused by multi-hop communication. The clustering topology decreased the power consumption and complexity of network management.

2.3.3 Bluetooth-Based Data Acquisition System

Bluetooth is another low-power commonly used short-distance wireless communication protocol that was firstly proposed in 1994 by Ericsson. Nowadays, it is managed by the Bluetooth Special Interest Group cooperation made up of numerous companies in the region of telecommunications technology, and the standard has evolved from version 1.0 to the current version 5.2. At the same time, more researchers are focusing on the field of IoT. It has been seen as one of the primary standards to support a wireless real-time communication solution capable of coping with the limitations of open environment deployment. As opposed to ZigBee, which is developed based on IEEE 802.15.4, Bluetooth is based on IEEE802.15.1. It operates at frequencies between 2.400 and 2.4835GHz, including a 2MHz wide guard band at the bottom end and a 3.5MHz wide one at the top, which is in the globally unlicensed ISM 2.4GHz short-range radio frequency band. A radio technology called frequency-hopping spread spectrum is used. Bluetooth is a packet-based protocol with a leader/follower architecture. The data are divided into packets, and each packet is transmitted on a 1MHz bandwidth Bluetooth channel. One leader may communicate with up to seven slaves in a piconet. All devices share the leader's clock. Packet exchange is based on the clock defined by the leader, which ticks at 312.5µs intervals. In the simple case of single-slot packets, the leader transmits in even slots and receives in odd slots. The follower, conversely, receives in even slots and transmits in odd slots.

Compared with the ZigBee protocol, Bluetooth is a more reliable transmission protocol that does not require additional management for data loss and has been used in many structural health monitoring systems. A design and realization of a multi-channel wireless data acquisition system based on IEEE 802.15.1 (Bluetooth) was presented [50] for laboratory-scale experiments on structural health monitoring. A board named KACC was designed and developed; it could acquire tri-axial acceleration data and transmit wirelessly. It had a long-lasting and easily rechargeable battery, and the power consumption was tested. When the device was in the transmit mode, the power consumption was measured at 120mW, while in the idle mode, it was 17mW, respectively. Furthermore, multi-channel data acquisition was also supported, and three channels were available in this study (can be enlarged by additional KACCs). However, this study has a major drawback which is the limited transmission range of the Bluetooth. Regardless of the number of channels, the transmission distance was limited and cannot be enlarged since the device can only transmit data from KACC to the developed software. B. M. Albaker also built a structural health monitoring system that implemented Bluetooth technology to send sensor readings remotely to a ground station [51]. Two flex sensors were attached to the sidebars of the bridge and transmitted collected data to the Arduino; two thresholds were used to split the level of risk into three levels, and a buzzer was added to generate an alarm for warning before a bridge might collapse. However, this system was specialized for highway bridges and might not be applicable for other engineering structures. The power consumption was also not mentioned. In [52], Chen proposed an application layer gateway system handling wireless communication between wireless sensor networks and cellular networks. Network packet loss management and re-transmission of the gateway system were included and discussed in detail. It has been proven that the communication quality of the gateway system was good when the communication distance was smaller than 300m. Furthermore, when ten sensor nodes transmitted data at the same time, the data loss rate could be controlled within 0.1%, which showed a good control effect and satisfied the practical requirement.

2.3.4 Bluetooth Low Energy-Based Data Acquisition System

Bluetooth Low Energy (BLE) is another wireless personal area network technology that operates in the same spectrum range as classic Bluetooth technology. It is available on all the latest smartphones, tablets, and computers. Compared to Classic Bluetooth, it is intended to provide considerably reduced power consumption and cost while maintaining a similar communication range. BLE can remain in sleep mode constantly if no wake-up source is triggered, while it consumes much less power in sleep mode compared with power consumption in active mode. For example, BlueNRG-2, a BLE micro-controller, consumes only 0.9µA in sleep mode with a DC-DC converter active (25°C and 3V voltage supply) which is much smaller than 1.9mA in active mode. This considerable save in power is achieved by stopping the CPU and disabling all the peripherals. In sleep mode, only the digital power supplies necessary to keep the Random Access Memory (RAM) in retention are used, and the low-frequency oscillator is switched on. With the low power consumption of BLE, applications can run on a small battery for even years. Furthermore, the lower cost of BLE modules and chipsets compared to other similar technologies meets the requirement of low production cost. BLE also operates in the same frequency band as Bluetooth, but it uses a different set of channels. Bluetooth Low Energy has 40 2-MHz channels instead of the classic Bluetooth 79 1-MHz channels. Within a channel, data is transmitted using Gaussian frequency shift modulation, which is similar to classic Bluetooth's Basic Rate scheme. The bit rate is 1Mbit/s (with an option of 2Mbit/s in Bluetooth 5), and the maximum transmit power is 10mW (100mW in Bluetooth 5).

The physical layer (PHY) involves the physical radio used for communication and for modulating/demodulating the data. Gaussian frequency-shift keying GFSK modulation is used. Moreover, an adaptive frequency hopping method is used over data channels to minimize interference from other technologies.

The Generic Attribute Profile (GATT) layer specifies sub-procedures that handle two BLE devices' data communications. It also defines the format of the data exposed by a BLE device and the procedures required to access the data exposed by a device.

Compared with other wireless communication protocols, only limited research used BLE in the SHM system. Paul et al. designed an IoT-based framework to detect the collapsing probability of structures and alert the inhabitants by buzzing an alarm and lighting a red led [53]. A portable device named 'Structure Analyzer' was designed. It was mounted on a concrete beam-column joint, metal beams, and bridge joints. At the same time, the built-in 6-axis accelerometer/gyroscope could measure the angle at that position. The data were analyzed by Arduino 101 and then sent to the smartphone app through BLE. However, Paul et al. also added a Wi-Fi module to upload data to the cloud directly since the low power consumption was not the focus of this study. Although the physical dimensions and current consumption were not mentioned, they were not expected to be low due to the Arduino 101 unit and Wi-Fi module.

2.3.5 Summary

Structural health monitoring network systems that implement four typical wireless transmission technologies are discussed, including Wi-Fi, ZigBee, Bluetooth, and BLE. Among these four protocols, Wi-Fi has the advantage of direct access to the Internet and computer and a large data rate. But the power consumption is high. Alternatively, ZigBee and Bluetooth have lower power consumption, which is particularly suitable for scenarios where only infrequent data package exchange is required, and power consumption is a priority. It is a more reliable transmission protocol that does not require additional data loss management than the ZigBee protocol and has been used in many structural health monitoring systems. BLE was built on the basis of Bluetooth, which consumes even less power. In that case, BLE was preferred for data transmission. However, most studies on IoT-based SHM platforms utilize Wi-Fi and ZigBee, and the power consumption of the IoT-based SHM systems is still one of the challenges [41-53].

Chapter 3. Hardware Development

3.1 Overview

The IoT-based structural health monitoring system developed in this research comprises three components: a) the sensing node, including the sensor and the sensing acquisition platform, b) the router, and c) the gateway. Section 3.2.1 provides the hardware design details and the rationale of the components' selection of the sensing node. Section 3.2.2 describes the router design and specification, and Section 3.2.3 briefly describes the selection of the gateway. Section 3.3 provides a summary.

3.2 Hardware Design

Figure 3-1 shows the block diagram of the IoT architecture, which includes sensing nodes, routers, a gateway, and a Cloud. A sensing node is a custom low power acquisition platform embedded with a 3D stress-strain MEMS sensor which was developed by Balbola et al. [38-39]. The measured data were transmitted to a router, a custom platform designed in this study, and then to the gateway via BLE. Unlike Wi-Fi communication, BLE consumes much low power and communicates with a shorter distance range. Therefore, routers were added in-between a node and a gateway to extend the BLE communication range in this design. A custom router was also designed, which served as an extender and a data storage buffer. A Raspberry pi 4B evaluation board (Raspberry Pi Foundation, United Kingdom) was selected as the gateway to upload data to the Cloud.

In this study, the developed IoT system aimed to monitor 3D stress and strain values of an oil-sand structure of Syncrude Canada to detect early cracking. The design requirements of the sensing node are the system must a) be able to handle temperature between -40°C to +70°C, b) consume low power, c) be able to last at least one week during the inspection period, d) be easy to install, e) be able to acquire data at 5 Hz and up to 20 Hz as required by Syncrude Canada, e) have physical dimensions smaller than 4cm by 4cm, and f) be able to transmit data to a gateway which could upload the data to Cloud.

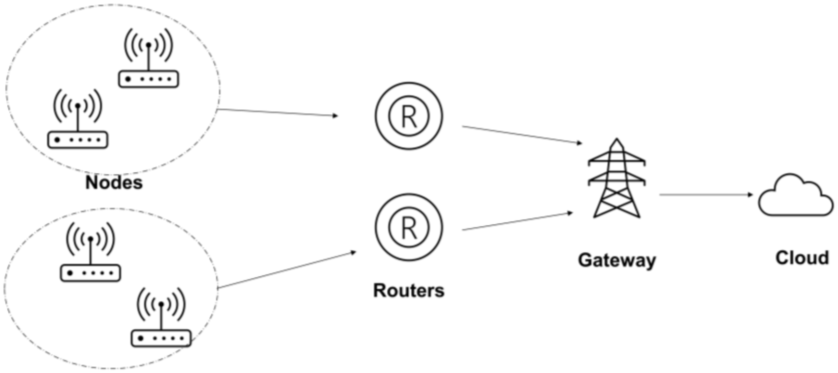


Figure 3-1 Internet of things architecture, routers are included to extend the communication range between nodes and gateway

3.2.1 Sensing Node

Each sensing node consists of an acquisition platform and a 3D stress-strain sensor. In a simple view, the functionality of the sensing acquisition platform is to continuously measure at a sampling frequency of 5Hz or 20Hz, store the stress-strain values, and then transmit the collected data to a router periodically. Figure 3-2 shows the block diagram of the developed sensing acquisition platform. From the block diagram, the platform was mainly composed of five components: a) a system-on-chip (SoC) which integrates a microcontroller with a built-in BLE radio, b) a 16-bit analog-to-digital converter (ADC), c) the 3D MEMS strain gauge typed sensor, d) an external 32 Mbytes flash memory, and e) a battery with the power management circuitry. Each of the individual components in the block diagram and the interfaces between them are discussed in the following sections. It has to be noted that all components selected are in commercial grades and have an operating temperature from -40°C to 125°C to ensure functionality and reliability in a harsh environment.

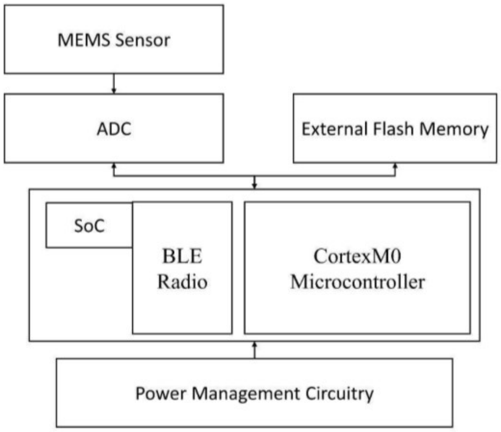


Figure 3-2 Block diagram of the node platform

3.2.1.1 System-on-Chip (SoC)

In a modern design, the system-on-chip (SoC) is usually the major component of an embedded system. There are many SoCs that integrate a microcontroller with a BLE radio to minimize the number of components, consequently reducing the overall dimensions. Since the node platform was designed to meet low power and small size requirements, the microcontroller has to have the following characteristics:

- Consists of Serial Peripheral Interface (SPI) to communicate with external ADC and external flash memory,
- Has an operating temperature from -40℃ to 125℃ to ensure functionality and reliability in a harsh environment,
- Contains internal flash memory with at least 200kB (150kB for programming code and 50kB for temporary data storage) to minimize frequently data transferring from internal MCU memory to external flash memory,
- Features a power-down/sleep mode in which the SoC consumes less than 1mA， and
- Small physical dimensions of approximately 8mm x 8mm. Hence, with the support of passive components, the data acquisition platform is smaller than 4cm x 4cm.

There are many SoCs available in the market which may meet our design specifications. Six of them, IS1870SF-202-TRAY, CY8C4247LQI-BL493, CYW20707, CC2640FRGZT, BLUENRG-232, and EFR32BG12P232F512IM68-C, have been considered due to their built-in BLE radio module and low power mode capability. Table 3-1 shows a comparison between these six SoCs based on the above six parameters: a) SPI interface, b) internal flash memory size, c) BLE version, d) current consumption at sleep or low power mode, e) physical dimensions package, and f) the number of general-purpose input/output pins (GPIO).

Among the six SoCs, BlueNRG-232 has the lowest current consumption in sleep or low power mode, which is 0.9 µA at a 3.0V power supply. Similarly, CC2640F128RGZT has a low sleep mode current consumption of 1 µA at 3.0V, followed by 1.5 µA of CY8C4247LQI-BL493 and EFR32BG12P232F512IM68-C at 3.3V. On the contrary, CYW20707 has a relatively higher current consumption, which is 2.69 µA at 3.0V. Other than current consumption in sleep mode, the BLE version is also essential for SoCs. Of the remaining five choices, only BLUENRG-232 and EFR32BG12P232F512IM68-C are compliant with BLE 5.0. BLE 5.0 has more advanced features and capabilities when compared with BLE 4.2. Table 3-2 shows a detailed comparison between BLE 4.2 and BLE 5.0.

Ultimately, BlueNRG-232 was selected because of the superior features mentioned above. It has 256kB internal flash and 15 general-purpose input/output pins (GPIO). As for the SPI interface, this SoC supports 1 MHz and 8MHz in slave and master mode, respectively.

Table 3-1 Five SoCs with built-in BLE

Part name	Serial interface (SPI)	Programmable flash （kB）	BLE	Sleep/Low power mode current consumption	Package (mm x mm)	# of GPIO
EFR32BG12P23 2F512IM68-C	Yes	512	v5.0	2.1 µA	68-QFN (8x8)	46
BLUENRG-232	**Yes**	**256**	**v5.0**	**0.9 µA**	**32-QFN (5x5)**	**15**
CC2640F128RG ZT	Yes	128	v4.2	1 µA	VQFN-48 (7 x 7)	31
CYW20707	Yes	848	v4.2	2.69µA	49-FBGA (4.5 x 4)	24
CY8C4247LQI-BL493	Yes	128	v4.2	1.5 µA	56-QFN (7x7)	36
IS1870SF-202-TRAY	Yes	256	v4.2	60 µA	48-QFN (6x6)	31

Table 3-2 Comparison between Bluetooth 5.0 and Bluetooth 4.2[54]

Specifications	Bluetooth 5.0	Bluetooth 4.2
Speed	2x the data rate, supports up to 2Mbps	Supports up to 1Mbps
Range	4x the range supports, 40 meters indoor environment	Supports 10 meters indoor
Power requirement	Low	High
Message capacity	Large message capacity, about 255 bytes	Small message capacity, about 31 bytes
Battery life	Longer	Less
Theoretical data	2Mbps	1Mbps
Reliability	High	Low
Digital life	Better	Less good compared to Bluetooth 5.0

3.2.1.2 Analog to Digital Converter

Since the sensing signals measured from the sensor are analog, an analog-to-digital converter (ADC) is required to convert them to digital signals so that they are readable and processable by the microcontroller. In the selected SoC, it has a built-in 10-bit analog-to-digital converter. However, a higher resolution ADC is needed to have a high sensitivity to detect the stress and strain on the MEMS strain gauge. A proper ADC should meet the following requirements:

- Has a sampling rate higher than 100k-Samples/s in total channels,
- Has a physical dimension smaller than 5mm x 5mm,
- Supports SPI interface to communicate with the microcontroller,
- Has at least four channels,
- Supports external voltage reference to increase sensitivity,

- Has an internal sequencer to sequentially and automatically convert a list of predefined input channels without requiring communications with the MCU,
- Has high signal-to-noise (S/N) ratio (80dB minimum) and low integral nonlinearity (INL) error (less than ±2 least significant bit (LSB), which is 30ppm full-scale range (FSR)) in order to have high accuracy.

To satisfy the above requirements, four 16-bit ADCs were considered: ADC7682, MCP3462, ADS8331IBRGET, and ADS7066IYBHT. They all support external voltage reference, support SPI interface, have at least four channels, and have a sampling rate higher than 100 kS/s in total channels, as shown in Table 3-3. They all feature an internal sequencer or timer, making the ADCs easy to use and suitable for acquiring multiple channels in sequence.

Signal-to-noise (S/N) ratio and integral nonlinearity (INL) error are two important factors to evaluate the performance of ADC. The S/N ratio is a metric that compares the strength of the desired signal to the strength of background noise. Meanwhile, integral nonlinearity is the measure of deviation from the ideal input threshold value. Among all the choices, MCP3462 has the highest S/N ratio and the smallest integral nonlinearity error. Besides, MCP3462 also has the lowest current consumption of 0.25mA, while ADS7066IYBHT consumes 0.91mA, and the other two consume more than 1mA. Furthermore, MCP3462 has many other features, such as oversampling ratio (OSR) and a wide programmable gain. Oversampling is a cost-effective process of sampling the input signal at a much higher rate than the Nyquist frequency. A high oversampling ratio results in overall reduced noise. Using a variable gain can also eliminate the usage of an external amplifier which reduces the physical dimensions of the overall design. After all of the consideration points, the MCP3462 was chosen.

Table 3-3 Four analog-to-digital converters

Part Name	ADC7682 (Analog Devices Inc.)	**MCP3462 (Microchip Technology)**	ADS8331IBRGET (Texas Instruments)	ADS7066IYBHT (Texas Instruments)
Sampling rate (kS/s)	250	**153.6**	500	250
SNR - Signal to Noise Ratio (dB)	92.5	**97.3**	91.5	91.9
Package (mm x mm)	20-LFCSP (4 × 4)	**UQFN-20** **(3 x 3)**	VQFN-24 (4 x 4)	DSBGA-16 (1.8 x 1.8)
# of channels	4	**2/4**	4	8
Current consumption (mA)	1.4	**0.25**	3.2	0.91
Analog & Digital Supply Voltage (V)	2.3 -5.5	**2.7-3.6** **1.8-3.6**	2.7-5.5 1.65-5.5	3-5.5 1.65-5.5
INL - Max Integral Nonlinearity Error (ppm FSR)	±23	**± 7**	±30	±15
Other features		**Programmable gain (from 1/3x to 64x);** **Oversampling Ratio (32 to 98304)**		Averaging filters

In addition, the MCP3462 is a Delta-Sigma ADC, which makes use of oversampling signal data and applies digital filters, ensuring high-resolution performance and providing high stability. If a 1.5V reference voltage and a gain of 1 were used, the resolution of this 16-bit ADC was 45.8µV. Equation (3-1) shows the resolution calculation.

$$V_{1lsb} = \frac{V_{REF}}{32768 \times GAIN} = \frac{1.5}{32768 \times 1} = 45.8\mu V \tag{3-1}$$

Where the least significant bit (LSB) is the smallest level of the ADC that can convert, and V_{1lsb} is the analog quantity corresponding to one step size in an ADC, and V_{REF} is the reference voltage.

3.2.1.3 MEMS Strain Gauge Typed Sensor

A 3D stress-strain MEMS sensor that can fully extract the six stress components with temperature compensation was selected for this thesis [38-39]. This sensor was designed and developed by a team in mechanical engineering at the University of Alberta. It is capable of extracting both in-plane and out-of-plane stress components. Figure 3-3 shows the MEMS sensor populated on a custom-designed printed circuit board (PCB).

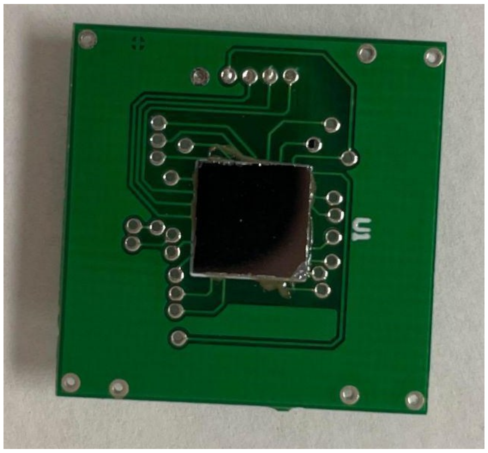


Figure 3-3 A PCB with the MEMS stress and strain sensor assembled in the middle

This stress-strain sensor consists of three sensing rosettes; one is located at the center, and two additional rosettes are located at two edges, as shown in Figure 3-4. The center rosette consists of ten sensing elements to measure the main monitored stress. In contrast, two edge rosettes are two reduced rosettes to collect the out-of-plane shear stress [3].

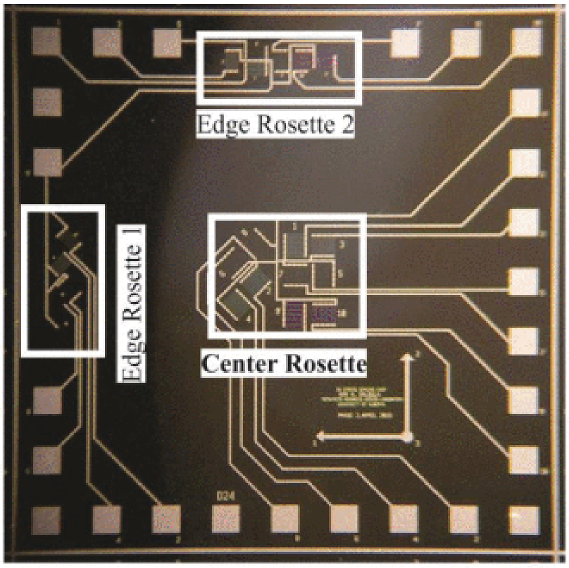


Figure 3-4 A graph of the actual scale of the sensor [39]

The developed chip was fabricated onto (111) silicon, where the six stress states have an influence on the resistivity of a piezoresistor oriented over this plane. To extract the six temperature-compensated stress components, ten linearly independent equations were obtained from those piezoresistors. The details of the description can be found in [39].

Inside the sensor, there are 20 sensing elements. Among those, 17 could be used to export 3D stress-strain information. The rest three were designed for temperature measurements. All the sensing elements were designed for low power application, and the resistance range was between 0.8kΩ to 1.5kΩ while the current drew on each sensing element was approximately 100 µA. Under different experiments, the tested sensor managed to extract both the out-of-plane shear and normal stress components with 16% and 11% full-scale error [55].

3.2.1.4 External Flash Memory

A temporary memory using external flash memory is needed before sending the collected data from the sensing node to the gateway. Since each channel requires 3 bytes (holding 16-bit data) for storage, a total of 0.9Mbytes and 3.7Mbytes for all 17 channels are collected per hour at a sampling frequency of 5Hz or 20Hz, respectively. Equations (3-2) and (3-3) show the data size calculation for 1-hour data. Given the fact that the SoC only has 256kbytes internal flash memory, external flash memory is required to avoid frequent data transmission. Non-volatile memory (NVM) is an option, and it can retain stored data even after power is off. In contrast, a volatile memory needs a constant and continuous power supply to retain data. If the battery dies before sensing out data, all data stored in volatile memory will be lost. Hence, only non-volatile memories were considered.

$$Data\ size\ (5Hz) = 5\frac{samples}{s} \times \frac{3600s}{h} \times 17channels \times 3bytes\ x\ 1\ hr = 0.9Mbytes \tag{3-2}$$

$$Data\ size\ (20Hz) = 20\frac{samples}{s} \times \frac{3600s}{h} \times 17channels \times 3bytes\ x\ 1\ hr = 3.7Mbytes \tag{3-3}$$

For a 32Mbytes flash memory, it can store 34.6 hours of sampling data if the sampling frequency is kept in 5Hz, or 8.7 hours if the sampling frequency is kept in 20Hz. There are few SPI flash memories with 32MBytes size available for selection, and three of them are S25FL256L, W25Q256JV, and GD25Q256DFIGR. They were considered due to their low current consumption in the power-down mode. These three memories were compared in three aspects: the package, the operation time for page program and sector erase, and current consumption in reading, programming, and erasing, as revealed in Table 3-4. All of them consume less than 2μA in low power mode. Although the maximum supported SPI clock frequencies are different, the microcontroller can only handle up to 8MHz, which is achievable for all three memories.

As shown in Table 3-4, S25FL256L and W25Q256JV have the erase time and the erase current consumption (50ms and 20 mA), and both values are lower than those of GD25Q256DFIGR. The packages of the S25FL256L and W25Q256JV are also the same and have smaller physical dimensions than that of GD25Q256DFIGR. In view of the page program, W25Q256JV has a lower current consumption, which is half of that of S25FL256L, but the time is more than twice. Additionally, S25FL256L has a lower current consumption in the read operation. Eventually, S25FL256L was selected to store data primarily due to the lower current consumption and the smaller physical dimensions.

Table 3-4 Three 32Mbytes non-volatile external flash memories

Part name			**S25FL256L**			W25Q256JV			GD25Q256DFIGR		
Package (mm x mm)			**8-WSON (6x8)**			8-WSON (6x8)			SOP-16 (7.5x10.3)		
	Page program time	Erase time (ms)		**0.3**	**50**		0.7	50		0.4	70
Read current (mA)	Program current (mA)	Erase current (mA)	**10**	**40**	**20**	20	20	20	15	30	30

3.2.1.5 The Battery and Power Management Circuitry

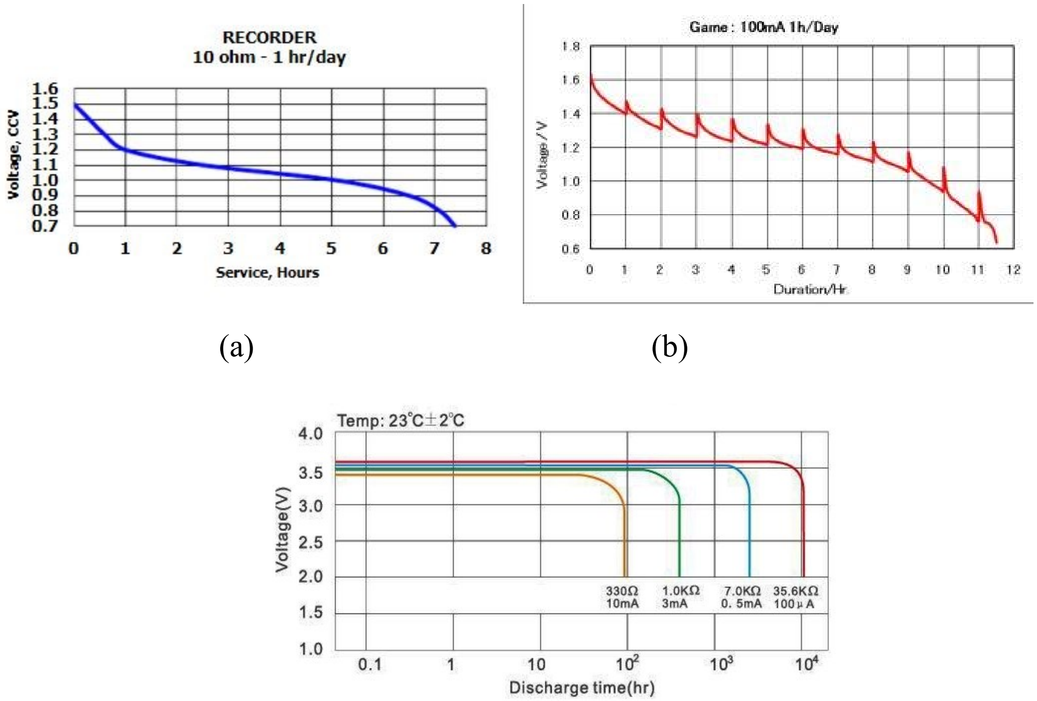
Batteries can be mainly divided into two types: rechargeable and non-rechargeable batteries. Rechargeable batteries can be charged and reused, which decreases the ownership cost and environmental impact in the long run. However, non-rechargeable batteries normally have a higher energy density compared with rechargeable batteries, except the lithium-ion typed batteries. Table 3-5 illustrates the energy density for several most common types of rechargeable and non-rechargeable batteries. Considering the request from Syncrude for a non-rechargeable battery as it is more convenient to use, non-rechargeable batteries were considered. There are many types of non-rechargeable batteries; carbon-zinc, alkaline, and lithium typed batteries are three of the most common types. A 1100mAh carbon-zinc battery (1215), a 1140mAh Alkaline battery (LR03C), and a 1200mAh lithium battery (ER14250) were considered in this study due to their small dimensions, less weight, and capacity. Table 3-6 shows a comparison table of them. Among these three, the lithium battery has a higher nominal voltage, smaller physical dimensions, and less weight. Unlike others, ER14250 has a larger operating temperature range; it can be operated in a harsh environment with an operating temperature between 40°C to 125°C, making it perfect for a harsh environment. Moreover, it was preferred over other non-rechargeable batteries because of its flatter discharge voltage curve. The battery voltage stays reasonably constant throughout the discharge cycle, which means less voltage variation that the design must tolerate. Figure 3-5 shows the discharge voltage curves for a) a 1140mAh alkaline battery (LR03C), b) a 1100mAh carbon-zinc battery (1215), and c) the selected 1200mAh lithium battery. Hence, a non-rechargeable 3.6V, 1200mAh lithium battery, ER14250, was chosen to supply power for the node platform.

Table 3-5 Battery cell comparison [56]

Battery type	Rechargeable	Energy density (Wh/kg)
Lead–acid	Yes	30-50
Nickel-cadmium (NiCd)	Yes	45-80
Nickel-metal hydride (NiMH)	Yes	60-120
Lithium-ion	Yes	110-160
Alkaline	No	85–190
Lithium	No	450-700

Table 3-6 Three non-rechargeable batteries

Battery name	1215	LR03C	**ER14250**
Type	Carbon-zinc	Alkaline	**Lithium**
Nominal voltage (V)	1.5	1.5	**3.6**
Capacity (mAh)	1100	1140	**1200**
Height (mm) Diameter (mm)	49 14.5	44.5 10.5	**25** **14.5**
Weight (g)	15	11	**10**
Operating temperature (℃)	-18 to +55	-20 to +54	**-40 to +125**



(a) (b)

(c)

Figure 3-5 Discharge voltage curves for a) a 1100mAh carbon-zinc battery (1215) [57], b) a 1140mAh Alkaline battery (LR03C) [58], and c) a 1200mAh lithium battery (ER14250) [59]

Since all the selected components operate at different supply voltage ranges, the system was designed to operate in 3.0V. During the circuit design, there are analog and digital circuits, so two 3.0V voltage regulators are required to provide both the power to the digital and analog circuits separately. A CMOS low dropout voltage regulator MCP1700 was selected. A 3.0V output voltage requires an input voltage between 3.2V to 6.0V, which is applicable for 3.6V lithium batteries. On the one hand, this voltage regulator has a low quiescent current (1.6 µA) and high accuracy (0.4% typical output voltage tolerance) of the output voltage. Also, it only requires two 1.0 µF ceramic output capacitors, which can then provide small signal stability with an output current of up to 250 mA. On the other hand, it also features protection for over temperature, over current, and short circuit protection, ensuring a robust solution for the application.

3.2.1.6 Interface between Sensor and ADC

Normally, there are two methods to power the MEMS strain-gauge sensor and measure the resistance change. The first method uses a voltage source with a Wheatstone bridge circuit for each element; the other uses a current source to provide a constant current of 100μA to a sensing element. The following paragraphs compare the design of these two methods.

For a Wheatstone bridge circuit, three extra balancing resistors are needed for each sensing element. Also, each Wheatstone bridge circuit requires its own power supply to avoid interference from other bridge circuits. In order to source 100μA to a 1kΩ sensing element, a large resistor needs to be added in series with the bridge as the equivalent resistance of the Wheatstone bridge is 1k. For a 1.5V power supply, a 14kΩ resistor is needed. This resistor leads to a significant power loss (1.4W). Figure 3-6 shows the Wheatstone bridge circuit for a single sensing element; differential signals are output from the Wheatstone bridge, which requires differential ADC. In total, four extra resistors, one voltage source, and two channels on an ADC to allow differential inputs are necessary. With one-ohm resistance change, i.e., the sensing element changes from 1000Ω to 1001Ω, the voltage difference between the differential signals is approximately 25 μV.

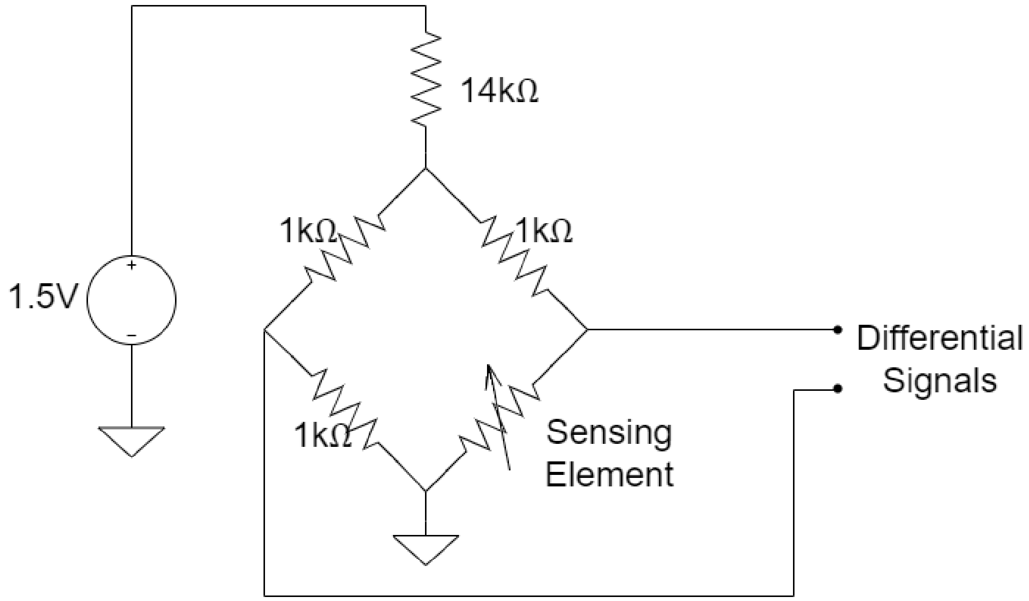


Figure 3-6 Voltage measurement of a single channel

In contrast, the constant current method does not require a large resistance to regulate the current, hence avoiding power loss. No matter how the resistance changes, a constant current source is capable of providing a constant current of 100µA. Besides, it does not require extra balancing resistors in the circuit. Figure 3-7 shows the circuitry for measuring the change in resistance for a single sensing element, representing the change in stress/strain. Only one current source and one voltage source are required for a single sensing element scenario. Moreover, multiple sensing elements can use the same voltage source. As for the sensitivity, a change of 100µV can be observed with a one-ohm resistance change, which is much larger than the 25 µV while using the Wheatstone bridge. Hence, the current source method was chosen mainly because of the higher sensitivity.

Many current source integrated circuits can provide a constant current of 100µA. PSSI2021SAY was selected mainly because of the low cost (CAD 0.68), small physical dimensions (2.1mm x 1.25mm), and small load stability error (0.5%). Only a single 7.25kΩ external resistor is needed to generate a constant current of 100 µA. The calculation is given by Equation (3-4) based on the manufacturer's datasheet. In that case, for a 1Ω resistance change of a sensing element, 100µV voltage change can be observed.

$$I_{out} = \frac{0.617}{7.25k} + 15\mu A = 100.1\mu\text{A} \qquad (3\text{-}4)$$

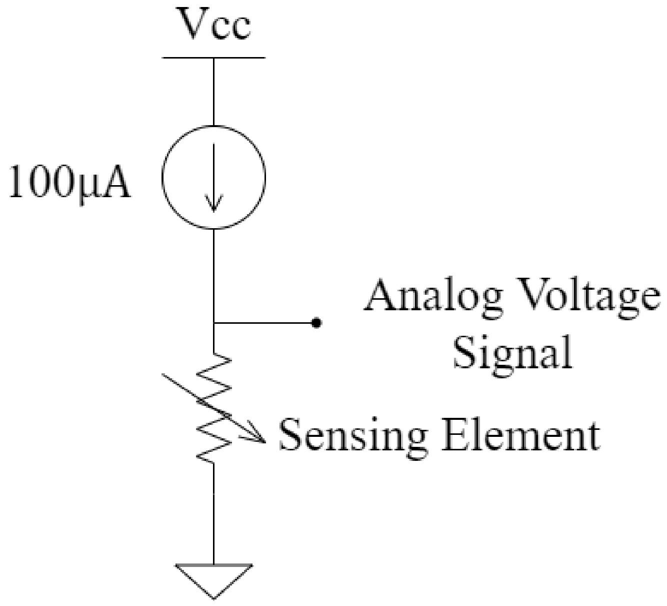


Figure 3-7 Voltage measurement of a single channel

So far, the single-channel scenario has been discussed. Given the limited number of ADC inputs, it is impossible to use a single ADC to handle all 17 channels of the sensor. Two methods can be implemented to solve this problem; one is to add multiplexers (MUX), and the other is to use multiple ADCs. A multiplexer is a device designed to select one of several input lines and forward it to a single output line according to the control signals. The advantages of using the approach with multiplexers are lower current consumption and smaller physical dimensions. The following paragraphs compare the design of these two methods.

Figure 3-8 shows the block diagram of the interface of the MEMS sensor to 3 channels of 16-bit ADC. Two 8-to-1 multiplexers, three current sources, and a single ADC were implemented. An 8-to-1 multiplexer was selected for the purpose of minimizing the number of integrated circuits (ICs), which consequently reduced both the cost, power consumption, and physical dimensions. Two I/O signals from the microcontroller were used as ENABLE1 and ENABLE2 to control the on/off state of the MUXs. Three I/O signals from the microcontroller were used to control ADD_A, ADD_B, and ADD_C on the MUX to decide which input signal was connected to the output. Many multiplexers are available; MAX4781 was selected because of its high-speed switching (25ns for switching on and 15ns for switching off) and low on-resistance (1 ohm). Each MUX only consumes 1µA current at a supply voltage of 3V. Hence, the total current consumption of the analog circuit during acquisition is:

$$\begin{aligned} I_{active} &= 1 \times\ I_{ADC} + 3 \times I_{current\ source} + 2 \times I_{multiplexer} \qquad (3\text{-}5) \\ &= 1.18mA + 3 \times 208\mu A + 2 \times 1\mu A = 1.81mA \end{aligned}$$

Where I_{ADC} is the power supply current of the ADC, $I_{current\ source}$ is the summation of supply current of the current source and the current provides to the sensor, and $I_{multiplexer}$ is the supply current of the multiplexer.

In contrast, if five ADCs are used to interface the 17 channels, 17 current sources are required. The current consumption during data acquisition is:

$$\begin{aligned} I_{active} &= 5 \times I_{ADC} + 17 \times I_{current\ source} \qquad (3\text{-}6) \\ &= 5 \times 1.18mA + 17 \times 208\mu A = 9.44mA \end{aligned}$$

According to the current consumption of the two scenarios, a significant current reduction (81%) occurred by adding two multiplexers. Furthermore, the total number of ICs was reduced from 22 (5 ADC plus 17 current source ICs) to 6 (1 ADC with 2 MUXs plus 3 current source ICs). Finally, 16 of 17 analog channel signals were fed into two multiplexers. The last analog channel signal and the two multiplexers output analog signals were fed into 3 of 4 channels of the ADC. The microcontroller read the ADC digital output signal directly.

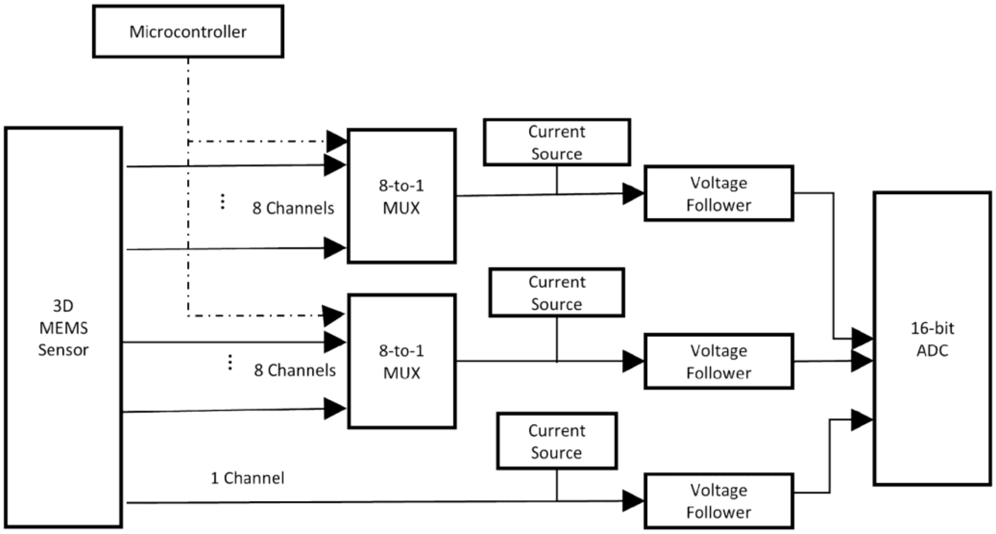


Figure 3-8 Stress and strain sensing block diagram

As mentioned in Section 3.2.1.2, an external 1.5V reference voltage was used to increase the ADC sensitivity. A CMOS low dropout voltage regulator MCP1700T-15 was selected. Three operational amplifiers were used in a voltage follower to provide low impedance signals to the ADC.

3.2.1.7 SPI Interface between the SoC and ADC/Flash

Both the ADC and flash memory communicate with the microcontroller via the SPI interface. Hence, two chip select (CS) signals are necessary to determine which IC is communicated. Figure 3-9 illustrates the SPI interface design between the microcontroller and two SPI peripherals; CSn_ADC is the chip select signal for ADC and CSn_Flash is the chip select signal for flash memory. The ADC and Flash Memory share the other three wires, including MOSI, MISO, and SCK.

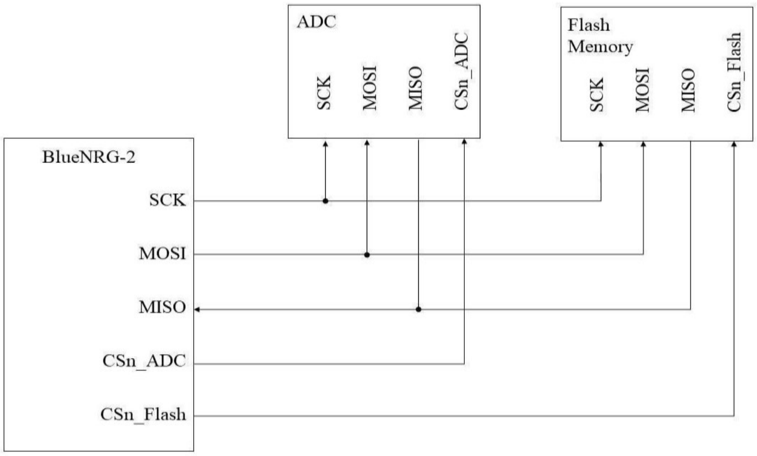


Figure 3-9 The connection between SoC and two SPI peripherals

3.2.2 Router

The router's functionality is to receive and store the stress-strain value from multiple sensing nodes and transmit them to the gateway periodically. Unlike the sensing acquisition platform, the router does not have sensors and ADC. Figure 3-10 shows the block diagram of the router platform. It is only composed of three components: a) a system-on-chip (SoC) which integrates a microcontroller with a built-in BLE radio, b) an external flash memory, and c) a power supply with the power management circuitry. Sections 3.2.1 to 3.2.3 discuss the three components in the block diagram in detail.

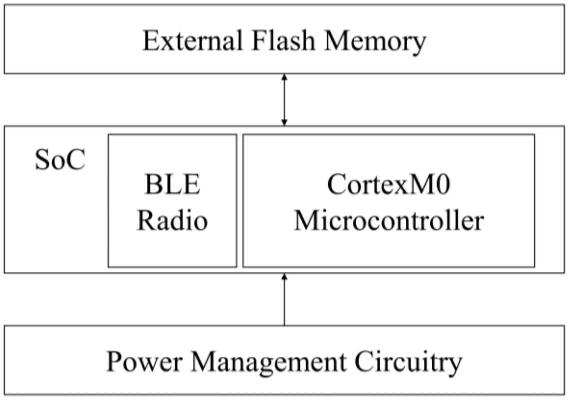


Figure 3-10 Block diagram of the router platform

3.2.2.1 System-on-Chip (SoC)

The SoC of the router is required to have the following characteristics:

- Has an operating temperature from -40 °C to 125 °C to ensure functionality and reliability in a harsh environment,
- Integrates a microcontroller with a BLE radio,
- Contains internal flash memory with at least 150kB for programming code,

and

- Small physical dimensions of approximately 8mm x 8mm.

Based on the discussion in Section 3.2.1.1, the same SoC as the node platform was selected for the router platform. Using the same SoC makes the development easier because the source code is similar and compatible.

3.2.2.2 External Flash Memory

Since we may need to store multiple node data before transmitting them to the gateway, a large external flash memory is necessary for the router. As mentioned before, 3.7Mbytes data is stored per hour at a sampling frequency of 20Hz. Thereupon, if each router receives data from five nodes, then 18.5Mbytes data are stored per hour for each router. Equation (3-7) demonstrates that a 1G bytes flash memory can store at least 2-days of data while connecting with five nodes.

$$\frac{1Gbytes}{18.5Mbytes/hour} = 2.27days \tag{3-7}$$

There are a few 1Gbytes external flash memory available for selection. Two of them, MX60LF8G28AD-XKI and AS5F38G04SND, were considered since they supported the SPI interface to communicate with the microcontroller and could operate at 3V. As shown in Table 3-7, they have the same max page program time (700µs), and max current consumption for reading, programming, and erasing operation are all the same (30mA). In the end, AS5F38G04SND was selected primarily because of the smaller physical dimensions. It consumes a maximum of 120µA in standby mode. Although the maximum supported SPI clock frequencies are different, the microcontroller can only handle up to 8 MHz, achievable for all three memories.

Table 3-7 Two 1Gbytes non-volatile external flash memories

Part number	MX60LF8G28AD-XKI	**AS5F38G04SND**
Package (mm x mm)	63-VFBGA (9x11)	**8-LGA(6x8)**
Page program time (μs)	700	**700**
Block erase time (ms)	4	**3**
Supply voltage (V)	2.7 - 3.6	**3.0V - 3.6**
Read current (mA)	30	**30**
Program current (mA)	30	**30**
Erase current (mA)	30	**30**

3.2.2.3 The Power Supply and Power Management Circuitry

Either a DC transformer or a larger lithium battery can be used to supply power for the router, making sure the router can last longer than the sensing node. Moreover, the system also operated at 3.0V, and the same voltage regulator MCP1700 was implemented to regulate battery voltage to 3V. Unlike the node board, the power management circuitry of the router board was composed of only one 3.0V voltage regulator.

3.2.3 Gateway

Gateway is a hardware device that interconnects and transfers data across different networks. In brief, it acts as a router and enables communication between multiple protocols. This research used a gateway to receive BLE data and upload data to the cloud via Wi-Fi. The design requirements of the sensing node are listed:

- It is compliant with both BLE and Wi-Fi.
- It should have a data storage space larger than 1Gbytes so that it can still store all received data from at least one router when some emergency happens, such as a lost Internet connection.

- For easy development, a display port and two USB ports are required to hook up to a monitor, mouse, and keyboard.

There are a few Bluetooth Low Energy to Wi-Fi connectivity gateways in the market. However, they usually do not have external flash memory larger than 1Gbytes, which means received data might be lost once the Wi-Fi is lost. Moreover, some of them are not compatible with all BLE devices. SGW6008 only works with specific BLE devices, such as SGW111X BLE Module and SGW8130 BLE Sensor Tag. Hence, three single board computers, Raspberry pi 4B+, UPC-CHT01-A20-0464-A11, and CC-SBP-WMX-JN58, were considered to be used as a gateway. As shown in Table 3-8, all three devices have more than two USB ports and a display port so that a screen, mouse, and keyboard can be attached for data browsing and functionality debugging. Furthermore, they all have more than 1GB of data storage space to store data, preventing data loss if Wi-Fi is momentarily lost. Among all three devices, UPC-CHT01-A20-0464-A11 has the smallest physical dimensions. However, it only supports BLE4.2, while the other two are compatible with BLE 5.0. Furthermore, Raspberry pi 4B+ and CC-SBP-WMX-JN58 also support ethernet.

Table 3-8 Three gateway devices

	Raspberry pi 4B+	UPC-CHT01-A20-0464-A11	CC-SBP-WMX-JN58
Company	**Raspberry Pi**	AAEON UP	DIGI
Price (CAD)	**129.38**	289.8	444.5
Processor	**Quad-core Cortex-A72 (ARM v8) 64-bit SoC**	Intel® Atom™ x5-Z8350 Processor SoC	NXP i.MX6UL-2, ARM® Cortex®-A7, with NEON™ MPE (Media Processor Engine) co-processor
Data storage	**SD card up to 64GB**	eMMC up to 64GB	microSD; on-board 4 GB eMMC flash
BLE	**Yes**	Yes	Yes
BLE version	**5.0**	4.2	5.0
Wi-Fi	**Yes**	Yes	Yes
Ethernet	**Yes**	No	Yes
USB	**USB 3.0 x2** **USB 2.0 x2**	USB 3.0 x 1 USB 2.0 x 2	USB 2.0 x2 USB OTG x1
Display	**micro-HDMI x2**	HDMI x 1	24-bit Parallel RGB; 18-bit LVDS
Dimensions (mm)	**85 x 56**	56.5 × 66	100 x 72

Finally, Raspberry pi 4B, as shown in Figure 3-11, a small single-board computer (SBC), embedded with a Quad-core Cortex-A72 (ARM v8) 64-bit SoC, was chosen as a gateway. The main reasons are the higher BLE version, low cost, and high integration. There are a series of external modules available to make it easy to use. A micro-SD card slot is included for loading operating systems and data storage. When there is no Internet, the 64GB SD card may store all data while waiting for the internet or read using a Linux computer. In summary, developers can rapidly and cost-effectively evaluate the gateway using raspberry pi and easily extend the gateway to support diverse connectivity options.

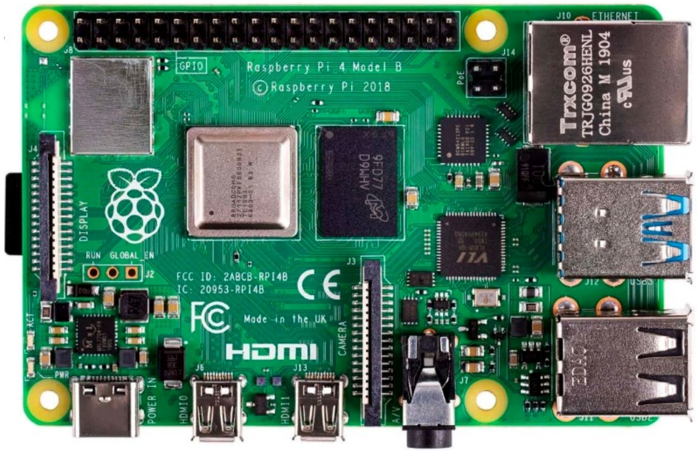


Figure 3-11 Raspberry Pi 4B [60]

3.3 Summary

All the components' selections have been made, and the design was done. After the hardware, the software needs to be developed. The next chapter describes the development of the software.

Chapter 4. Software Architecture

4.1 Overview

This chapter provides the details of the software design of the structural health monitoring system. Three firmware: a) the sensing node, b) the router, and c) the gateway are described and explained in detail. Prior to reporting the firmware, the roles of BLE inside each firmware are illustrated in Section 4.2. Then Sections 4.3, 4.5, and 4.6 report the firmware for the node platform, router platform, and gateway. Section 4.4 presents the Android user interface to interface with the node and router platforms for functional validation. A summary is supplied in Section 4.7.

4.2 BLE Communication Roles

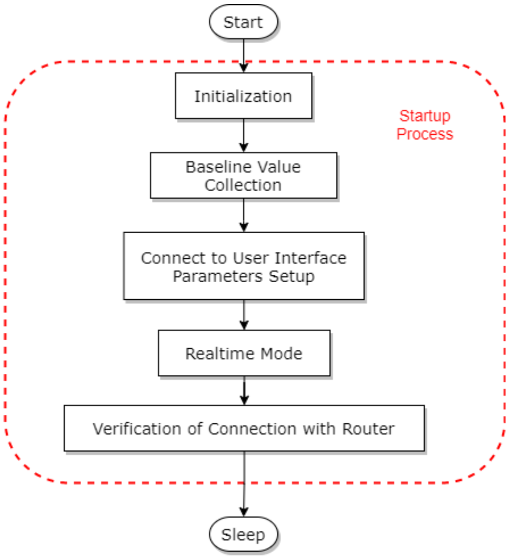
Generic Access Profile (GAP) roles and Generic Attribute Profile (GATT) roles are two essential factors in BLE communications.

The Generic Access Profile (GAP) layer handles various tasks, including device discovery and establishing a connection. The GAP role controls how the BLE radio connection is managed. A GAP central device can initiate an outgoing connection request to a peripheral advertising device, while a peripheral device is a BLE device that accepts the incoming connection request after advertising. In my research, to establish a connection between a sensing node and an Android device or a router, the sensing node always acts as the peripheral role, while the Android device and router assume the central role. On the other hand, the gateway or an Android device always functions as a central device to initiate a connection. A router plays different roles while connecting with the node platform and Android device or gateway. The router acts as a GAP peripheral when data is exchanged between the router and the gateway/an Android device. The router switches to the GAP central role while attempting to interact with a node device.

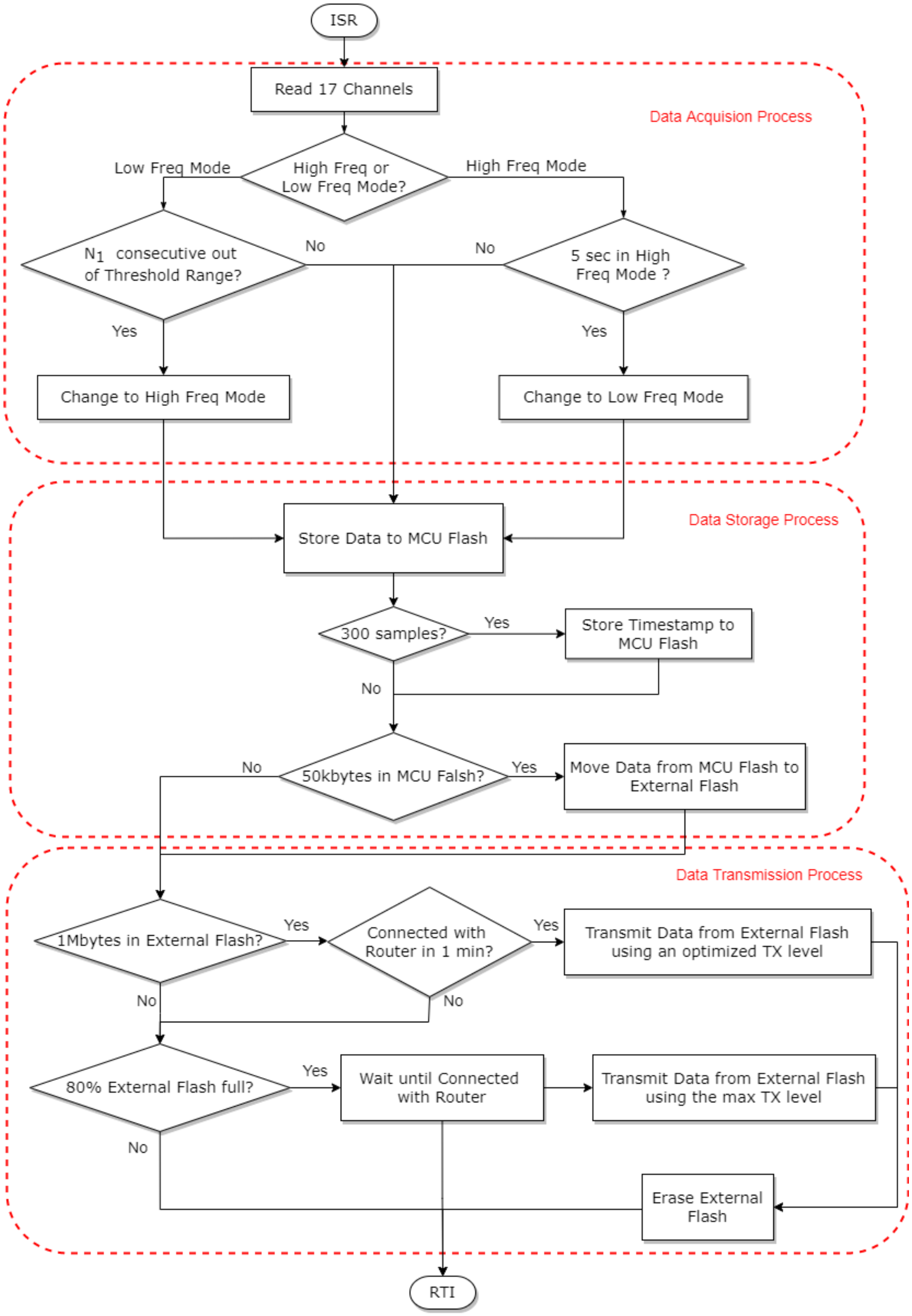
Generic Attribute Profile (GATT) is another crucial layer in BLE communication. It allows the device to communicate its attributes, including the characteristic values, to another device. A GATT server device has a structure called a GATT table, which contains special registers called characteristic values, which a client device can then read. Likewise, a GATT client device can send information to a GATT server by writing a value into one of the server's characteristic values. Although these roles are completely independent of the GAP central/peripheral relationship, my research initializes a GAP central device as a GATT client device.

4.3 Sensing Node Firmware

Figure 4-1 shows the flowcharts of the firmware inside the node. The startup process was performed first, as shown in Figure 4-1(a). Section 4.3.1 describes the details about the startup process. Then, the node would enter the interrupt subroutine periodically based on the sampling frequency. As shown in Figure 4-1(b), the interrupt subroutine comprises the data acquisition process, data storage process, and data transmission process using an optimized TX power level.


Start
Initialization
Startup Process
Baseline Value Collection
Connect to User Interface Parameters Setup
Realtime Mode
Verification of Connection with Router
Sleep

(a)


ISR
Read 17 Channels
Data Acquision Process
High Freq or Low Freq Mode?
Low Freq Mode
High Freq Mode
N1 consecutive out of Threshold Range?
No
No
5 sec in High Freq Mode ?
Yes
Yes
Change to High Freq Mode
Change to Low Freq Mode
Data Storage Process
Store Data to MCU Flash
300 samples?
Yes
Store Timestamp to MCU Flash
No
50kbytes in MCU Falsh?
No
Yes
Move Data from MCU Flash to External Flash
Data Transmission Process
1Mbytes in External Flash?
Yes
Connected with Router in 1 min?
Yes
Transmit Data from External Flash using an optimized TX level
No
No
80% External Flash full?
Yes
Wait until Connected with Router
Transmit Data from External Flash using the max TX level
No
Erase External Flash
RTI

(b)

Figure 4-1 The flowcharts of the firmware inside the node, (a) the startup process, and (b) the interrupt subroutine

Note: ISR – Interrupt subroutine; RTI – return from interrupt;

4.3.1 The Startup Process

At the beginning of the startup process, the node performed the initializations of all the hardware configurations, including the GPIO, clocks, and BLE stacks configurations. Then, baseline values of each sensing element were collected. The system scanned through all 17 channels and acquired data for two seconds with a sampling frequency of 5 Hz. Approximately ten samples were collected for each channel at that 2s. The averages of these samples were considered as the baseline values. During the initialization, the ADC was also programmed into a scan mode in which three channels were scanned sequentially on the ADC without requiring MCU communications. Two of the channels in ADC connect with the MUX outputs. Based on the select signals of the 8-to-1 MUX, a specific channel was chosen at a time.

Initial parameters setting was performed via a custom user interface developed in an Android platform. By using this interface, the sensing node synchronized the date and time with the Android device. A timestamp was added for time recording. The user interface also allowed users to verify that the sensing node and the MEMS sensor functioned properly. When a real-time mode command was received from the user interface program, the resistance values of the 17 elements were acquired and transmitted to plot a real-time graph. The details of the user interface are described in Section 4.4.

Once the sensing node functionality was confirmed, it would offer a handshake to the router using the maximum transmit (TX) power. The handshake was achieved by making the node discoverable so that a router could find it while scanning. A connection initiated by the router would terminate the discoverable mode. Using the maximum power was to make sure the node was able to communicate with the router. The blue LED indicated a successful connection on the sensing node. If the connection failed due to the distance range, the operator needed to reduce the distance between the sensor node and the router. Ten seconds after the successful connection, the sensing node would disconnect with the router, enter into a long-term sampling mode, and wait for the timer to generate an interrupt to wake up the MCU.

4.3.2 Data Acquisition Process

In each interrupt subroutine, the node scanned through all 17 channels, acquired data, calculated the corresponding resistance values, and went to the data storage process. The interrupt occurred when a timer timeout event occurred while the timer was controlled by the sampling frequency. In the sensing node, two sampling frequencies were implemented: 5 Hz and 20 Hz. The high sampling rate (20Hz) would allow detecting any fast impact. However, for a system utilized for long-term monitoring, if the sampling rate was maintained high, the amount of collected data would be four times more than the normal rate, requiring a larger memory capacity and a longer transmitting time. Hence the power consumption increased. In order to optimize the memory space and power consumption, a dynamic sampling approach was implemented under the software control.

Figure 4-2 shows the dynamic sampling approach at two frequencies exchange during data acquisition. Each pulse represents the 17 channels' measurements; the gray shaded pulse means at least one channel of the measurements was out of the threshold range. At the normal sample rate, the sampling frequency was 5 Hz. When N_1 (a programmable variable) of consecutive changes were not within the threshold (programmable as well) of stress or strain range, the sampling frequency would switch to 20 Hz for 5 seconds. If a long-term change occurs, 5Hz is sufficient to detect fast impact change, reducing the data size consequently. After 5 seconds, the sampling frequency was set back to the normal rate. The dynamic sampling approach was considered optimized because the system was capable of achieving an average sampling rate from 5Hz to 20Hz, which met the requirements from Syncrude Canada (maximum 20Hz). Furthermore, the sampling frequency was adjusted automatically without any user interactions.

The threshold range of each channel was calculated based on the input percentage from the user using the Android app and the baseline value acquired in the startup process. Equation (4-1) shows the threshold range defined by minimum and maximum values. The percentage value was 20% unless the user set them during the communication with the user interface. The threshold range was only re-calculated when the percentage was changed while communicating with the user interface (the android device).

$$\text{Threshold Range} = [\text{min}, \text{max}]$$
$$= [\text{baseline value} \times (1\text{-percentage}), \text{baseline value} \times (1\text{+percentage})] \quad (4\text{-}1)$$

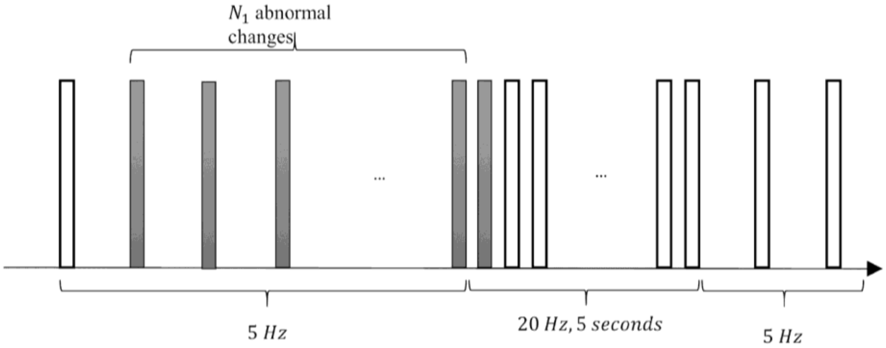


Figure 4-2 Optimized data acquisition system

In order to further save the current consumption, the system was put into sleep mode when no data was collected, i.e., at the end of each interrupt subroutine. Regarding the sleep mode implementation, all integrated circuits which were idle would be put into low power mode. The ADC entered partial shutdown mode, and the MCU entered sleep mode while the data was not collected, i.e., between two pulses. The only wake-up source was a timer determined by the sampling frequency.

4.3.3 Data Storage Process

Due to a limited flash memory space in the MCU, only a portion of the internal MCU memory could provide temporary storage for data collection. As illustrated in the flowchart, the collected data were stored in the MCU flash memory first. A timestamp was also stored in the MCU flash memory for every 300 samples per channel. When 50kBytes of data were filled, data were moved from the MCU flash memory to the external 32Mbytes flash memory. The MCU kept reading data in MCU flash and wrote them in the external flash until the last byte was programed. This approach would reduce the power consumption of the external 32Mbytes memory because it could be put in a standby mode while it was idle. This method could also minimize the chance of data collision on the SPI bus between the MCU and the ADC and the external memory. The MCU flash was erased after the data transfer. When 1Mbytes of data was stored in the external 32Mbytes flash memory, the node started the data transmission process. Otherwise, the node entered sleep directly.

4.3.4 Data Transmission Process

When the node tried to connect with a router to transmit the data, it would be set to discoverable by a router device and wait for a connection establishment in the next one minute. If the attempt failed, it would return to check for external flash capacity and wait for the next interrupt routine. If the connection succeeded, an optimized TX power level would be used for data transmission. The TX optimization process is discussed in Section 4.3.5. This TX optimization approach was designed for a multiple node system to prevent data transmission loss or wait long to upload data to a router. Otherwise, if the connection failed continuously within 1 minute and 80% of the external flash was full, the node would change to the maximum transmit power. Then, the sensing node would continue waiting until a successful connection between the router and upload all the data to minimize data loss.

Once connected, the node kept reading and sending all the external flash data before going back to sleep and waiting for the next interrupt routine. Data with a fixed length of 243 bytes were sent per package each time, while 240 bytes were actual data, two bytes were node data package indication, and the last byte was the node number. An unsuccessful data transmission due to a temporary lack of resources, such as packet pool or timer, would lead to retransmission. When all data in the external flash were transmitted, the node sent 230 bytes to the router, indicating the last byte transmitted. Then, the system went back to sleep mode. If a loss of connection occurred during data transmission, the node would try to connect with the router again and continue sending from the latest successful data package address.

4.3.5 Transmission (TX) Power between Node and Router Optimization

Regarding the transmission power to control the communication range, eight different TX power levels could be selected and set by the MCU inside the sensing node:

-14dBm, -11dBm, -8dBm, -5dBm, -2dBm, 2dBm, 4dBm, and 8dBm. The dBm is the decibels per milliwatt (dBm) of the TX power at which a node transmits its signal.

In order to optimize the TX power level, the sensing node started with the minimal TX power to send data to the router. The node sent five bytes of the TX optimization request command, including the TX power level set at that time, to the router and waited for ACK feedback. If no ACK was received within 10 seconds, a higher TX power level was set and sent to the router. The above procedures were repeated until a successful command exchange or reaching the maximal TX power level. Instead of always using the maximum TX power level, the above optimization process was designed to pick a suitable TX power level optimization to decrease the current consumption without compromising the BLE transmission distance. Figure 4-3 shows an example of the TX optimization process when -8dBm is the optimal TX power level.

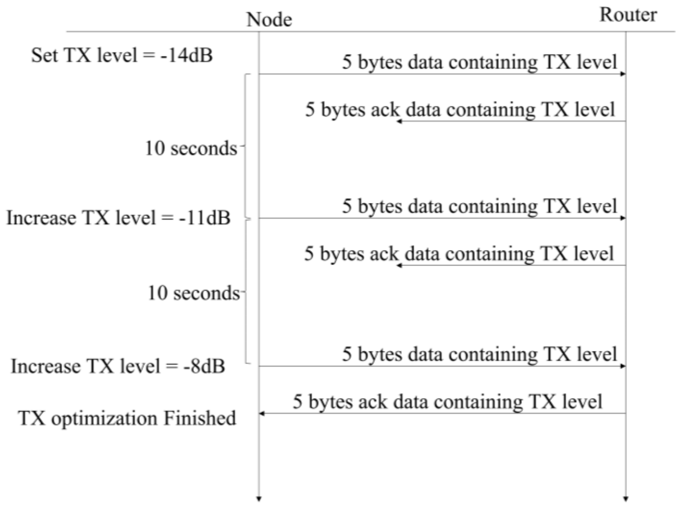
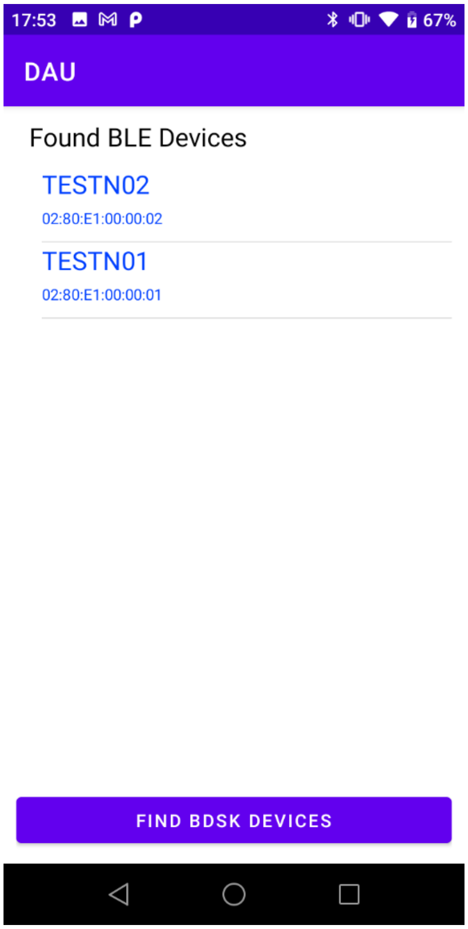

Node
Router
Set TX level = -14dB
5 bytes data containing TX level
5 bytes ack data containing TX level
10 seconds
Increase TX level = -11dB
5 bytes data containing TX level
5 bytes ack data containing TX level
10 seconds
Increase TX level = -8dB
5 bytes data containing TX level
5 bytes ack data containing TX level
TX optimization Finished

Figure 4-3 TX power optimization process

4.4 Android Graphical User Interface Program

An Android application, a graphical user interface program, was developed using the Android Studio, the unified, integrated development environment (IDE) for the Android operating system. The purpose of the Android application was to communicate with both sensing nodes and routers to confirm they function properly and set up user input parameters.

Once the Android application was executed with the BLE enabled on the Android device, the scanning function could be activated using the only key labeled FIND DEVICES. When the Android device was scanning for nearby advertising Bluetooth devices, this key changed to STOP SCANNING, which allowed the user to stop the scanning procedure. At the end of each scanning procedure, a filter blocked all other BLE devices, and only the sensing nodes and routers would show up, as shown in Figure 4-4. The names of the nodes were in the format of TESTN##, while the N indicated node and ## was the node number; similarly, the names of the routers had a format of TESTR##, while the R indicated router. The peer addresses were also displayed beneath. Users could select one of the found devices to communicate.



Found BLE Devices

TESTN02
02:80:E1:00:00:02

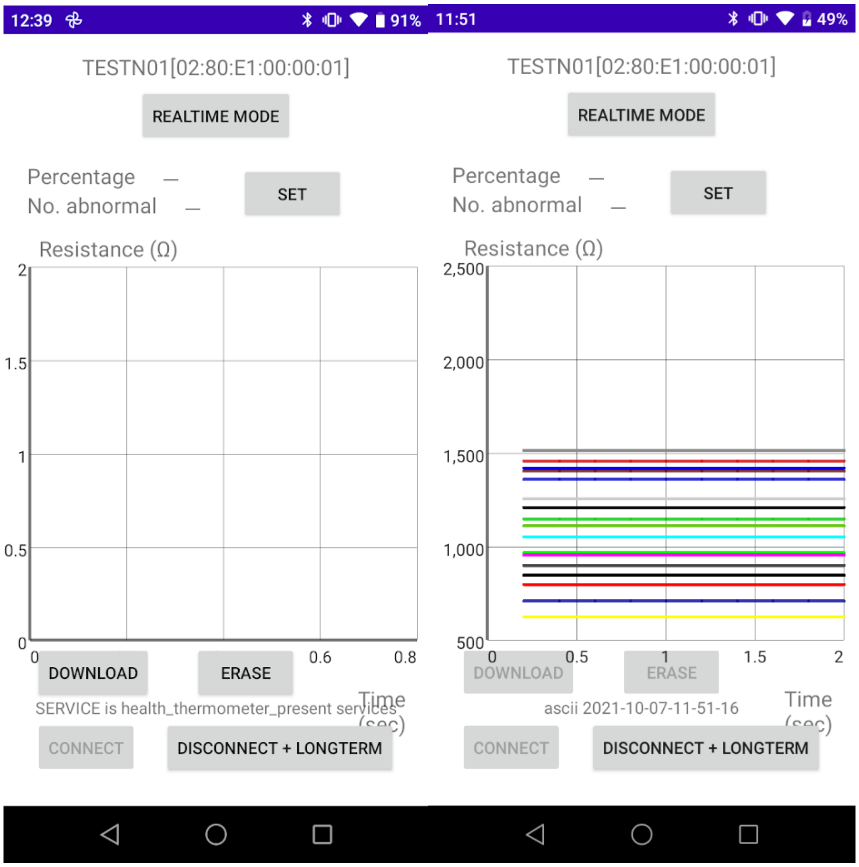
TESTN01
02:80:E1:00:00:01

Figure 4-4 Scan result page of the graphical user interface

After choosing the device, the communication page showed up to establish a connection. The only enabled button when the user first entered this page was the CONNECT. This command caused the user interface to initiate a connection, and a notification was automatically enabled for data transmission. Figure 4-5 shows the communication page after a successful connection establishment. Name and address were also displayed on the top, and other commands were enabled after a successful connection. As shown in Figure 4-5, five other functions could be selected: a) real-time mode, b) setting the N_1 for dynamic sampling and the percentage of the threshold, c) downloading data, d) erasing the data in the flash memory, and e) disconnecting the device from the Android and setting the sensing node in a long-term sample mode.

Functions b to d are optional, and the functionality of one of them does not affect others. As discussed in Section 4.3.2, to achieve dynamic sampling, two parameters are required: the number of consecutive abnormal changes detected to enter the high-sampling mode, N_1, and the percentage used to calculate the threshold ranges based on the baseline values. The SET button allowed the user to customize those two parameters; the android device sent the value entered at the corresponding space to the node. The node would re-calculate the threshold range after receiving them. 10 and 20% were used in default if the SET button was not clicked. Apart from this, as discussed in Section 4.3.4, the node would not send collected data to the router immediately after each data sampling. In that case, some data may be left in onboard memory if the node stops working before sending it out. For this reason, download and erase functions were developed for the remaining data in onboard memory. The downloaded data were stored in the memory of the Android device.

On the contrary, once entered the real-time mode, the only available function was disconnecting the device from the Android and setting the sensing node in a long-term sample mode. In real-time mode, the Android device received the 17 channels' measurements and displayed them on the graph immediately. Figure 4-5(Right) shows the 17 channels data display in real-time. The y-axis is the resistance in **Ω**, and the x-axis is the time in seconds. The actual date and time were sent to the node to initiate the real-time mode. The date and time were consistent with that of the android phone and in the order of Month-Day-Hour-Minute-Second. In this way, once disconnected, the node entered long-term sample mode.



SERVICE is health_thermometer_present services

ascii 2021-10-07-11-51-16

Figure 4-5 Communication page of the graphical user interface when just connected (Left) and in real-time mode (Right)

4.5 Router Firmware

The firmware of the router platform was stored inside the BlueNRG-232 SoC. The main tasks of the router platform are to respond to the node platform data received and transmit them to the gateway via BLE. The flowchart for the firmware inside the router is presented in Figure 4-6. Like the node, the router performed the startup process at the beginning. Section 4.5.1 describes the details about the startup process. Then, the router would check the external flash memory to see whether it has data to be transmitted to the gateway. If yes, the router went to the send routine indicated in Figure 4-6. Otherwise, the router went to the receive routine indicated in Figure 4-6. Section 4.5.2 describes the details about the send routine, and Section 4.5.3 discusses the receive routine.

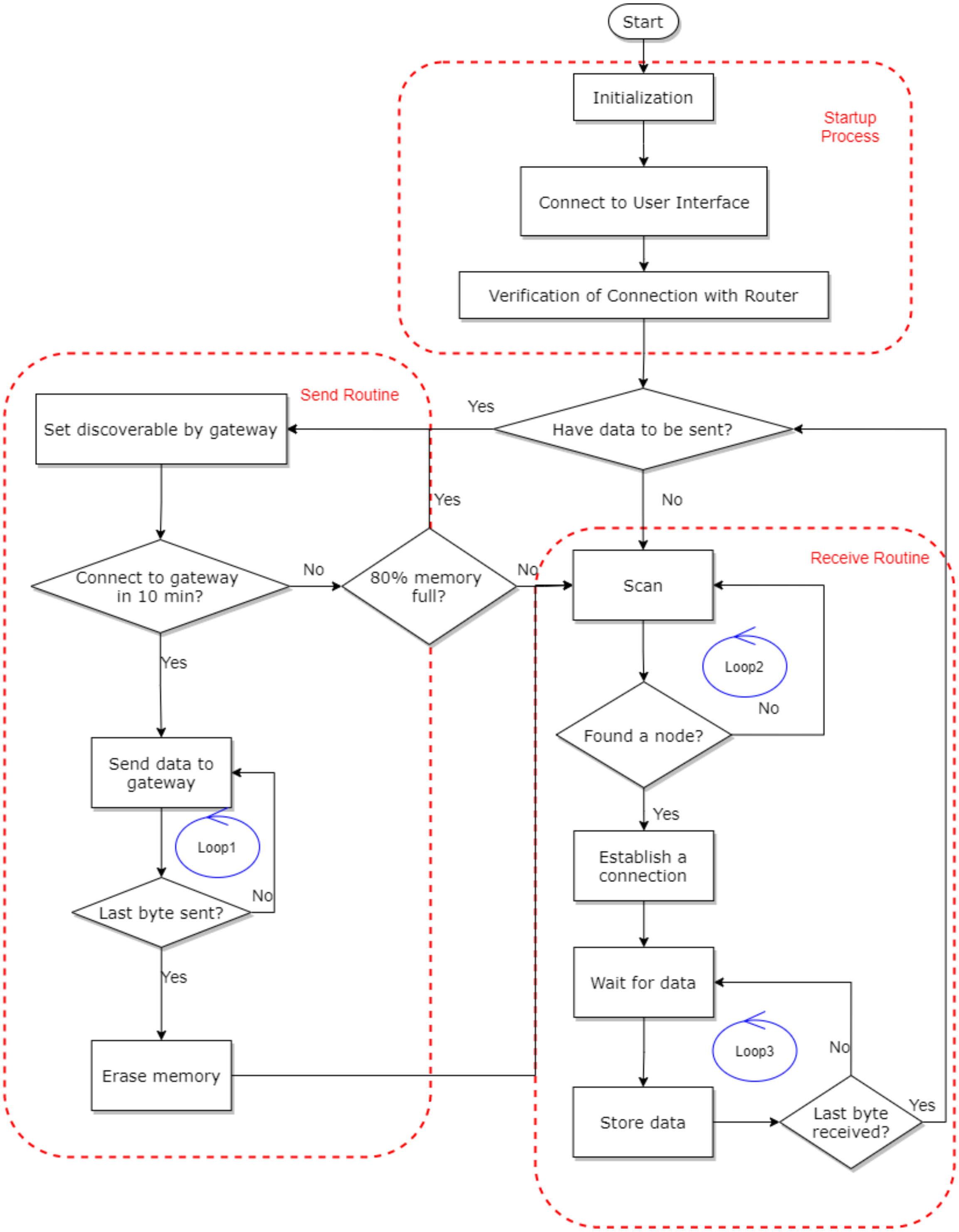


Figure 4-6 The flowchart of the firmware inside the router

4.5.1 The Startup Process

During the startup process, the SoC first performed the initializations of all hardware configurations, including the GPIO, clocks, and BLE stacks setups. Then, the router would also try to connect with the user interface discussed in Section 4.4 to verify that the wireless circuitry functioned properly. Once confirmed its functionality, it would offer a handshake to the gateway. The handshake was achieved by setting the router discoverable and waiting for a connection initiated by the gateway. This action aimed to make sure the router was in the transmission range of the gateway. A blue LED indicated a successful connection on the router. If the connection failed due to the distance range, the operator needed to reduce the distance between the router and the gateway. Followed by 10 seconds on LED indicating a successful connection, the router would disconnect with the gateway.

4.5.2 The Send Routine

In the send routine, the router would check the external flash memory to see whether it had data to be forwarded to the gateway. If yes, it would be made discoverable by the gateway and then waited 10 minutes for the gateway to start a connection. The local advertising name was set to ROUT### to differentiate it from other BLE devices, while ROUT indicated router and ### was the router number. After a successful connection establishment, the router entered Loop 1, where it continuously read data from external memory and sent out data. After the transmission of the last byte of data, the route erased the external flash memory and started scanning for a node device. Otherwise, if the connection failed continuously within 10 minutes and more than 20% of the flash memory was empty, the router would enter the receive routine and start scanning for a node device and receiving data. This approach was implemented in case a node stopped data sampling due to full flash memory. However, if 80% of the flash memory was full, the router was set to discoverable by gateway until a successful connection.

4.5.3 The Receive Routine

If the router had no data to be transmitted, it entered Loop 2, where it continuously scanned for advertising BLE devices. A series of devices might be discovered, and a list of discovered devices was created, but a connection was initiated only if the discovered advertising name was consistent with an expected node name, which had a format of NODE###. Although there might be multiple nodes advertising simultaneously, the router only connected with the earliest discovered node device, i.e., the top node device on the discovered list. An automatic maximum transmission unit (MTU) config exchange was performed to reach an agreement on maximum data length with the node. Then, the router entered Loop 3, where it kept waiting for data, which has a length of 243 bytes per package. The received data were stored in the external memory. When the router received 230 bytes per package, it knew that the last byte was received. Following this, it went back to recheck flash memory capacity and started a new cycle.

4.6 Gateway Firmware

Unlike the sensing node and the router, the gateway did not transmit data to another BLE device, but it had to upload them to Google Drive. In brief, its functionality is to continuously scan for router devices and upload received data to the Drive. Raspberry Pi OS was downloaded as the operating system. The pip command was used to install the Bluepy package, which provides an application programming interface (API) for accessing BLE devices from Python. Bluepy is a python module that allows communication with BLE devices. PyDrive was downloaded using the pip command as well to manage files in Google Drive. It is a wrapper library for the Google Drive API; it allows the user to easily upload, download and delete files in the Drive from a Python script. A client_secrets.json file was downloaded from the Google API Console, which contains the secret key to access the target Google Drive.

The flowchart for the router firmware is shown in Figure 4-7. Initialization was performed first, including Google Drive authorization and folder creation. Followed by the initialization, the gateway started a loop to scan for advertising BLE devices. A connection was initiated when a router device was found. Then it continuously waited for data. In order to avoid data loss, 2-way communication was implemented with routers. When data were received, an acknowledgment was sent to the router asking for the next data package. Then, the received data were stored in a local CSV file, in which each line is one data package. Scan and connection functions were initiated again if the connection was lost in between. Then, a data organization process was performed after the last byte was received. The organized file should have only 18 or 6 data per line, while the 18 data represent the node number and the resistances of 17 channels and the 6 data represent the node number and the date and time. The data and time are in Month, Date, Hour, Minute, and Second format. Then, the organized file was uploaded to Google Drive. The uploaded file name is the uploading date and time.

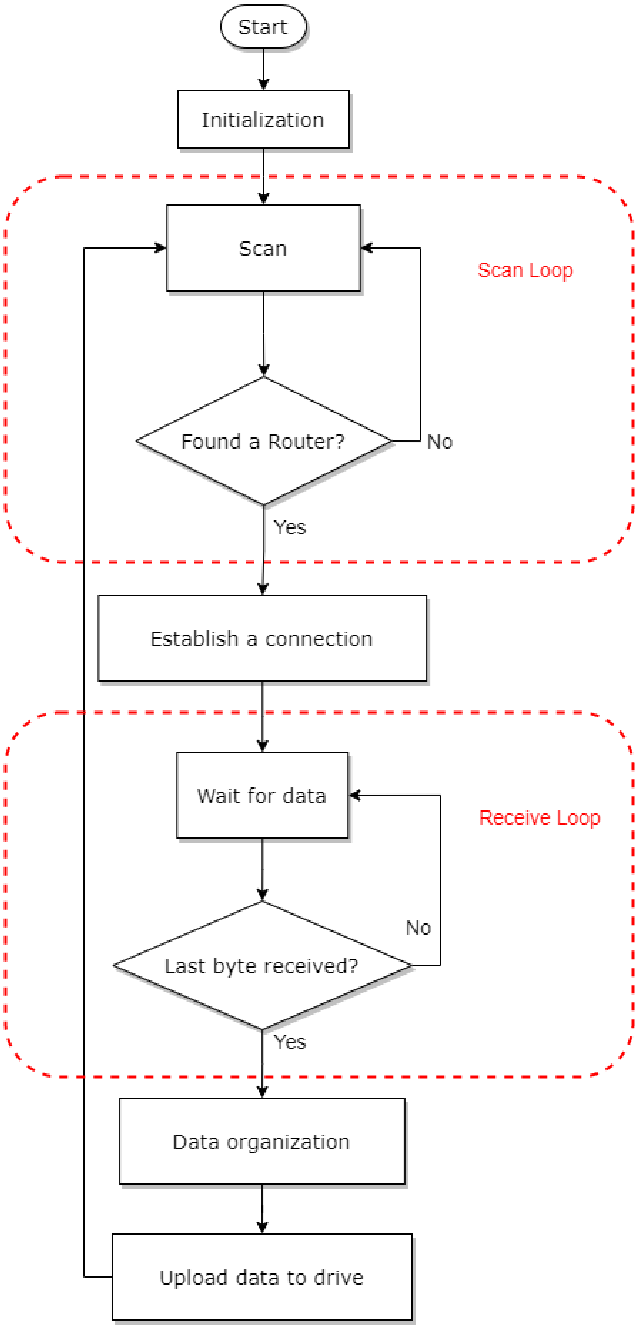


Figure 4-7 The flowchart of the firmware inside the gateway

4.7 Summary

All the four firmware have been developed, the prototype and the experimental validation will be described in chapter 5.

Chapter 5. Prototype Validation and Test Results

5.1 Overview

Custom printed circuit boards (PCBs) have been designed and built for the data acquisition board of the sensing node and the router board. A series of experimental studies were conducted to validate their functionality and accuracy. Section 5.2 describes the design details of the PCBs. Sections 5.3 to 5.7 report the experimental setups and procedures to evaluate the individual platform and the entire integrated system; results are also presented. A summary of the experimental results is provided in Section 5.8.

5.2 Printed Circuit Board Design and Prototypes

As mentioned in chapter 3, the acquisition board of the sensing node is composed of analog and digital circuits. Since the digital circuitry generates more noise and the analog circuitry is quite vulnerable to noise, separated power and ground rails are required to prevent digital noise from corrupting analog performance. Figure 5-1 shows the two-layer printed circuit board for the data acquisition board. The red area indicates the top layer, and the blue one indicates the bottom layer. There are big ground planes on both sides (the red and blue areas other than ICs and traces); The left side ground planes are the analog ground (AGND) planes, and the right ones are the digital ground (DGND) planes. A small black line in the middle indicates a region without any ground planes. A zero-ohm resistor, indicated by the yellow box in Figure 5-1, separated the analog and digital ground planes. As for the separated power supply, it was achieved by using two voltage regulators (U8 and U9).

The U1 is the SoC, BlueNRG-232, and the green box indicates the RF circuitry. E1 is the antenna, one of the major components of the RF design. A 2.45GHz wideband, small form factor SMD chip antenna, 2450AT43F0100 (2.1dBi), was used for the data acquisition board due to the small physical dimensions (6mm x 2mm x 1.2mm). The design of the antenna circuitry used the recommended circuitry based on the design of the evaluation board (STEVAL-IDB008V2). An ultra-miniature balun (1.4mm x 0.85mm), BALF-NRG-02D3, which provided the proper LC (inductor-capacitor) network for the RF circuitry, was chosen. The matching impedance was customized for the BlueNRG transceiver. This balun also integrates a harmonics filter for enhanced RF performance. For the SoC, two crystals were used to provide both the primary high-frequency and low-frequency clocks, indicated by the green boxes in Figure 5-1. Since the RF ground is noisy, the ground of the two external crystals must be isolated from the RF section of the board. This design ensures both crystals function properly and reduces noise interruption. CON2 and CON3 on the left edge of the circuit board are the connectors interface with the sensing board, CON1 on the top right corner is the programming connector.

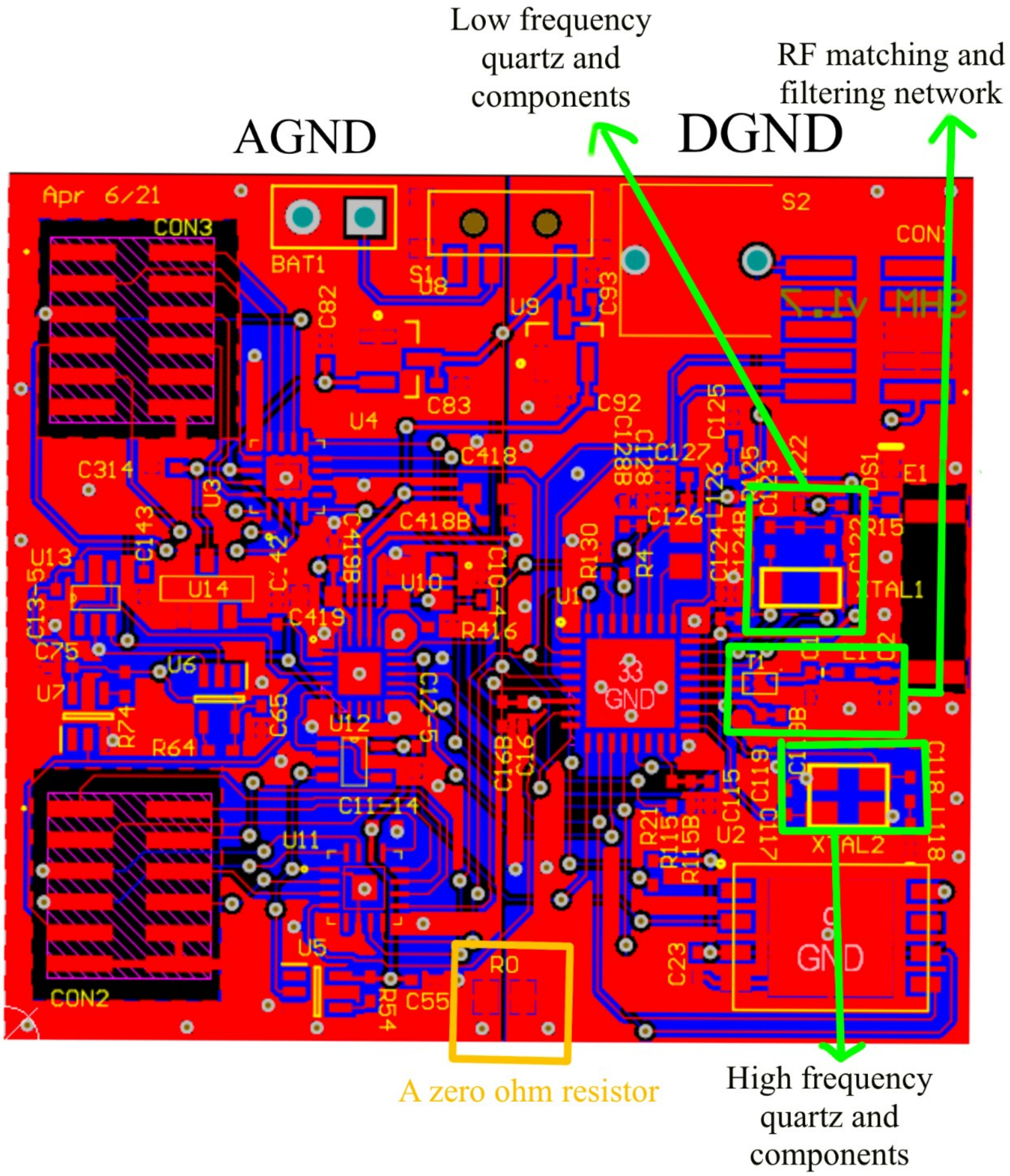


Figure 5-1 PCB design of the acquisition board

Figure 5-2 shows the prototype of the data acquisition board (35mm x 38mm). The components in the red box are the SoC and the BLE radio circuitry, while the green and yellow rectangles are the ADC and flash memory, respectively. The circuitry in the blue rectangle is the interface circuit between the sensor connectors, MUX, Op-Amp, and ADC. Figure 5-3 shows the top and bottom views of the prototype of the sensing board (20mm x 20mm). The arrows indicate the locations of the sensor and two interface connectors to communicate 17 channels. Figure 5-4 shows the connection between the data acquisition board and the sensor board via two flex cables.

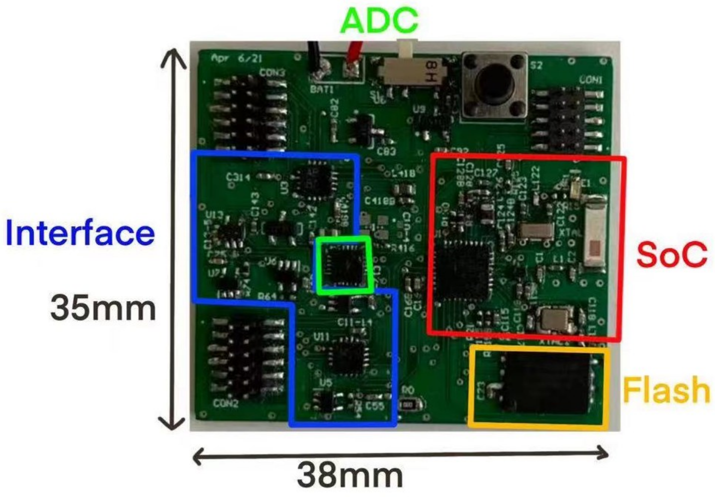


Figure 5-2 The prototypes of the data acquisition board

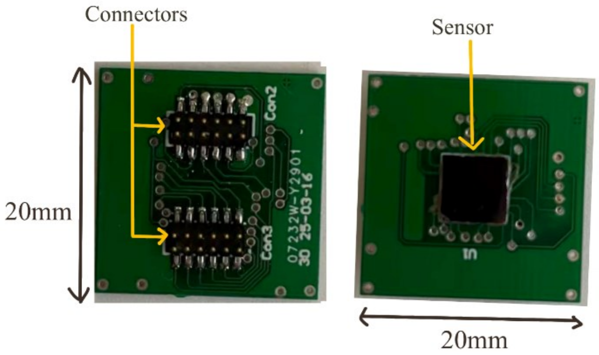


Figure 5-3 The prototypes of the sensing board

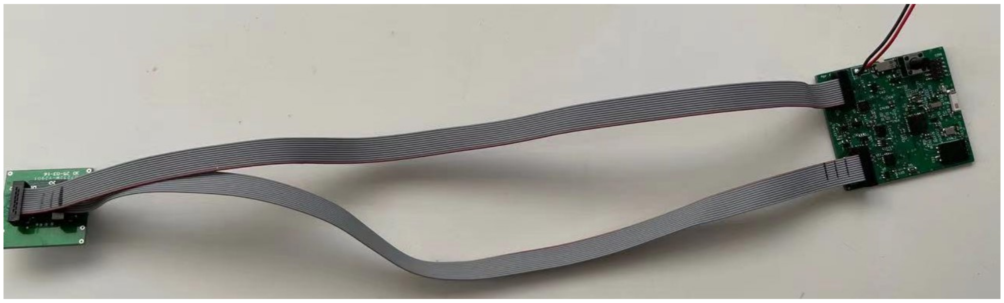


Figure 5-4 The connection between the data acquisition board and the sensing board

A two-layer printed circuit board was also designed for the router board. Unlike the data acquisition platform, the router has no analog circuits and does not require separated voltage supply and ground rails. Since the router was intended to receive data from nodes located from all directions, a 2.4GHz whip antenna, ANT-2.4-CW-HWR-SMA (3.2dBi), was used. An SMA connector connected the whip antenna, as shown in Figure 5-5. The design of the antenna circuitry also used the suggested circuitry used evaluation board circuitry. The same balun, BALF-NRG-02D3, was included for the RF circuitry. Figure 5-5 shows the prototype of the custom router board (42mm x 31mm), and Figure 5-6 shows the selected gateway (56mm x 85mm).

The physical dimensions of the designed data acquisition board and router board are much smaller compared with the platforms described in the previous literature [41-44, 46-48, 50-51]. For example, the Arduino Nano described in [41] has a dimension of 43mm x 19mm x 19mm. The baseboard and radio board proposed by Gao et al. [46] had a dimension of 13.5cm x 7.5cm and 6cm x 6cm, respectively, with the thickness of the power bank (which was not mentioned in the manuscript).

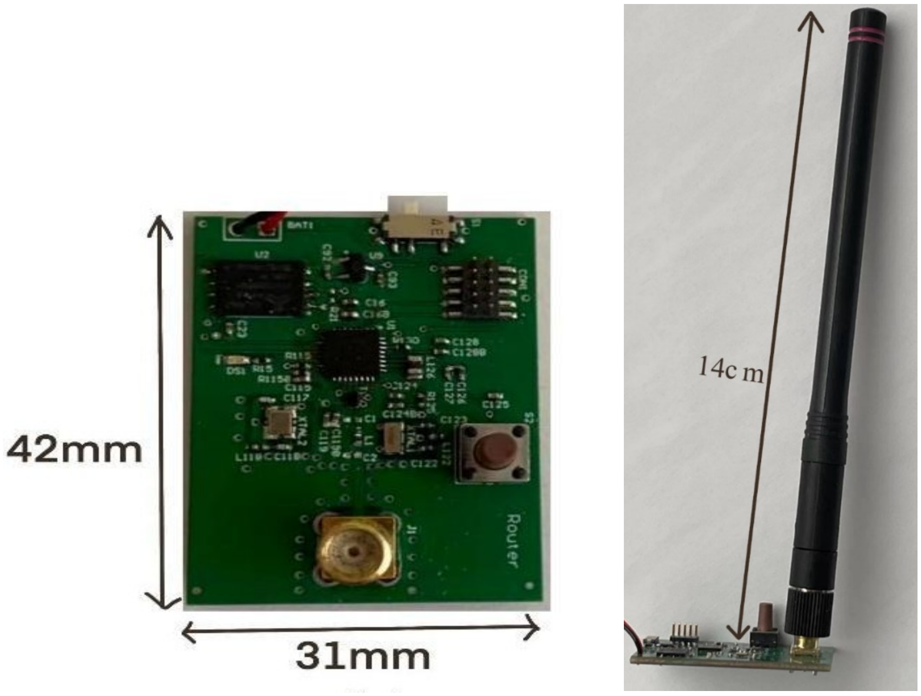


Figure 5-5 The top view of the prototype of the custom router board without antenna (Left) and side view (Right) of the prototype of the custom router board with antenna

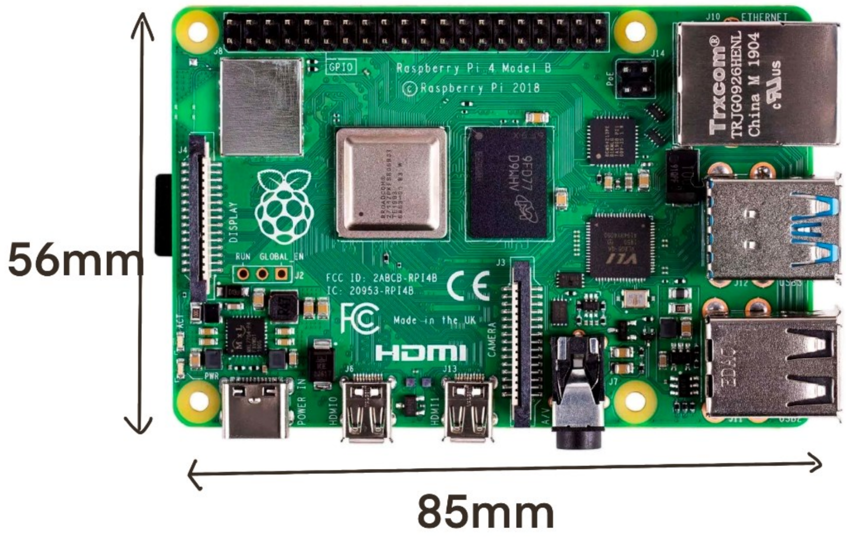


Figure 5-6 The selected gateway (Raspberry Pi 4B) [60]

5.3 ADC and Data Acquisition Experiments

To verify the functionality of the ADC on the sensing board, three experiments related to the ADC and data sampling were performed: a) experiment to determine the ADC conversion time for all 17 channels, b) experiment to understand the sensitivity and crosstalk of the ADC, and c) experiment to investigate the accuracy of the data acquisition platform.

5.3.1 ADC Conversion Time Experiment

The ADC conversion time is the time it takes from the start of the first channel measurement to the end storage of the last channel measurement. As mentioned in Section 4.3.1, eight repeated 3-channel scanning procedures and changing MUX selection signals are required to achieve scanning for all 17 channels. Consequently, the ADC sampling rate is not the actual sampling rate of all 17 channels. A short conversion time allows for a longer sleep time at a fixed sampling rate, reducing the current consumption consequently. It also allows for a possible higher sampling rate in the future. As mentioned in Section 3.2.1.2, the ADC provides multiple oversampling ratios (OSRs) ranging from 32 to 98304. According to the datasheet, a high optimized OSR slows down the data conversion but increases the accuracy. Despite the fact that using a high OSR can reduce the noise, utilizing the maximum value is impossible since a high OSR decreases the sampling rate. Hence, experiments were performed to determine the optimized OSR and investigate the conversion time of all 17 channels. Also, this conversion time would be used to calculate the power consumption of the system.

5.3.1.1 Methods

During the experiments to determine the actual conversion time, starting with an OSR of 32, timestamps were added before the first channel conversion and after the storage of the last channel measurement. A built-in function converted the obtained time difference from internal timer units (with a resolution of 2.5μs) into milliseconds. The node platform was connected to a PC via a Seggar J-link EDU and SEGGER Viewer software to display the results of the conversion time. In total 6 different OSR values: 32, 64, 128, 256, 512, and 1024 were tested. For each OSR, ten conversion times were obtained to get the average and standard deviation.

5.3.1.2 Results

As shown in Table 5-1, an average of 8ms, 9ms, 16ms, 24ms, 40ms, and 51ms were obtained when the ADC was configured with an OSR of 32, 64, 128, 256, 512, and 1024, respectively. These results indicated that a high OSR led to a larger conversion time, in line with the hypothesis. In order to maintain the sampling rate of 20Hz, the microcontroller should finish the conversion and enter sleep mode within 50ms. Additionally, an extra 5ms was required at each sampling period to ensure the SoC could put the flash memory in power-down mode, put the SoC into sleep mode, wake up the SoC from sleep mode, and restore all settings of the SoC. Based on the results, either 256 or 512 OSR could be used. Eventually, an OSR of 256 was chosen, allowing a possible higher sampling rate (~40Hz) in the future.

Table 5-1 Conversion time for all 17 channels with different OSRs

OSR	32	64	128	256	512	1024
Conversion time (Average ±SD) (ms)	8±0.32	9±0.00	16±0.32	24±0.00	40±0.42	51±0.00

5.3.2 Sensitivity and Crosstalk of ADC Experiment

An ADC converts analog signals to digital signals, which plays an important role in data sensing. Since a small resistance change needed to be detected by the acquisition platform, the sensitivity and the crosstalk of the ADC are crucial. Experiments were conducted to investigate the sensitivity and crosstalk of the ADC.

5.3.2.1 Methods

A simulated sensor board was designed to test the ADC sensitivity. As shown in Figure 5-7, it is a printed circuit board which consists of 17 1kΩ variable resistors connected with 500Ω resistors, achieves resistance ranging from 0.5kΩ to 1.7kΩ consequently. The sensing board was connected to the data acquisition board via two connectors, which had the same configuration as the sensor. Figure 5-8 shows the connection between the simulated sensor board and the data acquisition board of the sensing node. The data acquisition board was connected to a PC via a Seggar J-link EDU mini to display the resistor value to screen in real-time. A multimeter (FLUKE 867B, ±0.07% basic accuracy in resistance measurement) was used to measure the actual resistance values. A single channel (channel 16) was selected, and its resistor was adjusted between 200Ω to 1kΩ with a step approximate of 50Ω, 17 steps with final resistance ranging from 700Ω to 1.5kΩ. The resistance was controlled by rotating a fixed number of turns on the variable resistors. All other variable resistors were kept unchanged. At each step, 100 samples of channel 16 were collected for the sensitivity test. Meanwhile, 100 samples at channels 12 and 15 were also collected to investigate the crosstalk.

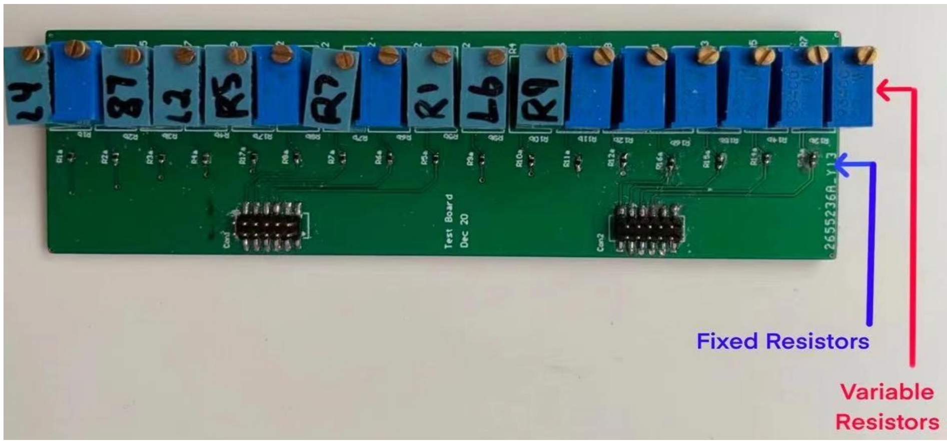


Figure 5-7 The simulated sensor board

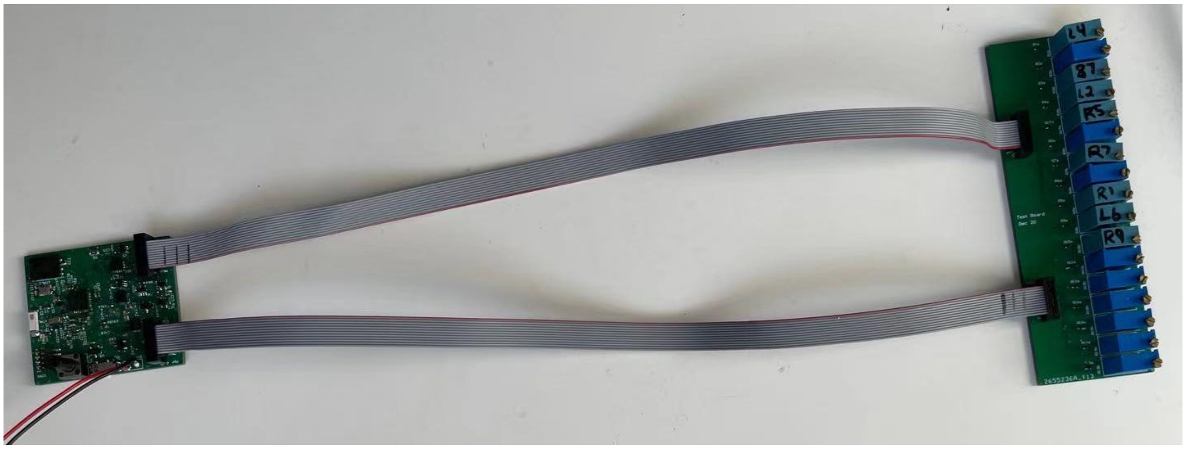


Figure 5-8 The connection between the simulated sensor board and node data acquisition board

5.3.2.2 Results

As the resistance of channel 16 increased, the ADC outputs increased linearly. Figure 5-9 shows a good linearity result on channel 16. The error bars at each of the sample points were too small to display. Using the resistance range from 700Ω to 1.5kΩ, the generated equation is:

$$RES = \frac{ADC\ (16 - bit\ steps) - 145.19}{24.109} \qquad (5\text{-}1)$$

Where RES is the calculated resistance in Ω, and ADC (16-bit steps) is the corresponding ADC output value ranges from 0 to 32767.

The coefficient of determination (denoted by R^2) equals one. The ADC output changed 24 steps for 1-ohm resistance change, while the error was 12.4, equivalent to 0.51Ω. Figure 5-11 shows the Bland Altman plot of the output error for channel 16 versus the average resistance value between ADC output (converted to resistance) and the measured R-value. The output error is the difference between the measured resistance minus the value calculated from the above equation using the ADC outputs. As shown in the plot, there are no systematic errors. Also, there is no bias between the two measurements, and the 95% confidence interval was [-0.504, 0.505]. Hence, a sensitivity of 0.51Ω could be concluded. On the contrary, if a 10-bit ADC is implemented with a 1.5V reference voltage, it can only detect 1.46mV change ideally, equivalent to 14.6Ω.

For channels 12 and 15, the ADC output was kept nearly constant when channel 16 resistance increased, as shown in Figure 5-10. The average difference between the maximum and minimum ADC outputs for channels 12 and 15 were 9 and 14 steps of the 16-bit resolution, equivalent to 0.37Ω and 0.58Ω, respectively. Hence, the crosstalk on the ADC was minimal as the variation of channels 12 and 15 was less than 0.58Ω.

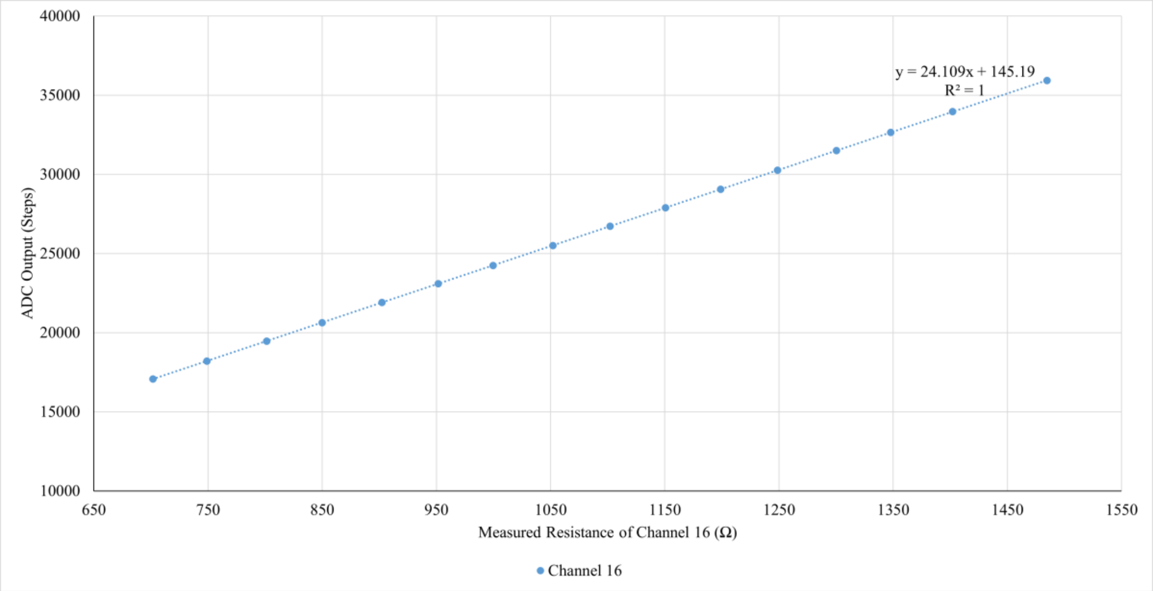


Figure 5-9 ADC outputs of channel 16 at different channel 16 resistance

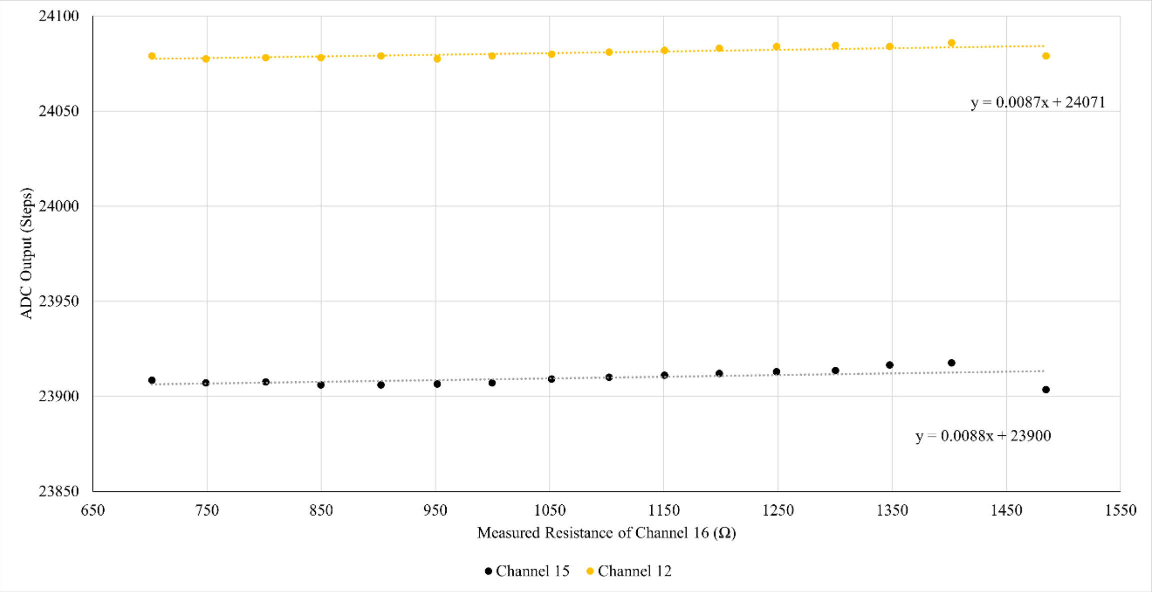


Figure 5-10 ADC outputs of channels 12 and 15 at different channel 16 resistance

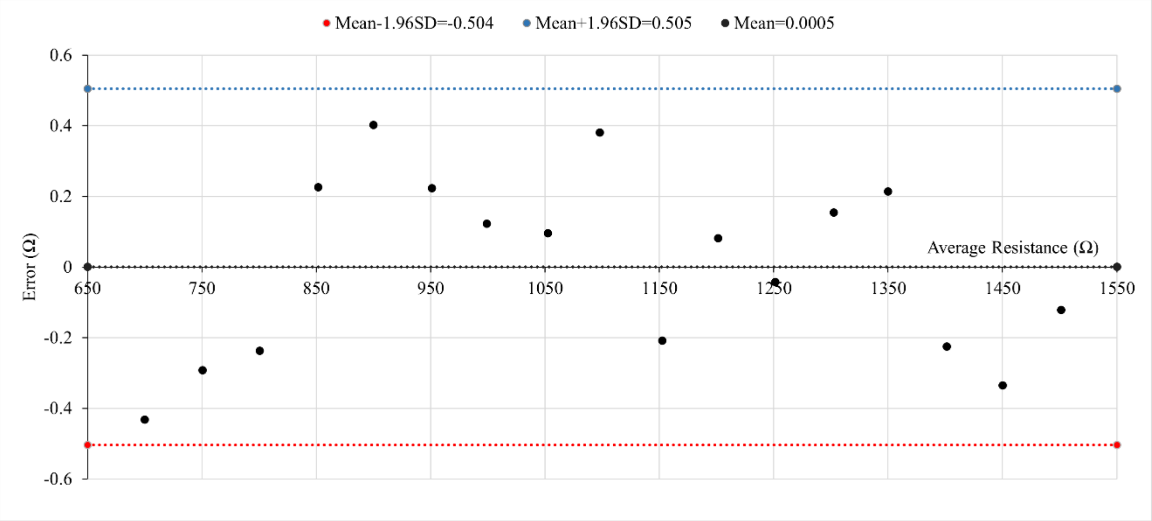


Figure 5-11 Bland Altman plot of the output error

5.3.3 Accuracy of the Data Acquisition Board Experiment

The overall accuracy of the data acquisition unit is crucial. After the data acquisition, the MCU converted the digital value of ADC steps into resistances in Ω with 0.1Ω resolution. The 17 resistance values were then stored in the MCU memory first and transmitted to the gateway or cloud later. An experiment was conducted to determine the accuracy.

5.3.3.1 Methods

Similar to the sensitivity experiment, the simulated sensor board was connected to the data acquisition board, which was connected to a PC via a Seggar J-link EDU mini to display the resistor value onto the screen in real-time, and a multimeter (FLUKE 867B, ±0.07% accuracy in resistance measurement) was used to measure the actual resistance values. All 17 resistors were randomly adjusted. Twenty measurements were recorded for each channel to obtain an average and the standard deviation for comparison.

5.3.3.2 Results

Table 5-2 compares the resistance measured by a multimeter versus the mean and standard deviation value calculated from the ADC with a resolution of 0.1Ω. The difference between the measured and the average values were in the range of [-1.4,0.8]. For all 17 channels, the average absolute difference was 0.58Ω with a standard deviation of 0.34Ω. The percentage difference does not have a correlation with the measured resistance value. Referring to Table 5-2, most of the differences between the measured and average ADC are less than 1Ω, and only 1 out of 17 channels has 1.4Ω. This error could be a consequence of a combination of several factors, including the environmental effects, the inaccuracy of the ADC on that channel, the error from the multimeter, and the error from conversion between ADC steps.

The resolution and accuracy of this sensing node were excellent. According to Kayed et al. [38], the 3D MEMS strain gauge sensor had a sensitivity of 10Ω for 4000µstrain. Therefore, this sensing node was able to detect 400µstrain (1Ω), which was much better than a traditional metal foil strain gauge.

Table 5-2 Comparison between the measured resistances versus the ADC values on all 17 channels

Channel	R (Multimeter) (Ω)	R (ADC output) Mean ± SD (Ω)	△R Mean (Ω)	Absolute △R/R(measured) (%)
1	844.8	845.8±0.056	-1.0	0.12
2	1049.9	1050.7±0.042	-0.8	0.08
3	796	796.9±0.044	-0.9	0.11
4	966.9	966.5±0.049	0.4	0.04
5	622.9	622.9±0.014	0.0	0.00
6	953.7	954.6±0.045	-0.9	0.09
7	1418	1419.4±0.047	-1.4	0.10
8	899	899.2±0.045	-0.2	0.02
9	1513.9	1513.1±0.028	0.8	0.05
10	1255.4	1254.9±0.048	0.5	0.04
11	1456.9	1456.4±0.039	0.5	0.03
12	1207	1206.6±0.054	0.4	0.03
13	1403.2	1403.2±0.052	0.0	0.00
14	1147.7	1147.2±0.051	0.5	0.04
15	1359.4	1358.8±0.027	0.6	0.04
16	1111	1110.5±0.039	0.5	0.05
17	706.9	707.5±0.035	-0.6	0.08

5.4 Bluetooth Low Energy Wireless Communication Range Experiments

After confirming the accuracy of the data acquisition board, Bluetooth Low Energy communication experiments were conducted to investigate both the maximum communication range and transmission time between two devices in a laboratory environment at room temperature. Sections 5.4.1 and 5.4.2 report the experimental setups and results for the range tests from the sensing node to the router and from the router to the gateway, respectively. Based on these results, the maximum distance between node and gateway was also verified (Section 5.4.3).

5.4.1 Communication Range between Node and Router Experiments

5.4.1.1 Methods

In the selected SoC, there are 8 TX power levels (-14dBm, -11dBm, -8dBm, -5dBm, -2dBm, 2dBm, 4dBm, and 8dBm) available. However, the surface mount antenna gain is restricted to 2dBi; selecting the maximum TX power does not mean it can provide the maximum distance range. Therefore, experiments were conducted at each TX power level to determine the maximum communication range between a sensing node and a router board. On the other hand, a fixed TX power level (2dBm) was used for the router since this experiment focused on the TX power of the node; the TX power of the router is not that important as long as the router can send requests to the node. In the beginning, the TX power level of the sensing node was set to the minimum level (-14dBm). A sensing node and a router board were placed on the ground level starting at 0.5m apart; the distance was measured from the antenna to the antenna using a meter ruler. The router board was connected to a laptop via a Seggar J-link EDU mini to display the received data packages onto the screen in real-time. Figure 5-12 shows the experimental setups and the boards' orientation. To consider a successful communication, the router sent a data request to the node and waited for a 200bytes data packet as the response. This process was repeated five times before moving to the next distance range with a step of 0.25m until an unsuccessful communication occurred. Following the same procedures, the maximum communication distances at seven other TX power levels were determined.

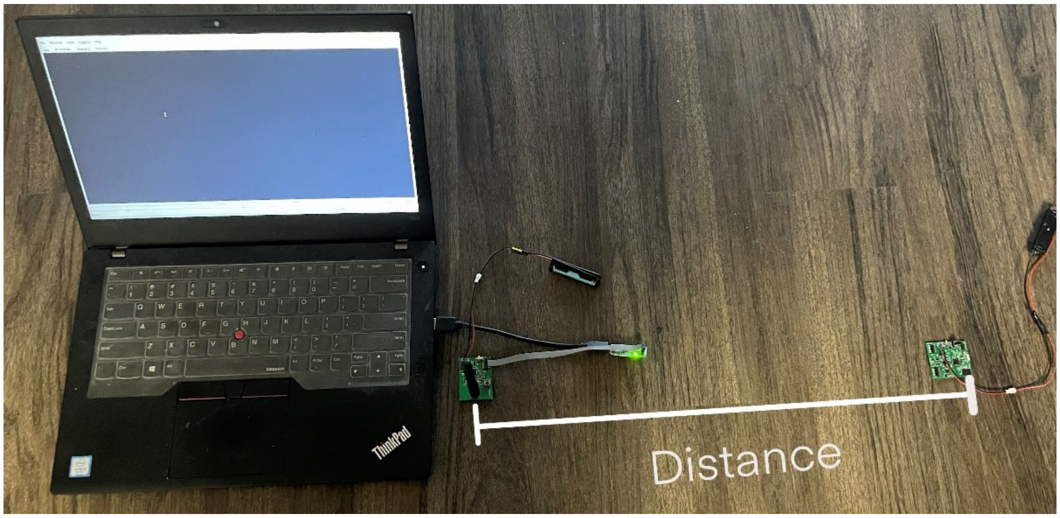


Figure 5-12 Experimental setups for communication range test between node and router

5.4.1.2 Results

The maximum transmission distances between the node and router at different TX power levels are shown in Figure 5-13. The results showed that the max transmission distance increased logarithmically with the increase of TX power level until 4dBm. The maximum distance from a node to a router was 17m and occurred when the node TX power level was 4dBm. At 8dBm, the maximum transmission distance was only 6.5m, which might be explained by the limitation of the surface mount antenna.

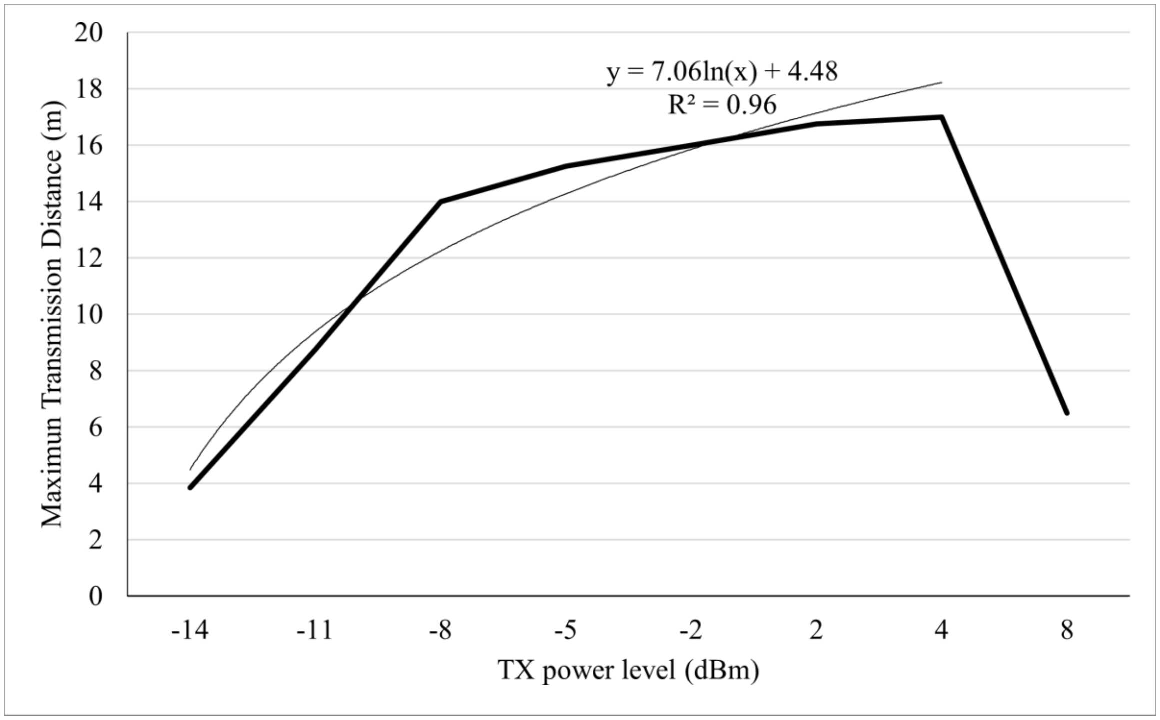


Figure 5-13 Maximal transmission distance between node and router at different TX power levels

5.4.2 Communication Range between Router and Gateway Experiments

5.4.2.1 Methods

There are also 8 TX power levels (-14dBm, -11dBm, -8dBm, -5dBm, -2dBm, 2dBm, 4dBm, and 8dBm) available for the router. Experiments were conducted to determine the maximum transmission distances between a router and the gateway at different TX levels. In the beginning, the TX power level of the router was set to the minimum level (-14dBm). Both the router and gateway were placed on the ground level starting at 0.5m apart. The gateway was connected with a monitor to view the received data package. Figure 5-14 shows the experimental setups for the communication range test between router and gateway. The router sent a 200bytes data package. Then, the data package received by the gateway and displayed on the monitor was compared with the original data package. If the sent and received data are the same, it was considered as a successful communication. This process was repeated five times before moving to the next distance with a step of 0.25m until an unsuccessful communication occurred. Following the same procedures, the maximum communication distances at higher TX power levels were determined until a significant decrease in the maximum distance occurred.

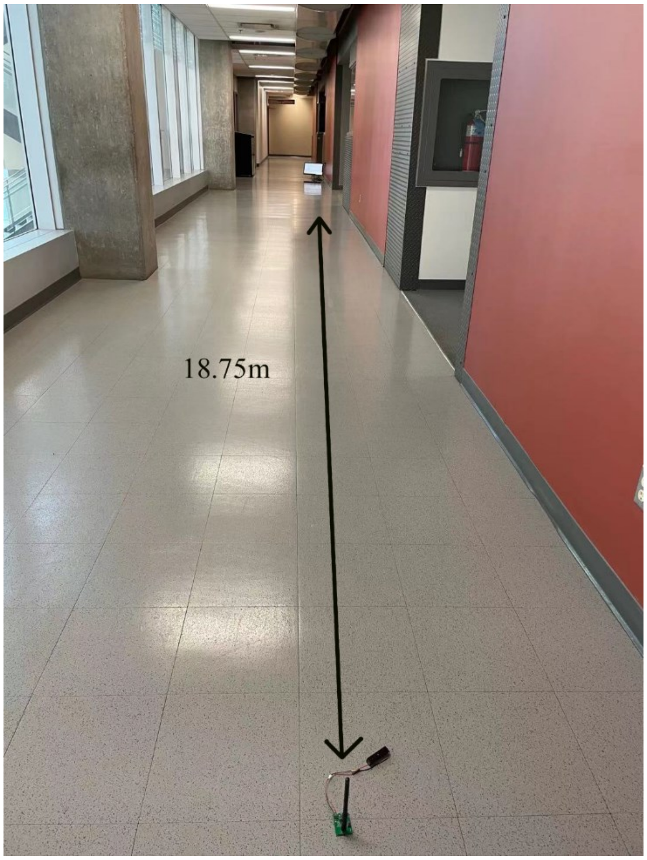


Figure 5-14 Experimental setups for communication range test between router and gateway

5.4.2.2 Results

The maximum transmission distances between the router and gateway at seven different TX power levels are shown in Figure 5-15. The results showed that the max transmission distance increased logarithmically with the TX power level until 2dBm (R^2=0.93). The maximum distance between the router and gateway was 18.75m when the router was set to 2dBm. Then, a sudden decrease in the distance occurred at 4dBm. At 4dBm, the maximum transmission distance was only 9.75m, which might be explained by the limitation of the antenna. A similar BLE max range was obtained for router and gateway, comparing with node to a router.

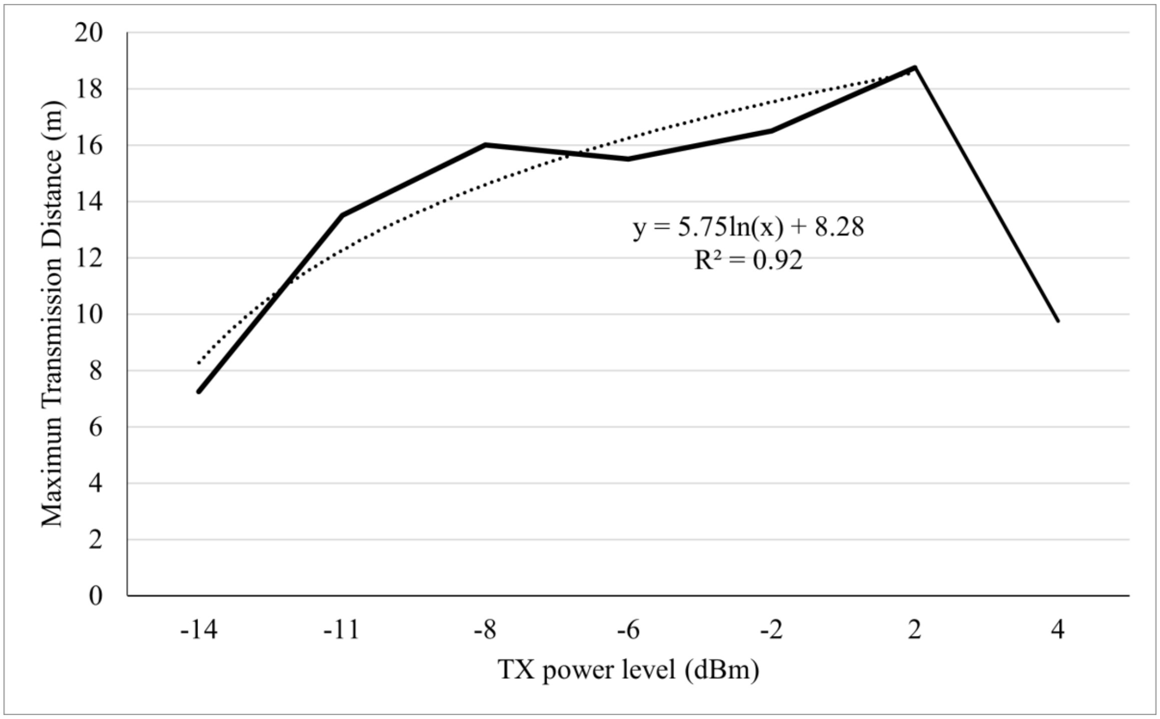


Figure 5-15 Maximal transmission distance between router and gateway at different TX power levels

5.4.3 Communication Range between Node and Gateway Experiments

5.4.3.1 Methods

A node, a router, and a gateway were placed on the ground level in a straight line and line of sight. In the first experiment, the node and router were placed 17m apart while the router and gateway were placed 18.75m apart, as shown in Figure 5-16. The node was put at 4dBm TX level, and the router was put at 2dBm. The node sent 1kbytes known data with multiple data packages to the router; then, the router sent the received 1kbytes to the gateway. A monitor was connected with the gateway to compare the received data with known data. The above procedures were repeated five times. If an unsuccessful communication occurs, 1m was deducted from both the distances from the router to gateway and node to router until five successful communications occurred.

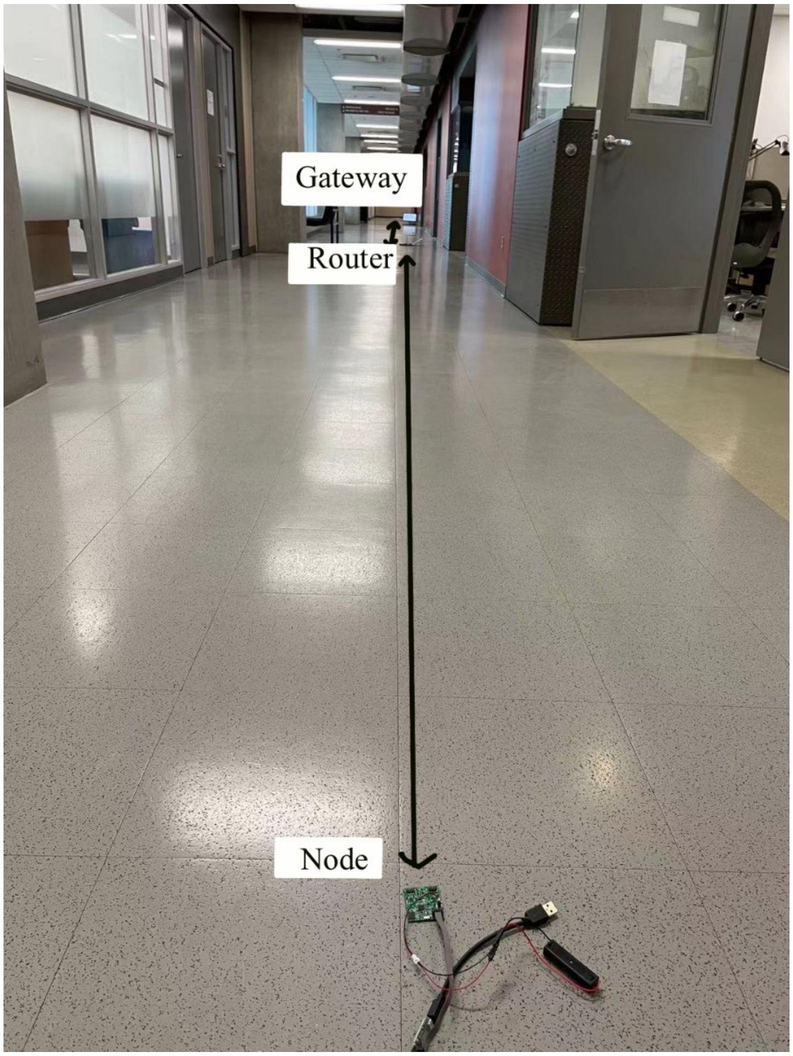


Figure 5-16 Experimental setups for communication range test between node and gateway

5.4.3.2 Results

Eventually, five successful communications occurred at the first step. Consistent with the results in Section 5.4.1.2 and Section 5.4.2.2, the maximum distance between the node and gateway was 35.75m, when the node and router were put 17m apart, and the router and gateway were put 18.75m apart.

5.5 Bluetooth Low Energy Transmission Time Experiments

Data acquisition cannot be performed simultaneously during the data transmission between the sensing node and the router. Experiments were conducted to determine the transmission time. Section 5.5.1 reports the transmission time for 1Mbytes for different distances. Section 5.5.2 reports the transmission time for 1Mbtyes with 1m apart at different height levels. The node data may accumulate when the router is busy; therefore, sometimes, more than 1Mbytes data may be transmitted at a time. Section 5.5.3 investigates the relationship between the transmission time and the amount of data to be transmitted.

5.5.1 Transmission Time for Different Distances Experiments

5.5.1.1 Methods

This section describes the transmission time experiments for 1Mbytes of data with a fixed orientation between the sensing node board and router at different distances. The TX power levels of the sensing node and router were 4dBm and 2dBm, respectively. A sensing node and a router board were placed on the ground level starting at 3m apart. The antenna orientation was the same as that shown in Figure 5-12. The sensing node continuously read and transmitted data from the external flash memory, and timestamps were added at the beginning and end of the 1Mbytes data transmission. The node board was connected to a laptop to display the two timestamps, which could be used to obtain the transmission time in milliseconds. The above procedure was repeated five times to obtain the average and standard deviation. The experiment was repeated at the distances: 5m, 8m, 10m, 11m, 14m, and 15m.

5.5.1.2 Results

Table 5-3 shows the transmission time results in minutes, including five trials and the calculated average and standard deviation for 1Mbytes data. Figure 5-17 shows that the transmission time increases exponentially as the communication distance increases (R^2=0.97). When the node and router were 14m apart, approximately 18 minutes was required for 1Mbytes data transmission, while 1Mbytes is equivalent to 65-minutes of data at 5Hz and 17 channels. Hence, placing a sensing node and a router at the maximum distance increased the transmission time significantly.

Table 5-3 Transmission time in min for 1Mbytes data at different communication distances

The horizontal distance (m)	Transmission time (min)					
	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Average ±SD
3	2.097	1.767	1.792	1.797	1.880	1.867±0.136
5	1.725	1.870	1.602	1.607	1.937	1.748±0.152
8	7.792	7.673	5.595	5.654	5.795	6.502±1.127
10	12.330	11.660	8.425	7.628	11.349	10.278±2.105
11	10.264	11.823	12.194	11.262	11.972	11.503±0.773
14	19.703	15.783	18.371	16.277	20.323	18.091±2.017
15	23.979	17.874	26.603	25.009	22.906	23.274±3.312

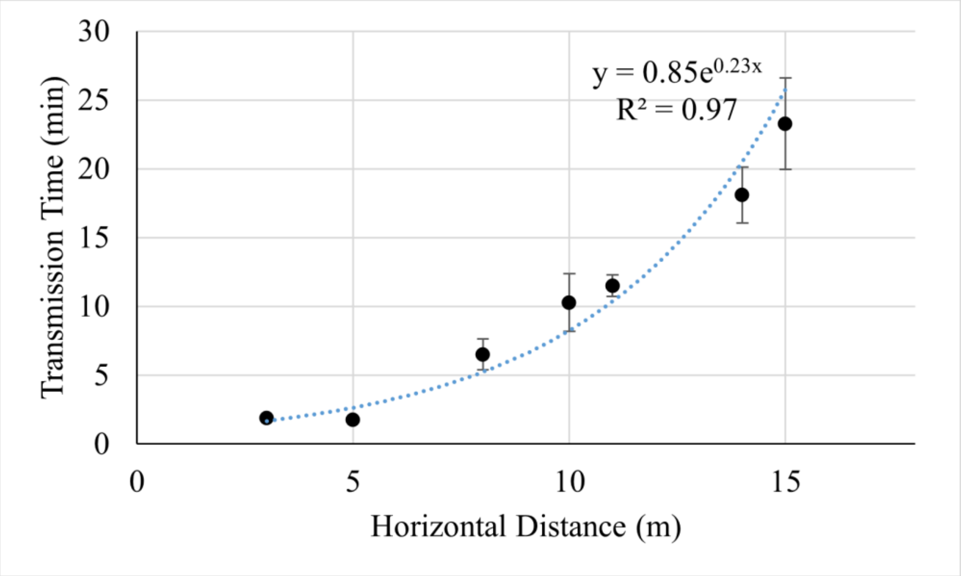


Figure 5-17 Transmission time for 1Mbytes data at different communication distances

5.5.2 Transmission Time for a 1 m at Different Height Levels Experiments

5.5.2.1 Methods

Similar to the experiment described in Section 5.5.1, the router was increased from the ground level to 1m at 0.25m per step. In this series of experiments, the distance between the sensing node and router was fixed at 1m. The sensing node continuously read and transmitted data from the external memory; timestamps were added at the beginning and end of the 1Mbytes data transmission. The procedure was repeated five times to obtain the average transmission time. Figure 5-18 illustrates the experimental setups described in this section.

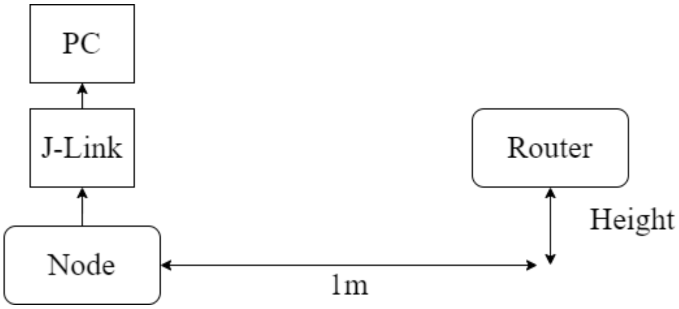


Figure 5-18 Experimental setups for 1Mbytes data transmission time vs. heights

5.5.2.2 Results

Table 5-4 shows the results of the transmission time at different heights between the node and router with a fixed 1 m away on the ground level, including five trials and the calculated average and standard deviation for 1Mbytes data in minutes. Figure 5-19 indicates that the transmission time increased exponentially as the height increased with $R^2 = 0.99$. When the router was 1m above the ground level, approximately 3.6 minutes were required. The result indicated that the transmission time was double when the ground level and the height were 1m. However, the direct distance only increased from 1m to 1.4m. The increase in transmission time could be attributed to the decrease of signal strength due to the surface mount antenna on the sensing node.

Table 5-4 Transmission time for 1Mbytes data at different heights

Height (m)	Transmission time (min)					
	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Average ±SD
0	1.761	1.821	1.921	1.856	1.788	1.829±0.062
0.25	2.114	2.098	2.035	2.039	2.020	2.061±0.042
0.5	2.595	2.641	2.837	2.549	2.546	2.634±0.120
0.75	2.833	2.973	2.848	3.067	2.782	2.901±0.117
1	3.643	3.501	3.681	3.446	3.521	3.558±0.099

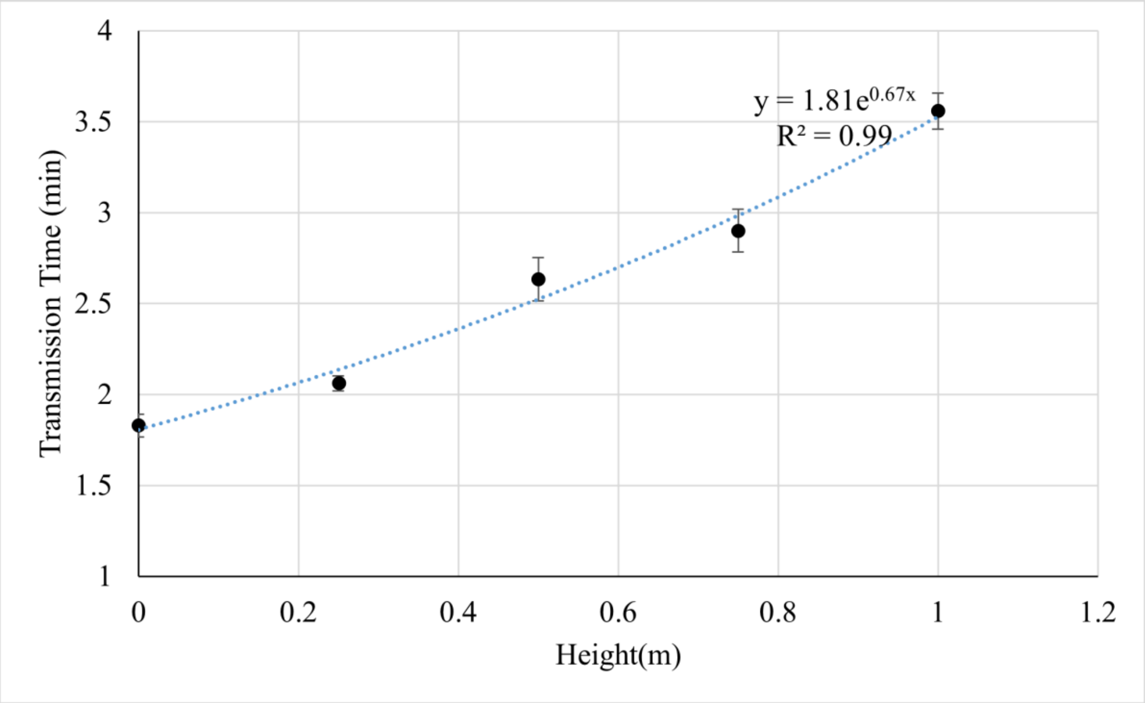


Figure 5-19 Transmission time for 1Mbytes data at different heights

5.5.3 Transmission Time for Different Data Size Experiments

5.5.3.1 Methods

Experiments were performed to investigate whether the transmission time had a linear relationship with the amount of data transferred. Similar to the experiments in Section 5.5.1, the TX power levels of the sensing node and router were set to 4dBm and 2dBm, respectively. Both the sensing node and the router were placed on the ground level 2m apart. The sensing node kept reading data from the external memory and transmitting it to the router. A timestamp was added at the beginning of the data transmission, and three more timestamps were added when 1Mbtye, 2Mbytes, and 3Mbytes were completely transmitted. The node board was connected to a laptop to calculate the transmission times. The above procedure was repeated five times to obtain the average transmission time.

5.5.3.2 Results

On average, 3.78minutes, 7.48minutes, and 11.38minutes were required for 1Mbytes, 2Mbytes, and 3Mbytes transmission, respectively. As shown in Table 5-5, although the transmission time varies between trials, the transmission time for 2Mbytes data is approximately two times that of 1Mbytes under each trial. Similarly, the transmission time for 3Mbytes data is approximately three times that of 1Mbytes under the same trial. Therefore, the transmission time at a fixed distance, orientation, and transmit (TX) power level is linearly proportional to the data size.

Table 5-5 Transmission time for 1Mbytes, 2Mbytes, and 3Mbytes data at fixed communication distance, antenna orientation, and transmit power level

Data Size	Transmission Time (min)					
	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Average ± SD
1Mbytes	1.249	1.243	1.255	1.508	1.340	1.319±0.113
2Mbytes	2.508	2.497	2.480	2.870	2.751	2.621±0.178
3Mbytes	3.760	3.804	3.890	4.389	4.052	3.979±0.255

5.6 Current Consumption for Sensing Node Experiments

Since the low power consumption of the IoT design was one of the main focuses of this study, experiments were conducted to evaluate the current consumption of the node platform in different operational modes. Section 5.6.1 describes the study methods, and Section 5.6.2 reports the experimental and theoretical current consumption. Furthermore, based on results in Section 5.6.2, Section 5.6.3 provides a theoretical calculation of the current consumption of the sensing node with and without the intelligent data acquisition approach implemented.

5.6.1 Methods

The average current consumption of the sensing node was measured at three different operation modes, including a) data transmission mode, in which the node continuously reads data from the external memory and sends them to a router, b) the active mode, in which the node continuously scans all 17 channels and stores data into MCU flash memory, and c) sleep mode, in which the MCU, external flash memory, and ADC are all put into low-power mode, this is the mode of operation that the node used when waiting for the next sampling cycle. A laboratory power supply provided a 3.6V was connected in series to a multimeter and a sensing node. The multimeter was used to monitor the average current consumption for 20 seconds at each mode. Figure 5-20 shows the experimental setups for current tests.

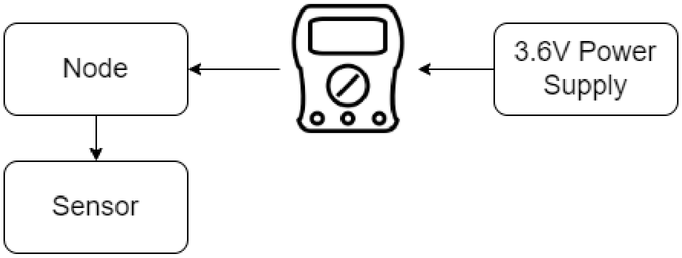


Figure 5-20 Experimental setups for current tests

5.6.2 Results

Figure 5-21 shows the average current consumption of the node platform in transmission mode at different TX power levels. The current consumption increased linearly with the TX power level with $R^2 = 0.94$. At the 4dBm TX power level, the node platform consumed 10.28mA in transmission mode.

The multimeter measured 6.45mA and 2.61mA while the node was in active mode and sleep mode, respectively. However, based on the current consumption stated in the manufacturers' datasheets, theoretical current consumption at active mode and the sleep mode should be 4.75mA and 1.35mA, as listed in Tables 5-6 and 5-7. The percentage difference between the theoretical and the actual values are 35.8% (6.45-4.75/4.75) and 93.3% (2.61-1.35/1.35). Potential discrepancies might be attributed to the combination of multiple factors. First of all, some of the I/O pins of the MCU had internal pull-up/down resistors, and it would draw current into it. In addition, the efficiency of the voltage regulator is not reported in the datasheet. With an efficiency of 75%, the node platform was supposed to draw 6.33mA (4.75/75%). As in the general situation, the lower the load current on a voltage regulator, the lower the conversion efficiency. 2.61mA in sleep mode might be explained by a lower conversion efficiency of 50% (1.35mA/50%=2.7mA). Furthermore, a standby current (35µA) for the external flash memory was used while calculating the theoretical current consumption in active mode. However, the external flash memory's reading/programming/erasing operations could draw a maximum of 40mA, according to the datasheet. Besides, the SoC features multiple sleep modes. One possible explanation for higher sleep mode current consumption might be that the SoC was not put into the complete sleep mode. Some internal modules were still active, including the RAM, system controller, GPIO, and the wake-up timer. They might draw some extra 12 µA.

Compared with other reported IoT SHM platforms, this platform consumed a very low current consumption. In [42], the current consumption of the proposed was roughly 38 mA and 117 mA without and with the Wi-Fi module running, which is significantly larger.

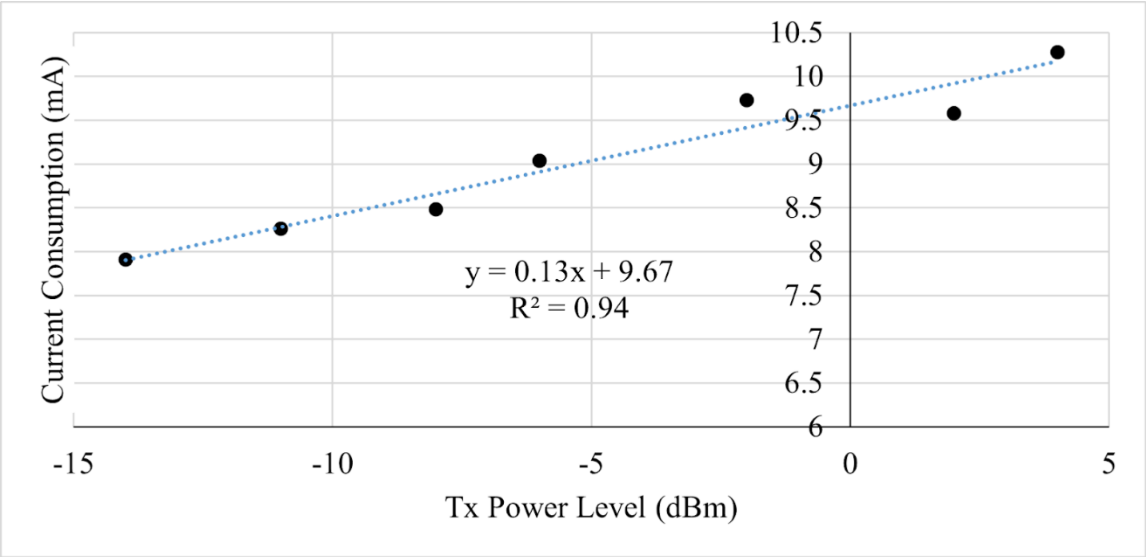


Figure 5-21 Current consumption in transmission mode at different TX power levels

Table 5-6 Theoretical current consumption in active mode

Component	Current consumption per unit	Number of units	Total current consumption
SoC	1.9mA	1	1.9mA
ADC	1.67mA	1	1.67mA
Current source	108 µA	3	324 µA
Sensor	300 µA	1	300µA
Multiplexer	1µA	2	2µA
Flash memory	35µA	1	35µA
Op-amp	170 µA	3	510 µA
Voltage regulator	4µA	3	12µA
			4.75mA

Table 5-7 Theoretical current consumption in sleep mode

Component	Current consumption per unit	Number of units	Total current consumption
SoC	2.1µA	1	2.1µA
ADC	180µA	1	180µA
Current source	108 µA	3	324 µA
Sensor	300 µA	1	300µA
Multiplexer	1µA	2	2µA
Flash memory	20µA	1	20µA
Op-amp	170 µA	3	510µA
Voltage regulator	4µA	3	12µA
			1.35mA

5.6.3 Comparison of the Current Consumption with and without Optimized Data Acquisition Approach

This research implemented an optimized data acquisition approach, in which a low power mode and a dynamic sampling frequency of either 20Hz or 5Hz were implemented. Although a high sampling frequency (over MHz) described in [46] ensures detection of fast changes, the flash memory capacity requirement and power consumption also increased dramatically. This section shows a current consumption comparison between the optimized data acquisition approach and the constant sample rate at the 20 Hz method. In this comparison, data were continuously transmitted when 1Mbytes of data were filled into the external memory; ten times abnormal stress/strain were detected per day; the node and router were placed on the ground level 8m apart; an OSR of 256 and a TX power level of 4dBm were used for the node. Hence, 6.502minutes and 10.28mA were used in the calculation regarding the 1M bytes data transmission, as shown in Section 5.5.1. As shown in Section 5.6, the node platform drew 6.45mA and 2.61mA in active mode and sleep mode. As shown in Section 5.3.1, the data acquisition time for 17 channels required 24ms.

For the current consumption with the optimized data acquisition, the node was put into 20Hz for 50 seconds when there were ten times abnormal stress/strain detected, and 1000 samples were obtained per channel during the 50s. Assumed the node was put into 5Hz for T seconds, 5T samples were obtained per channel during Ts. Hence, the total number of samples per channel per day (N), data size for all 17 channels, and transmission time for these data could be derived:

$$N = 5T + 1000 \tag{5-2}$$

$$Data\ Size\ (bytes) = \mathrm{N} \times 17 \times 3 \tag{5-3}$$

$$Transmission\ Time\ (s) = \frac{Data\ Size}{10^6} \times 6.502min \times 60s/min \tag{5-4}$$

The summation of transmission time and time in 5Hz and 20Hz is 24hours:

$$Transmission\ Time\ (s) + 50 + T = 24 \times 60 \times 60 \tag{5-5}$$

By solving Equations (5-2) to (5-5), 78519s and 393595samples were obtained for T and N. The transmission time was approximately 2.18hours. The active time and sleep time could then be calculated:

$$Active\ Time = \mathrm{N} \times 24\mathrm{ms} = 2.62\mathrm{h} \tag{5-6}$$

$$\begin{aligned} Sleep\ Time = {} & 24 - Active\ Time(h) - Transmission\ Time(h) \\ & = 19.20\mathrm{h} \end{aligned} \tag{5-7}$$

Table 5-8 summarizes the current consumption per day at different modes, assuming ten times abnormal stress/strain are detected. On average, the current consumption per hour was 3.76mA (90.2mAh/24h). The battery life span was calculated based on 960mAh (using 80% of the battery capacity 1200mAh with adding 20% safety factor in my calculation). The power consumption of the node platform was 13.54mW (3.76mA x 3.6V).

Table 5-8 Calculation results with optimized data acquisition approach

	Time /day (hour /day)	Current consumption (mA)	Current consumption per day (mAh)
Active mode	2.62	6.75	17.7
Sleep mode	19.20	2.61	50.1
Transmission mode	2.18	10.28	22.4
Total current consumption per day (mAh)	90.2		
Battery lifetime (days)	10.6		
Data size (Mbytes)	20.1		

The total current consumption, battery lifetime, and data size without implementing the optimized acquisition approach were calculated for comparison. Without the optimized acquisition approach, the node continuously acquired data at 20Hz, and the low power mode was not integrated. With T seconds in the active mode, 20T samples were collected per channel, and data size equaled 1020T (20Hz x 17channels x 3bytes/sample x Ts). By substituting this data size into Equation (5-4), the transmission time could be expressed as a function of T, given by Equation (5-8). The active and transmission times could be solved using Equations (5-8) and (5-9).

$$Transmission\,Time\;(s) = \frac{1020T}{10^6} \times 6.502min \times 60s/min \tag{5-8}$$

$$Transmission\,Time\;(s) + T = 24 \times 60 \times 60 \tag{5-9}$$

In the end, 6.91h and 17.09h were obtained for the active time (T) and transmission time, respectively. Based on these values, the total current consumption per day, battery lifetime, and data size without the optimized data acquisition process were calculated, as shown in Table 5-9. The average current consumption was 7.77mA (186.4/24), resulting in 28.0mW power consumption (7.77×3.6).

Table 5-9 Calculation results without optimized data acquisition approach

	Time /day (hour /day)	Current consumption (mA)	Current consumption per day (mAh)
Active mode	17.09	6.75	115.4
Transmission mode	6.91	10.28	71.0
Total current consumption per day (mAh)	186.4		
Battery lifetime (days)	5.1		
Data size (Mbytes)	62.8		

Table 5-8 and Table 5-9 show that 52% (186.4-90.2/186.4) current consumption was saved with the dynamic data acquisition approach implemented for ten triggers per day scenario. In addition, 68% of memory capacity ((62.8-20.1)/62.8) was saved.

Figure 5-22 shows that the total current consumption per day increases slowly and linearly as the number of triggers increases (R^2=1). For 120 triggers per day scenario (10 mins in 20Hz), the active sleep and transmission times were solved to be 2.67h, 2.22h, and 19.11h. The total current consumption per day increased to 90.7mAh. The average current consumption per hour was 3.78mA (90.7/24), resulting in 13.6mW power consumption (3.78×3.6).

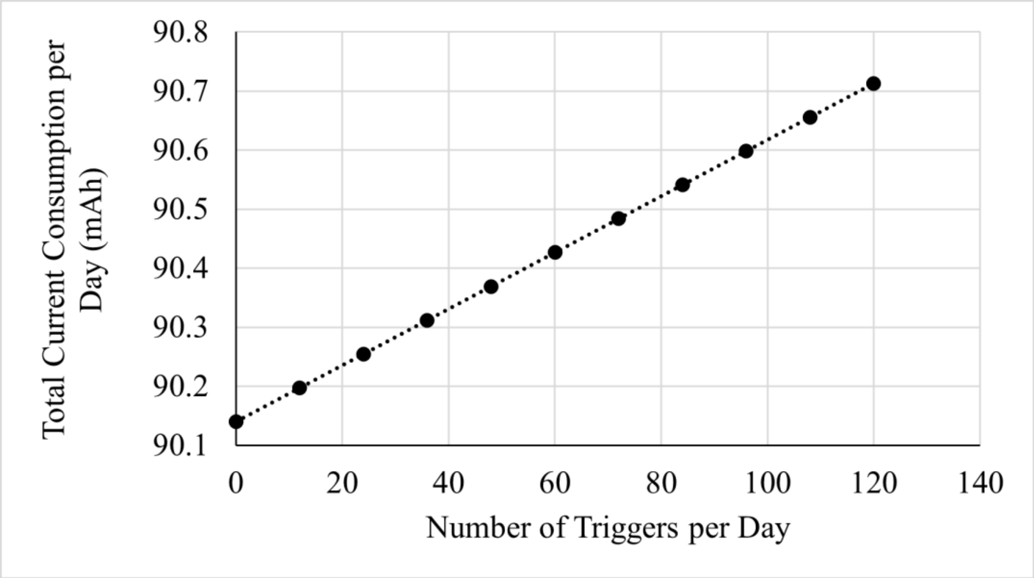


Figure 5-22 Total current consumption for different scenarios

5.7 Validation of the Entire System Experiment

5.7.1 Methods

In order to validate all the functionalities of the entire IoT system, the system was operated for seven days at room temperature. The node was connected with the simulated sensor board, which allowed adjusting the resistance to trigger the high sampling rate. The node and router were placed at 2m apart; the gateway and router were put at 1 meter apart. Since the communication distance has been validated, the distance in this experiment was not important. Before the experiment started, the battery level was measured using the multimeter. First, I used the Android interface to input 13 for N_1 and 10% for the threshold percentage. During the experiments, when 13 (N_1) consecutive samples at 5Hz deviated from the 10% (threshold) of the baseline value, the sensing node would switch to a 20Hz sampling rate for 5seconds.

Each day, the variable resistor of a channel was randomly adjusted with multiple turns to ensure the resistance value was below the threshold value. This action would ensure triggering the high sampling frequency. The time of resistance adjustment, channel number, adjustment action (increase/decrease), and the number of turns were recorded. After one minute, the same variable resistor was adjusted with the same number of turns in the reverse direction, ensuring the resistance was back to approximately the beginning state. Then, the battery voltage was monitored using the multimeter when the node platform was running. The above procedures were repeated three times per day at different channels. The first change was for channel one and then moved to the next channel. After channel 17, it switched back to channel one.

After seven days, all platforms were turned off. The battery was disconnected from the circuit, and its voltage was measured using the multimeter after five hours since the battery voltage increased when the discharge was stopped. Data on Google Drive were downloaded to the laptop for validation.

5.7.2 Results

The system validation experiment showed that all wireless communication between node, router, gateway, and the cloud worked properly. In the beginning, the battery voltage at no load was 3.671V.

Based on the recorded time and channel number, the corresponding data was checked to see whether the change in resistance at that channel and a dynamic sampling rate could be observed. Figure 5-23 shows the corresponding one-hour resistance data for channel 8 when channel 8 was adjusted. The channel 8 resistance decreased from 1383 to 1224 ((1383-1224)/1383=11.5%), which exceeded the 10% limit and started the high-frequency sampling period. Approximately 860 measurements were taken during the high-frequency sampling period, in which a lower resistance of channel 8 was observed. With an average sampling rate of 14.87Hz, this high-frequency sampling period was equivalent to 57.8 seconds (860/14.87), approximately 1 minute. Equation (5-10) shows the calculation of the average sampling rate when N_1 equals 13. As mentioned in Section 4.3, a time was stored for every 300 samples per channel. The recorded time showed that 300 samples took around 60s during the normal period and 20s during the high-frequency sampling period. These findings demonstrate that a dynamic sampling rate was implemented, and these values are consistent with the theoretical values. Theoretically, those values should be 60 (300/5) and 20.1 (300/14.87) when N_1 equals 13.

$$Average\ sampling\ rate = \frac{Number\ of\ samples}{Time} \tag{5-10}$$

$$= \frac{13 + 20 \times 5\ samples}{13 \times 0.2s + 5s} = 14.87Hz$$

Following the same procedures, the changes in resistance of all channels were detected, and a dynamic sampling rate was observed. In total, 21 high-frequency sampling periods (1minute/period) occurred. In each period, approximately 900 samples were collected per channel, equivalent to 60.5s (900/14.87) data. At the end of the seven-day experiment, around 3million samples were collected per channel in total. This value is consistent with the theoretical value of 2.95million (0.422x7), while the 0.422 is the theoretical number of samples per channel per day, N. Equations (5-11) to (5-13) show the calculation of N, where T is the time of node in 5Hz, and 2700 (900 samples/period x 3periods/day) is the number of samples taken in 20Hz. Equation (5-12) shows that the transmission time could be derived based on the data size. Since the node and router were placed 2m apart instead of 8m, 1.867min was used for 1M bytes data transmission. By substituting Equations (5-11) and (5-12) into (5-13), T could be solved.

$$N = 5T + 2700 \tag{5-11}$$

$$Transmission\ Time$$

$$= \frac{Nsamples\ \times\ 17channels\ \times\ 3bytes/sample}{1 \times 10^6} \times\ 1.867min \tag{5-12}$$

$$Transmission\ Time\ +\ 50s\ +\ T = 24\ hours \tag{5-13}$$

Eventually, 0.422million was obtained for N, the theoretical total number of samples per channel per day.

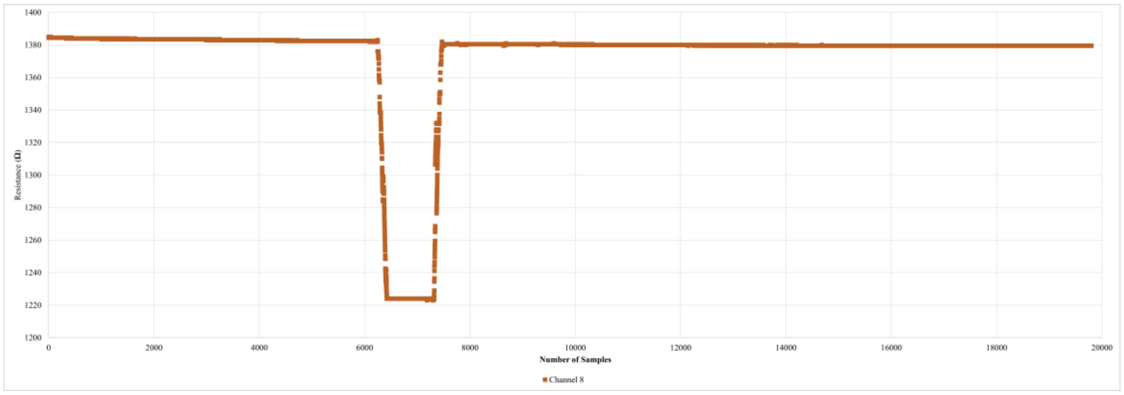


Figure 5-23 One hour data for channel 8

The battery voltage at load kept falling slowly and from 3.54V to 3.47V while the system was running, as shown by the black points in Figure 5-24. After the experiment was completed, the no-load battery voltage decreased from 3.671V to 3.660V, as shown by the two orange points in Figure 5-24. Compared with no-load battery voltage, a lower battery voltage was obtained at load, which could be explained by the internal resistance of the battery. The results demonstrate that the battery voltage decreases gradually, which also fits with the discharge curve of the battery (Figure 3-5(c)). The discharge curves of the battery show that the battery voltage decreases slowly until reaching the limit points regardless of how many currents were drawn.

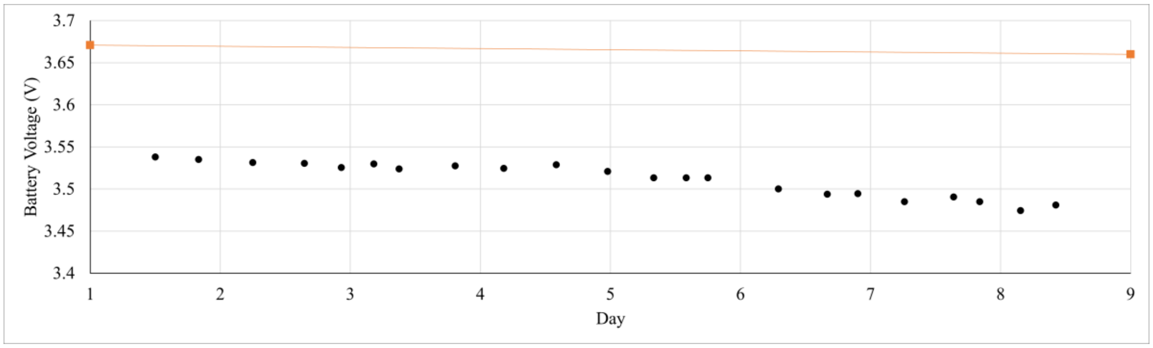


Figure 5-24 Measured battery voltage in-between seven days

5.8 Summary

Compared to other reported IoT SHM systems, this work reported a very low power (14.0mW) miniaturized (35mm x 38mm x 10mm) data acquisition platform to measure 3D stress and strain on a structure. The accuracy, communication performance, and current consumption have been tested, as well as the entire system validation. The sensitivity of the ADC was 0.51Ω, and the accuracy of the data acquisition board was 0.58±0.34Ω. The maximum distance from a node to a router was 17m when the TX power level of the sensing node was at 4dBm. Also, the maximum distance between the router and gateway was 18.75m when the TX power level of the router was at 2dBm. The node platform consumes 6.45mA and 2.61mA in active mode and sleep mode. For a ten triggers per day scenario, Implementing the dynamic data acquisition approach saved 52% of the current consumption and 68% of memory compared with a constant sampling rate of 20Hz. The validation of the entire system also demonstrated that the node could detect resistance changes of all 17 channels and could last for more than seven days with a battery capacity of 1200mAh while the node was put in the high-frequency mode for around 3 minutes per day.

Chapter 6. Conclusions and Recommendations

6.1 Conclusions

Many researchers have studied stress-strain sensors and IoT-based structural health monitoring systems. There are many different types of stress-strain sensors, but in my thesis, I selected the MEMS strain-gauge type sensor because they were designed locally and had high accuracy, stability, small size, and lower power consumption. From the literature, Wi-Fi and ZigBee are the most common wireless communication protocols that have been used for SHM. At present, most studies on IoT-based SHM platforms focus on low cost and low power consumption. The most challenging aspect of new IoT design is power consumption. This project was undertaken to design and develop an IoT-based SHM system that included a miniaturized low-power IoT acquisition platform that was integrated with a 3D stress-strain MEMS sensor and implemented with an intelligent data acquisition process.

In this thesis, an IoT-based structural health monitoring system was designed and developed. It consisted of an extreme low-power acquisition platform, a MEMS 3D stress-strain gauged typed sensor, a router, and a gateway (Raspberry pi). The BLE wireless communication protocol was selected as the data transfer between node and gateway. In contrast, the gateway uploaded data to Cloud storage (e.g., Google Drive) via Wi-Fi. During the hardware design, component selections were mainly based on current consumption, physical dimensions, operating temperature range, BLE version, and accuracy. The three major components of the acquisition platform were BlueNRG-232 (SoC), MCP3462 (ADC), and S25FL256L (memory). During the design, the current sources circuit was compared with voltage sources plus the Wheatstone bridge circuit. Using the current sources circuit demonstrated that the number of ICs was reduced, but the sensitivity was improved. To further reduce the number of components, multiplexers were used instead of using multiple ADCs to interface the MEMS sensor to allow 17 channels acquisition. The current consumption and physical dimensions were further reduced in this configuration. While designing the printed circuit board of the acquisition unit, separation of ground for analog and digital powered was layout to reduce noise and improve the signal to noise ratio. In addition, an intelligent data acquisition approach was implemented to further reduce power consumption and memory capacity requirement. An Android user interface was developed for functional validation and parameters set up for sensing node and router units.

Prototypes were designed and built. The physical dimensions of the sensing node platform were 35mm x 38mm x 10 mm. Experiments were conducted to validate the node platform's sensitivity, crosstalk, accuracy, and current consumption at different modes. The sensing node platform had good sensitivity and accuracy in order to detect a sub-ohm resistance change on the 3D MEMS strain-gauge type sensor. Only 6.45mA and 2.61mA were consumed in active mode and sleep mode, respectively. Experiments were also performed and found the maximum wireless communication range from the sensing node to the gateway was 35.75m. Finally, the entire system was validated and confirmed that the sensing node could detect resistance changes of all 17 channels and could operate over seven days using a non-rechargeable 1200mA lithium battery.

The main contributions of this research were to develop a low-power, small physical dimensions IoT system that could interface with the MEMS strain gauge, which was developed by a team in mechanical engineering at the University of Alberta. The MEMS strain gauge can measure 3D stress and strain values. The power consumption of the acquisition unit was 24mW in active mode and 9.4mW in sleep mode, respectively. The results indicated that by implementing the dynamic sampling approach, 52% of the current consumption and 68% of memory were saved for the ten triggered per day scenario. This research also provides a proof of concept that the communication range can be extended using a router. Table 6-1 summarizes the specifications of the developed system and compared with the requirements.

Table 6-1 The summary of the specifications

Requirements	Specifications
Temperature range: -40°C to +70°C	Temperature range: -40°C to +125°C
Low power consumption	An average of 14.0mW
Last for at least 7 days	Able to last for 10 days
Sampling rate up to 20Hz	Dynamic sampling rate from 5Hz to 20Hz (which can be expanded in the future)
Small physical dimensions	35mm x 38mm x 10 mm (the acquisition platform)
Able to detect sub-ohm resistance changes	Sensitivity of 0.51Ω Accuracy of 0.58 Ω

6.2 Future Recommendations

Although most of the functionalities of the IoT system have been validated, the performance of the system under extreme temperature conditions has not been verified. This should be incorporated in the future since the structural health monitoring system may be implemented in some harsh environments. So far, all the experiments were conducted at room temperature. To investigate the temperature effect on the ADC sensitivity and accuracy, repeat the sensitivity and crosstalk of ADC experiments discussed in Section 5.3.2 with a different temperature. The change in temperature could be achieved by placing the boards into a temperature chamber.

Another aspect is to extend the communication range between a node to router and router to the gateway. A better antenna network should be designed on both the sensing node and the router platform without increasing their physical dimensions. Simulations or a network analyzer can be performed to tune the LC impedance matching circuits and ensure the most extensive range is obtained.

Also, experimental examinations in Chapter 5 focused on the single sensing node scenario. The system with multiple sensing nodes has not been tested. Experiments should be conducted in the future to determine whether a sensing node's transmission influences the other one. A multiple routers system should also be developed to further extend the communication distance between node and gateway.

At this stage, resistor values were collected. In the future, the sensor should be attached to a beam and tested with different loads. The collected resistor values should be converted into stress-strain values and interpreted using AI data processing and analysis techniques.

Furthermore, the current PCBs are two layers boards only, and components are populated on the top layer. The physical dimensions can be further reduced by developing a multi-layer PCB and populating components on both sides.

References

[1] I. Lombillo, H. Blanco, J. Pereda, L.Villegas, C. Carrasco and J. Balbás, “Structural health monitoring of a damaged church: design of an integrated platform of electronic instrumentation, data acquisition and client/server software,” Structural Control and Health Monitoring, vol. 23, no. 1, pp. 69-81, 2016.

[2] H. Lu, T. Iseley, S. Behbahani and L. Fu, “Leakage detection techniques for oil and gas pipelines: State-of-the-art.” Tunnelling and Underground Space Technology, vol. 98, pp. 103249, 2020. Available: 10.1016/j.tust.2019.103249.

[3] H. M. H. Gharib, “An Investigation of Using n-Si Piezoresistive Behavior to Develop a Three-Dimensional Stress Sensing Rosette,” Ph.D. dissertation, Dept. Mech. Eng., University of Alberta, Edmonton, Canada, 2013.

[4] V. Annamdas and M. Radhika, “Electromechanical impedance of piezoelectric transducers for monitoring metallic and non-metallic structures: A review of wired, wireless and energy-harvesting methods.” Journal of Intelligent Material Systems and Structures, vol. 24, no. 9, pp. 1021-1042, 2013. Available: 10.1177/1045389x13481254.

[5] D. Posenato, F. Lanata, D. Inaudi and I. F. C. Smith, “Model-free data interpretation for continuous monitoring of complex structures,” Adv. Eng. Inform., vol. 22, no. 1, pp. 135–144, 2008.

[6] H. Salehi, R. Burgueño, S. Das, S. Biswas and S. Chakrabartty, “Structural health monitoring from discrete binary data through pattern recognition,” Insights and Innovations in Structural Engineering, Mechanics and Computation, pp. 1840–1845, 2016.

[7] H. Salehi, S. Das, S. Chakrabartty, S. Biswas and R. Burgueño, “Structural damage identification using image-based pattern recognition on event-based binary data generated from self-powered sensor networks,” Structural Control and Health Monitoring, vol. 25, no. 4, p. e2135, 2018. Available: 10.1002/stc.2135.

[8] A. H. Alavi, H. Hasni, N. Lajnef and S. Masri, “A novel self-powered approach for structural health monitoring.” 2015 International Conference on Sustainable Mobility

Applications, Renewables and Technology (SMART), Kuwait, pp. 1-6, 2015. Available: 10.1109/SMART.2015.7399231.

[9] E. Z. Moore, J. M. Nichols and K. D. Murphy, “Model-based SHM: Demonstration of identification of a crack in a thin plate using free vibration data,” Mechanical systems and signal processing, vol. 29, pp. 284–295, 2012. Available: 10.1016/j.ymssp.2011.09.022.

[10] G. Jayavardhana, B. Rajkumar, M. Slaven and P. Marimuthu, “Internet of things (IoT): a vision architectural elements and future directions,” Elsevier Future Generation Computer Systems, vol. 29, no. 7, pp. 1645-1660, 2013.

[11] B. Glišić and D. Inaudi, Fiber optic methods for structural health monitoring, Chichester, England: John Wiley & Sons, pp. 19-40, 2007.

[12] Y. Liu, D. N. Wang and W. P. Chen, “Crescent shaped Fabry-Perot fiber cavity for ultra-sensitive strain measurement,” Scientific Reports, vol. 6, no. 1, 2016. Available: 10.1038/srep38390.

[13] B.H. Lee, Y.H. Kim, K.S. Park, J.B. Eom, M.J. Kim, B.S. Rho and H.Y. Choi, “Interferometric Fiber Optic Sensors,” Sensors, vol. 12, no. 3, pp. 2467-2486, 2012. Available: 10.3390/s120302467.

[14] K. Hill and G. Meltz, “Fiber Bragg grating technology fundamentals and overview,” Journal of Lightwave Technology, vol. 15, no. 8, pp. 1263-1276, 1997, Available: 10.1109/50.618320.

[15] J. Kumar and D. Chack, “FBG based strain sensor with temperature compensation for structural health monitoring,” 2018 4th International Conference on Recent Advances in Information Technology (RAIT), Dhanbad, pp. 1-4, 2018.

[16] Z. Zhou, Z. Wang and L. Shao, “Fiber-reinforced polymer-packaged optical fiber Bragg grating strain sensors for infrastructures under harsh environment,” Journal of Sensors, vol. 2016, pp. 1–18, 2016. Available: 10.1155/2016/3953750.

[17] V. M. Murukeshan, P. Y. Chan, L. S. Ong and A. Asundi, “Intracore fiber Bragg gratings for strain measurement in embedded composite structures,” Applied Optics, vol. 40, no. 1, pp. 145-149, 2001. Available: 10.1364/ao.40.000145.

[18] V.P. Matveenko, I. Shardakov, A.A. Voronkov, N. Kosheleva, D.S. Lobanov, G. Serovaev, E.M. Spaskova and G.S. Shipunov, “Measurement of strains by optical fiber Bragg grating sensors embedded into polymer composite material,” Structural Control and Health Monitoring, vol. 25, no. 3, p. e2118, 2017. Available: 10.1002/stc.2118.

[19] D. Kinet, P. Mégret, K. W. Goossen, L. Qiu, D. Heider and C. Caucheteur, “Fiber Bragg grating sensors toward structural health monitoring in composite materials: challenges and solutions,” Sensors (Basel), vol. 14, no. 4, pp. 7394–7419, 2014. Available: 10.3390/s140407394.

[20] J. Sirohi and I. Chopra, “Fundamental understanding of piezoelectric strain sensors.” Journal of intelligent material systems and structures, vol. 11, no. 4, pp. 246-257, 2000. Available: 10.1106/8bfb-gc8p-xq47-ycq0.

[21] T. Kobayashi, T. Yamashita, N. Makimoto, S. Takamatsu and T. Itoh, “Ultra-thin piezoelectric strain sensor 5 × 5 array integrated on flexible printed circuit for structural health monitoring by 2D dynamic strain sensing,” 2016 IEEE 29th International Conference on Micro Electro Mechanical Systems (MEMS), Shanghai, pp. 1030-1033, 2016.

[22] S. Ying, W. Zhichen and Du Yanliang, “Theoretical and Experimental Research on Piezoelectric Sensors Response to Dynamic Strain,” 2007 8th International Conference on Electronic Measurement and Instruments, pp. 4-194-4-198, 2007. Available: 10.1109/ICEMI.2007.4351115.

[23] F. G. Baptista, D. E. Budoya, V. A. D. de Almeida and J. A. C. Ulson, “An experimental study on the effect of temperature on piezoelectric sensors for impedance-based structural health monitoring,” Sensors, vol. 14, no. 1, pp. 1208-1227, 2014. Available: 10.3390/s140101208.

[24] S. Roy, P. Ladpli and F.-K. Chang, “Load monitoring and compensation strategies for guided-waves based structural health monitoring using piezoelectric transducers,” Journal of Sound and Vibration, vol. 351, pp. 206–220, 2015. Available: 10.1016/j.jsv.2015.04.019.

[25] Y. Kim, Y. Kim, C. Lee and S. Kwon, “Thin Polysilicon Gauge for Strain Measurement of Structural Elements,” IEEE Sensors Journal, vol. 10, no. 8, pp. 1320–1327, 2010. Available: 10.1109/jsen.2009.2039565.

[26] P.G.S. “Jackson, Improvement in or relating to strain gauges.” British Patent Specification 720,325. Application Date Aug. 21, 1952.

[27] “Gauge factor,” En.wikipedia.org, 09-Nov-2020. [Online]. Available: https://en.wikipedia.org/wiki/Gauge_factor. [Accessed: 01-Nov-2021].

[28] “Strain Gage: Sensitivity,” Efunda.com, 2021. [Online]. Available: https://www.efunda.com/designstandards/sensors/strain_gages/strain_gage_sensitivity.cfm. [Accessed: 01- Nov- 2021].

[29] J. K. Sell, H. Enser, M. Schatzl-Linder, B. Strauß, B. Jakoby and W. Hilber, “Nested, meander shaped strain gauges for temperature compensated strain measurement,” 2017 IEEE SENSORS, Glasgow, pp. 1-3, 2017. Available: 10.1109/ICSENS.2017.8234099.

[30] D. Gräbner, E. Meyer and W. Lang, “Foil-based strain gauges with nanogranular platinum structures for the integration in elastomer gaskets,” 2016 IEEE Sensors Conference, Orlando, FL, pp. 1-3, 2016. Available: 10.1109/ICSENS.2016.7808844.

[31] T. Toriyama and S. Sugiyama, “100 nm square single crystal silicon shear strain gauge,” Proceedings of 2002 International Symposium on Micromechatronics and Human Science, Nagoya, Japan, pp. 223-226, 2002. Available: 10.1109/MHS.2002.1058039.

[32] H. Kuo, J. Guo and W. H. Ko, “High Performance Piezoresistive Micro Strain Sensors,” 2007 2nd IEEE International Conference on Nano/Micro Engineered and Molecular Systems, Bangkok, pp. 1052-1055, 2007. Available: 10.1109/NEMS.2007.352199.

[33] S. P. Olson, J. Castracane and R. E. Spoor, “Piezoresistive strain gauges for use in wireless component monitoring systems,” 2008 IEEE Sensors Applications Symposium, Atlanta, GA, pp. 36-39, 2008. Available: 10.1109/SAS13374.2008.4472939.

[34] Y. Kim, Y. Kim, C. Lee and S. Kwon, “Thin Polysilicon Gauge for Strain Measurement of Structural Elements,” in IEEE Sensors Journal, vol. 10, no. 8, pp. 1320-1327, 2010. Available: 10.1109/JSEN.2009.2039565.

[35] H. Miura, A. Nishimura, S. Kawai and G. Murakami, “Structural Effect of Ic Plastic Package on Residual-Stress in Silicon Chips,” 40th Electronic Components and Technology Conference, Vols 1 and 2, pp. 316-321, 1990.

[36] D. A. Bittle, J. C. Suhling, R. E. Beaty and R. C. Jaeger, “Piezoresistive stress sensors for structural analysis of electronic packages,” Journal of Electronic Packaging, vol. 113, no. 3, pp. 203-215, 1991. Available: 10.1115/1.2905397.

[37] J. C. Suhling and R. C. Jaeger, “Silicon piezoresistive stress sensors and their application in electronic packaging,” Sensors Journal, IEEE, vol. 1, no. 2, pp. 14-30, 2001. Available: 10.1109/jsen.2001.923584.

[38] M. Kayed, A. Balbola, E. Lou and W. Moussa, “Development of MEMS-based piezoresistive 3D stress/strain sensor using strain technology and smart temperature compensation,” Journal of Micromechanics and Microengineering, vol. 31, no. 3, p. 035010, 2021. Available: 10.1088/1361-6439/abdbd6.

[39] A. Balbola A, M. Kayed, E. Lou and W. Moussa, “Development of a Self-Monitored 3D Stress Sensor for Multilayer Assemblies Monitoring,” IEEE Sensor Journal, vol. 20, no. 24, pp. 14676-14684, 2020. Available:10.1109/JSEN.2020.3010503.

[40] A. S. Algamili, M. H. M. Khir, J. O. Dennis, A. Y. Ahmed, S. S. Alabsi, S. S. B. Hashwan and M. M. Junaid, “A Review of Actuation and Sensing Mechanisms in MEMS-Based Sensor Devices,” Nanoscale Research Letters, vol. 16, no. 1, 2021. Available: 10.1186/s11671-021-03481-7.

[41] S. Pandey, H. Mohammad and U. Nasim, “Design and implementation of a low-cost wireless platform for remote bridge health monitoring,” Int. J. Emerg. Technol. Adv. Eng., vol. 6, no. 6, pp. 57-62, 2016.

[42] H. Malik, K. S. Khattak, T. Wiqar, Z. H. Khan and A. B. Altamimi, “Low Cost Internet of Things Platform for Structural Health Monitoring,” 2019 22nd International

Multitopic Conference (INMIC), pp. 1-7, 2019. Available: 10.1109/INMIC48123.2019.9022801.

[43] M. A. Mahmud, K. Bates, T. Wood, A. Abdelgawad and K. Yelamarthi, "A complete Internet of Things (IoT) platform for Structural Health Monitoring (SHM)," 2018 IEEE 4th World Forum on Internet of Things (WF-IoT), 2018. Available: 10.1109/WF-IoT.2018.8355094.

[44] A. Girolami, D. Brunelli and L. Benini, "Low-cost and distributed health monitoring system for critical buildings," 2017 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems (EESMS), Milan, pp. 1-6, 2017.

[45] L. Sun, B. Sun and Y. Huang, "A Hardware Platform for Wireless Structural Health Monitoring Based on Compressed Sensing," 2018 13th World Congress on Intelligent Control and Automation (WCICA), Changsha, China, pp. 159-164, 2018.

[46] S. Gao, X. Dai, Z. Liu and G. Tian, "High-Performance Wireless Piezoelectric Sensor Network for Distributed Structural Health Monitoring," International Journal of Distributed Sensor Networks, vol. 12, no. 3, p. 3846804, 2016. Available: /10.1155/2016/3846804.

[47] L. Sadoudi, M. Bocquet, E. Moulin and J. Assaad, "ZigBee Sensor Network Platform for Health Monitoring of Rails Using Ambient Noise Correlation," J. of Electrical Engineering, vol. 5, no. 3, pp. 143-150, 2017. Available: 10.17265/2328-2223/2017.03.004.

[48] X. Liu, J. Cao and P. Guo, "SenetSHM: Towards Practical Structural Health Monitoring Using Intelligent Sensor Networks," 2016 IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom) (BDCloud-SocialCom-SustainCom), Atlanta, GA, pp. 416-423, 2016.

[49] E. Setijadi, B. P. Slamet, A. A. Muntaqo, A. E. Nur, P. Suprobo and F. A. Febry, "Design of large-scale structural health monitoring system for long-span bridges based on wireless sensor network," 2013 International Joint Conference on Awareness Science and

Technology & Ubi-Media Computing (iCAST 2013 & UMEDIA 2013), pp. 169-174, 2013. Available: 10.1109/ICAwST.2013.6765428.

[50] K. Gürkan, G. Gürkan and A. A. Dindar, "Design and realization of multi-channel wireless data acquisition system for laboratory-scale experiments on structural health monitoring," Journal of Measurements in Engineering, vol. 6, no. 1, pp. 64–73, 2018. Available: 10.21595/jme.2018.19699.

[51] M. A. Baraa, "Development of Health Monitoring System in Highway Bridges," 2019 International Conference on Wireless Communication, Network and Multimedia Engineering (WCNME 2019), 2019. Available:10.2991/wcnme-19.2019.53.

[52] Y. Chen, "Design and Implementation of Wireless Sensor Cellular Network Based on Android Platform," International Journal of Online Engineering (iJOE), vol. 13, no. 05, p. 56, 2017. Available: 10.3991/ijoe.v13i05.7048.

[53] P. Paul, N. Dutta, B. A. Biswas, M. Das, S. Biswas, Z. Khalid and H. D. Saha, "An Internet of Things (IoT) Based System to Analyze Real-time Collapsing Probability of Structures," 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), pp. 1070-1075, 2018. Available: 10.1109/IEMCON.2018.8614743.

[54] "10 Major Differences - Bluetooth 5 vs 4.2 - Feature Comparisons," Amar Infotech, 01-Sep-2021. [Online]. Available: https://www.amarinfotech.com/differences-comparisons-bluetooth-5-vs-4-2.html. [Accessed: 14-Oct-2021].

[55] A. A. Balbola, M. O. Kayed, E. Lou and W. A. Moussa, "Testing of a Strained Silicon Based 3-D Stress Sensor for Out-of-Plane Stress Measurements," IEEE/ASME Transactions on Mechatronics, vol. 26, no. 2, pp. 1076-1083, 2021. Available: 10.1109/TMECH.2020.3016294.

[56] Battery University, "What's the Best Battery?" Battery University, 04-Mar-2020. [Online]. Available: https://batteryuniversity.com/article/whats-the-best-battery. [Accessed: 20-Oct-2021].

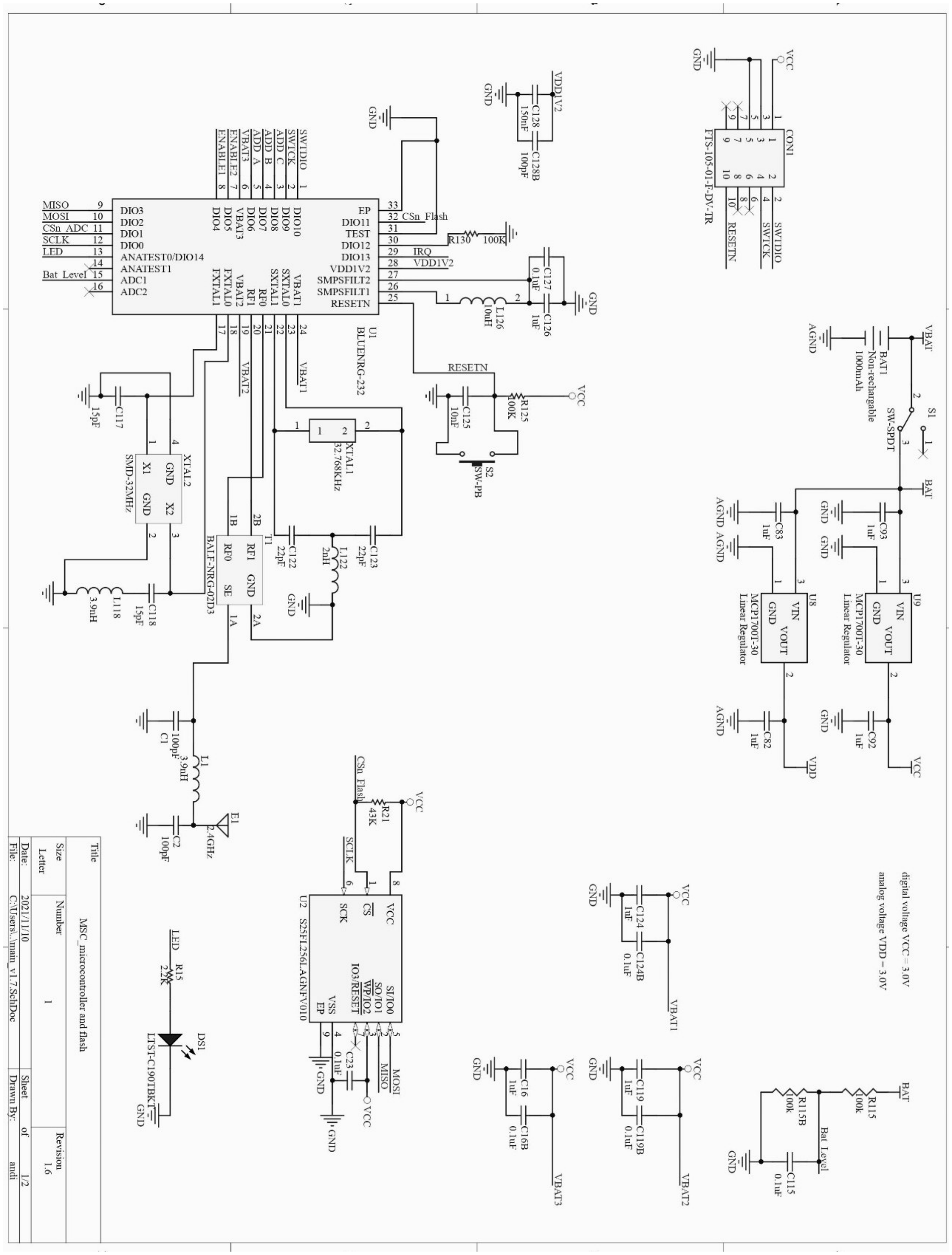
[57] Eveready," BATTERY ZINC 1.5V AA," 1215NA0218 datasheet.

[58] FDK CORPORATION, “AAA size/ LR03C Universal Power,” LR03C datasheet.

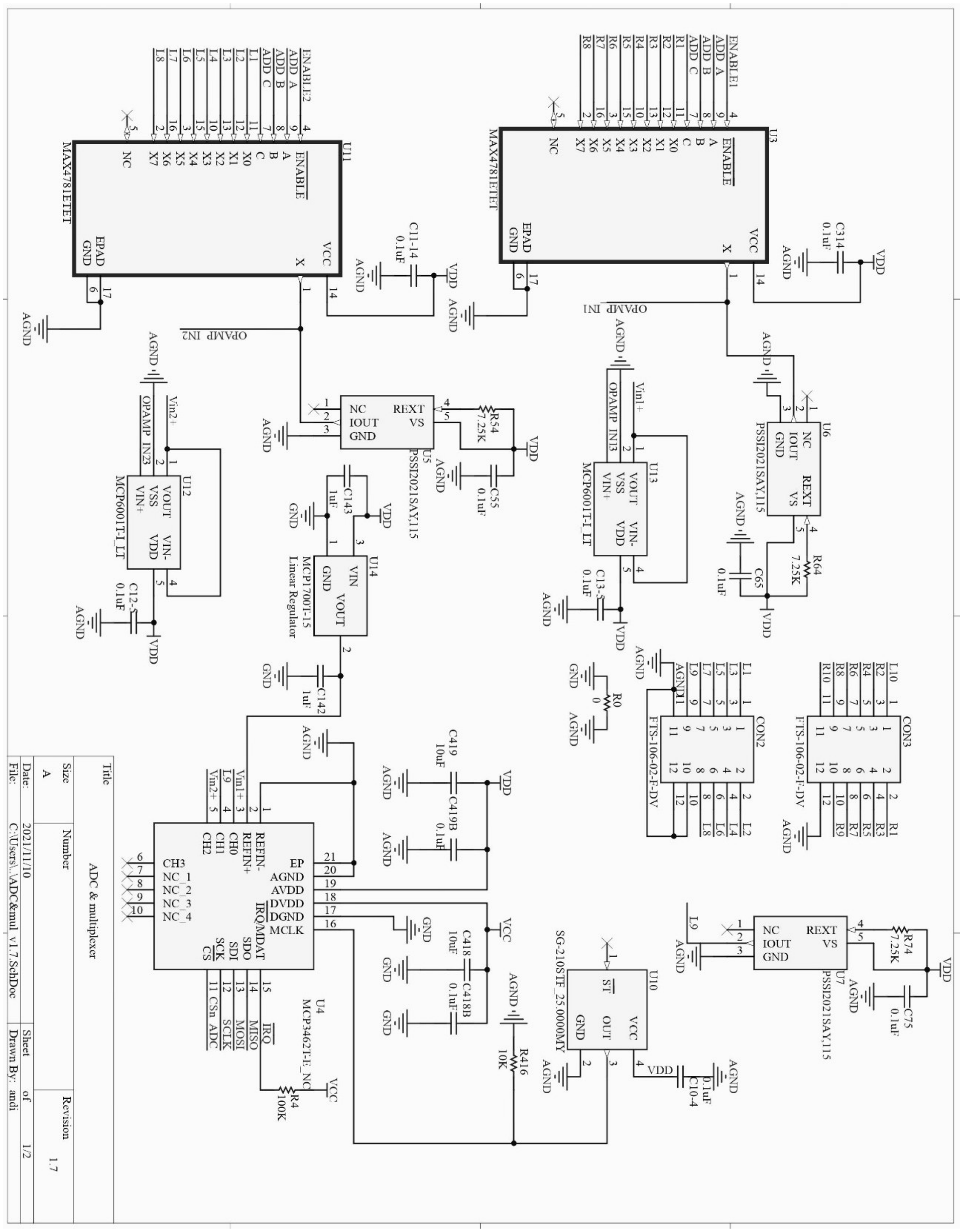
[59] PKCELL, “Primary Lithium Battery Technology Specification,” ER14250 datasheet.

[60] Raspberry Pi, “Raspberry Pi 4 Model B Product Brief,” Raspberry Pi 4 Computer Model B datasheet.

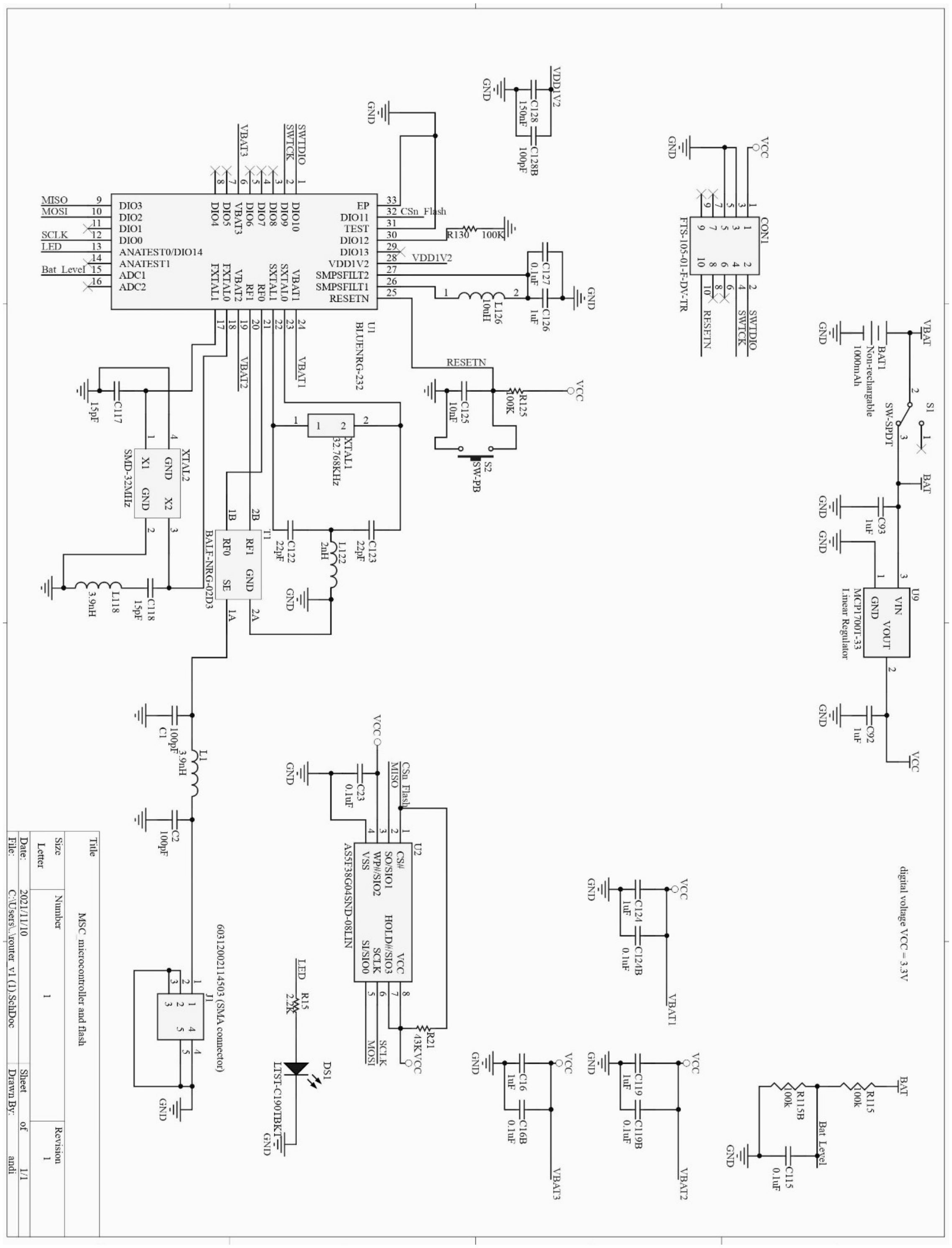
Appendix A



Appendix A-1: Data acquisition platform schematic (sheet 1)



Appendix A-2: Data acquisition platform schematic (sheet 2)



Appendix A-3: Router platform schematic (sheet 1)