2019

Cloud-based multi-sensor remote data acquisition system for precision agriculture (CSR-DAQ)

Jiztom Kavalakkatt Francis
Iowa State University

Recommended Citation

Kavalakkatt Francis, Jiztom, "Cloud-based multi-sensor remote data acquisition system for precision agriculture (CSR-DAQ)" (2019). *Graduate Theses and Dissertations*. 17715.
https://lib.dr.iastate.edu/etd/17715

Cloud-based multi-sensor remote data acquisition system for precision agriculture (CSR-DAQ)

by

Jiztom Kavalakkatt Francis

A thesis submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: Computer Engineering (Computing and Networking Systems)

Program of Study Committee:
Manimaran Govindarasu, Major Professor
Matthew Darr
Ajay Nair

Iowa State University

Ames, Iowa

2019

DEDICATION

I would like to dedicate this thesis to my parents Dr. K Francis Jacob and Binie Francis without whose support, I would not have been able to pursue and complete this work.

I would also like to thank my friends, and family for their loving guidance and financial assistance during the writing of this work. I would also extend my gratitude to my colleagues and mentors who gave guidance and ideas to simplify the process.

TABLE OF CONTENTS

Page

LIST OF TABLES v

LIST OF FIGURES vi

NOMENCLATURE vii

ACKNOWLEDGMENTS viii

ABSTRACT ix

CHAPTER 1. INTRODUCTION TO PRECISION AGRICULTURE 1
  1.1 Overview 2
    1.1.1 Internet of Things 2
    1.1.2 Volumetric Water Content 6
    1.1.3 Data Acquisition Systems (DAQ) 8
  1.2 Research Objective 8
  1.3 Thesis Organization 9

CHAPTER 2. LITERATURE AND INDUSTRY REVIEW 10
  2.1 Literature Review 11
  2.2 Industry grade prototype review 13
    2.2.1 Active Systems 14
    2.2.2 Passive Systems 18
    2.2.3 Comparison 21

CHAPTER 3. PROPOSED ARCHITECTURE OF CSR-DAQ 23
  3.1 Design Goals 23
  3.2 Test bed Overall Architecture 24
  3.3 Node 27
    3.3.1 Hardware Design 27
  3.4 Aggregator / Base station 34
    3.4.1 Firmware 35
  3.5 Cloud Database 36
    3.5.1 MQTT 36
    3.5.2 Amazon AWS RDS 38
    3.5.3 Client Application 39

CHAPTER 4. TESTING AND EVALUATION 40
4.1 Implemented Hardware 40
4.2 Testing 42
4.2.1 Hardware and range test 42
4.2.2 Greenhouse Data collection 44
4.2.3 Outdoor Field Test 45
4.2.4 Greenhouse Regression Test 47
4.3 Power Analysis 53
4.4 Cost Evaluation 57

CHAPTER 5. CONCLUSION 59
5.1 Summary 59
5.2 Future Work 62

BIBLIOGRAPHY 63

APPENDIX. ADDITIONAL MATERIAL 66

LIST OF TABLES

Page
Table 2.1 Comparison of proposed model with existing literature 13
Table 2.2 Comparison table with industry standards and proposed hardware 22
Table 3.1 Modules of proposed CSR-DAQ 25
Table 3.2 Available Compatible Technology 35
Table 4.1 Testing Phases with Location and Duration 42
Table 4.2 Cost of Node 58
Table 4.3 Cost of Entire System 58
Table 5.1 Updated Comparison table with industry standards and proposed hardware 61

LIST OF FIGURES

Page
Figure 1.1 Precision Agriculture Model 2
Figure 1.2 An example of IoT Ecosystem 3
Figure 1.3 An example Soil structure 6
Figure 2.1 Categories of research Focus on Precision Agriculture 10
Figure 2.2 Onset Hobonet Field monitoring system 15
Figure 2.3 PYCNO Core - All in-one soil sensors 16
Figure 2.4 Wireless Soil Moisture Sensor Long Range IoT Transmitter 17
Figure 2.5 Decagon - em5b Data-logger 19
Figure 2.6 Delta T DL6 Soil Moisture Data Logger 20
Figure 3.1 End-to-End Architecture of the CSR-DAQ 26
Figure 3.2 Overall Node Architecture 28
Figure 3.3 Buck Circuit Diagram 30
Figure 3.4 3.5 mm analog jack design and ADC socket 31
Figure 3.5 ADS1115 4 input ADC unit 32
Figure 3.6 ATMEGA328P Pinout [1] 33
Figure 3.7 Node Wake up Cycle 34
Figure 3.8 Basex function architecture 36
Figure 3.9 MQTT payload format 39
Figure 4.1 Test-bed and Field Location 41
Figure 4.2 Phase 1 Prototype 43
Figure 4.3 Test Bed set at the greenhouse 45
Figure 4.4 Base station Deployed 46
Figure 4.5 Phase 3 Field Site with deployed sensors 47
Figure 4.6 Phase 4 base station 48
Figure 4.7 Phase 4 Node 49
Figure 4.8 Phase 4 Test Bed Setup 50
Figure 4.9 VWC Calibration of Proposed Data logger with Commercial Data logger (Plant 1) 51
Figure 4.10 VWC Calibration of Proposed Data logger with Commercial Data logger (Plant 2) 52
Figure 4.11 VWC Regression analysis for plant 2 54
Figure 4.12 VWC Scatter Plot with Plant1 55
Figure 4.13 VWC scatter Plot with Plant 2 56
Figure 4.14 Power Drain vs Count 57

NOMENCLATURE

ADC	Analog-Digital Converter
AWS	Amazon Web Services
CSP	Cloud Service Provider
Cloud-IoT	Cloud Based Internet of Things
DAQ	Data Acquisition System
DAC	Digital Analog Converter
IDE	Integrated Development Environment
IoT	Internet of Things
ISPA	Internation Society of Precision Agriculture
LAN	Local Area Network
LoRa	Long Range
LPWAN	Low-Power Wide-Area Network
MQTT	Message Queuing Telemetry Transport
NIST	National Institute of Standards and Technology
RDS	Relational Data Base System
RPi	Raspberry Pi - System on Chip (SoC)
SoC	System on Chip
TCP IP	Transmission Control Protocol/Internet Protocol
UDP	User Data gram Protocol
UID	Unique Identifiers
WAN	Wide Area Network
WSN	Wireless Sensor Network

ACKNOWLEDGMENTS

I would like to take this opportunity to express my thanks to those who helped me with various aspects of conducting research and the writing of this thesis. First and foremost, Dr. Manimaran Govindarasu for his valuable guidance, patience, and support throughout this research and the writing of this thesis. His deep insights and words of encouragement have often inspired me and renewed my hopes for completing the research thesis for my graduate study.

I would also like to express my sincere thanks to my committee members for their efforts and contributions to this research work a reality: Dr. Matthew Darr and Dr. Ajay Nair.

I would also like to extend my heartfelt thanks to Leeland Harker and Chris Saleh in their wholehearted support during the design of the Sensor nodes. My sincere thanks to Dr. Ravikumar Gelli for his valuable support.

Several key ideas came into the picture more vividly with the great discussion I had with several people who influenced the growth and development of this research product. I remember all my well-wishers and friends for their suggestions, support, and prayers.

This list will be incomplete if I fail to acknowledge the support and constant encouragement from my parents and grandparents during my work for the thesis. Their great interest, supreme sacrifices, and deep love made this venture a great success.

The last but not least, I am deeply indebted to all graces showered from above. But for His Wishes, this would have never been a reality, It is for all of this, I dedicate this dissertation to God Almighty.

ABSTRACT

Many of the current agriculture systems have deployed analog/digital sensors to measure crop monitoring, weather forecast, and environmental sensor data. The significant problems of the current agriculture system are 1) the inability to combine the collected sensor data into useful information for farmers to make the right decision to optimize the crop produce; 2) legacy infrastructure and manual data collection; 3) lack of scalability and incompatibility due to the vendor-dependent sensors and legacy data loggers. With the advent of the Internet of Things (IoT), the ad-hoc and traditional agriculture systems adopt precision agriculture methods to improve the quality and quantity of harvest. To realize such precision agriculture methods in Smart farming, we require a platform that collects the sensor data, processes it into information and helps in visualizing the results. The existing custom-made prototype solutions and the industry-grade data acquisition systems are expensive and have limited functionalities to realize the precision agriculture methods.

In this thesis, we propose an architecture and testbed-based implementation for a cost-effective active data acquisition system that can autonomously collect, transmit, and process the raw data. The proposed architecture includes four modules - Nodes, Aggregators, Cloud-based Database, and Client-side applications. The functionalities of these modules are 1) Node collects sensor data at specified intervals and transmits the sensor data streams to the aggregator; 2) Aggregator executes a data serializer for converting the sensor data streams, buffer for local storage, and data transmitter for sending them to the cloud-based database system; 3) Cloud-based Database is hosted on Amazon Relational Database Services (RDS) and uses Postgres SQL to facilitate multiple reads, write, and no overwrite functionality; and 4) Client-side applications include web pages, mobile apps, and services that communicate the cloud-based database system for the field sensor data.

The test-bed was set up at the Iowa State University greenhouse environment to read controlled environmental data. Collected data from a commercial sensor validated the measurements

as a benchmark tool. The end-to-end test setup and obtained results were congruent with the design specifications and satisfied the user requirements. Analog sensors with the proper specifications are compatible with the proposed hardware to read environmental data without additional modifications. Field test implementation also successfully validated the design with real-time data collection. The results with the VWC from measured sensors have 98% R^2 values on performing linear regression. Battery optimization was also found to allow the Data-logger to work for an entire harvest season. Thus, CSR-DAQ solves the need for smart systems for small-scale farmers by providing them active data acquisition units at cost-effective budgets and allows them to make a decision or automate certain parts of farming such as irrigation and fertilizer control.

CHAPTER 1. INTRODUCTION TO PRECISION AGRICULTURE

The modern day world has made life simple with apps and connected devices. With the growing population and its dependence on technology, it is a need that agricultural practices be upgraded to keep with the demand. Precision Agriculture solves this issue by blending the latest technologies with farming practices to obtain the optimum harvest with maximum efficiency. To develop such systems, it requires lots of data from sensors out in the environment and methods of understanding such data. The aforementioned brings out the need for effective mechanisms to collect the data for future use. Current solutions cater to the large scale farming or hobby farmers with limited options for small-to-mid scale farmers. With the availability of Cloud-based services such as Amazon AWS RDS and Wireless Sensor Network (WSN) it becomes possible to provide simple solutions to collect data based on existing sensors in the field. The proposed solutions allow users to remotely collect data from their fields and monitor them remotely without the being at its proximity. Customized technology flavors allow for maximum efficiency of data collection with the transmission range from a few meters up to 10+ miles. The field testing performed proves the feasibility of the design. Comparative market values determine it to be an optimum solution for precision agriculture (Figure 1.1).

As per ISPA [2] we can define Precision Agriculture as follows:

> *Precision Agriculture is a management strategy that gathers, processes and analyzes temporal, spatial and individual data and combines it with other information to support management decisions according to estimated variability for improved resource use efficiency, productivity, quality, profitability and sustainability of agricultural production.*

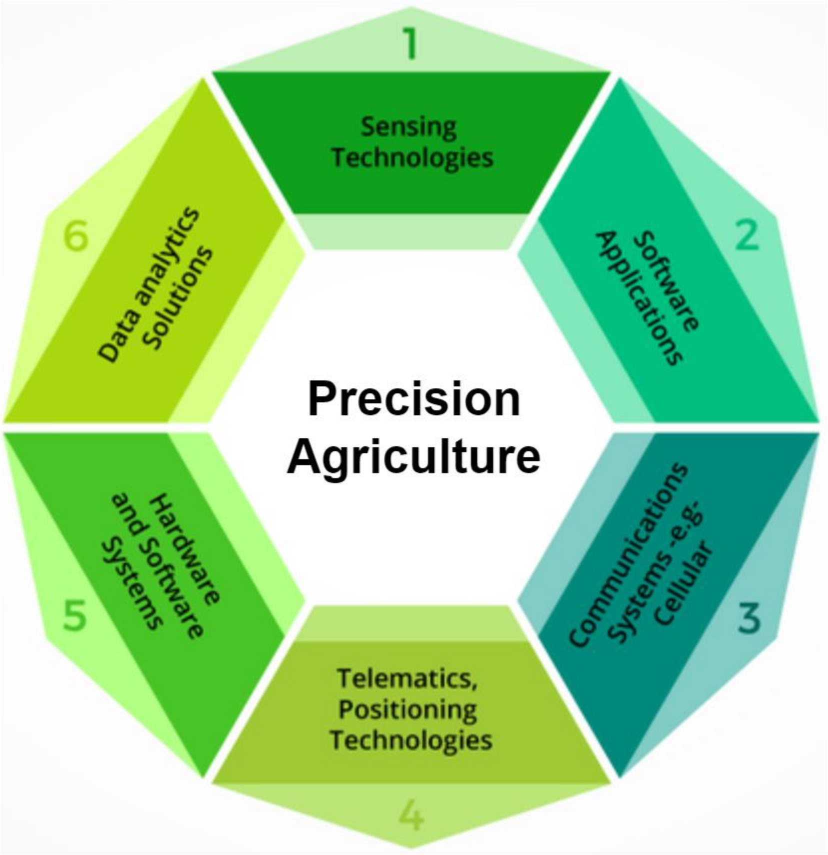


Figure 1.1 Precision Agriculture Model

1.1 Overview

This section briefly explains the initial concepts and conditions are explained and several hypothesis are mentioned in brief.

1.1.1 Internet of Things

As per the Internet of things Agenda [3], IoT is defined as follows:

> *The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers*

> *(UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction.*

A thing in the internet of things can be a person with a heart monitor implant, a farm animal with a bio-chip transponder, an automobile that has built-in sensors to alert the driver when tire pressure is low or any other natural or man-made object that can be assigned an IP address and is able to transfer data over a network.

Increasingly, organizations in a variety of industries are using IoT to operate more efficiently, better understand customers to deliver enhanced customer service, improve decision-making and increase the value of the business.

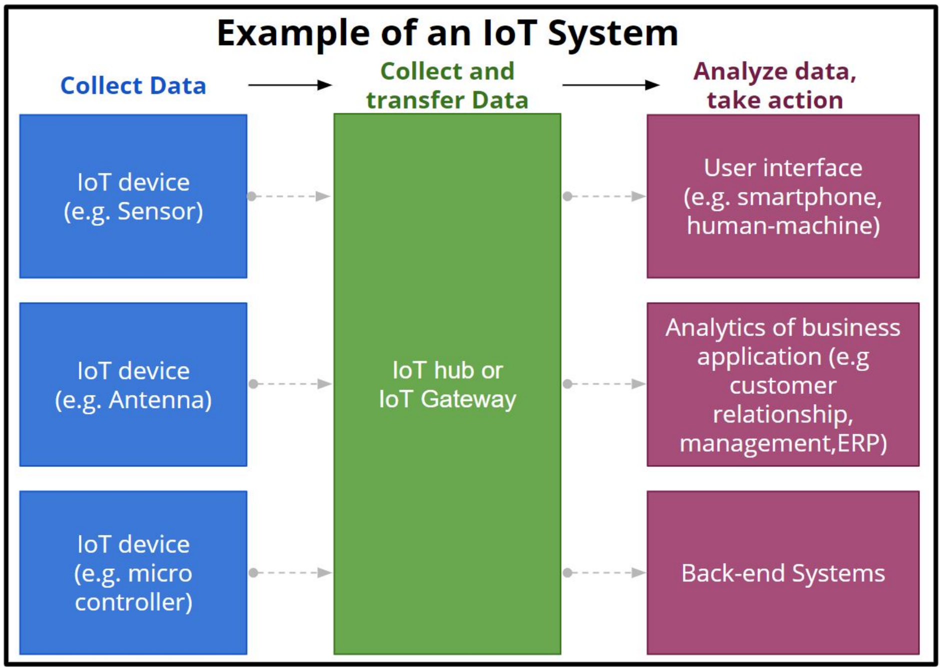


Figure 1.2 An example of IoT Ecosystem

1.1.1.1 History of IoT

The Jargon IoT [3] was introduced during the late 1990's to bring towards attention of using Radio Frequency ID (RFID)to the attention of the Proctor and Gamble (P&G) management as a cool marketing tool. The idea of connected devices comes from further back in the 1970's but under the names such as embedded internet or pervasive computing.

The first internet application was for example, was a Coke machine at Carnegie Mellon University in the early 1980s. This was to check if the vending machine had coke for them before physically walking towards it.

IoT evolved from machine-to-machine (M2M) communication, i.e., machines connecting to each other via a network without human interaction. M2M refers to connecting a device to the cloud, managing it and collecting data.

Taking M2M to the next level, IoT is a sensor network of billions of smart devices that connect people, systems and other applications to collect and share data. As its foundation, M2M offers the connectivity that enables IoT.

The internet of things is also a natural extension of SCADA (supervisory control and data acquisition), a category of software application program for process control, the gathering of data in real time from remote locations to control equipment and conditions. SCADA systems include hardware and software components. The hardware gathers and feeds data into a computer that has SCADA software installed, where it is then processed and presented it in a timely manner. The evolution of SCADA is such that late-generation SCADA systems developed into first-generation IoT systems.

1.1.1.2 How IoT Works

An IoT ecosystem (Figure 1.2) consists of web-enabled smart devices that use embedded processors, sensors and communication hardware to collect, send and act on data they acquire from their environments. IoT devices share the sensor data they collect by connecting to an IoT gateway or other edge device where data is either sent to the cloud to be analyzed or analyzed locally. Sometimes, these devices communicate with other related devices and act on the information they get from one another. The devices do most of the work without human intervention, although people can interact with the devices – for instance, to set them up, give them instructions or access the data.

The connectivity, networking and communication protocols used with these web-enabled devices largely depend on the specific IoT applications deployed.

1.1.1.3 Benefits of IoT

The internet of things offers a number of benefits to organizations, enabling them to:

- Monitor their overall business processes;
- Improve the customer experience;
- Save time and money;
- Enhance employee productivity;
- Integrate and adapt business models;
- Make better business decisions; and
- Generate more revenue.

IoT encourages companies to rethink the ways they approach their businesses, industries and markets and gives them the tools to improve their business strategies.

1.1.1.4 Pros and cons of IoT

- Some of the advantages of IoT include:
  - Ability to access information from anywhere at any time on any device;
  - Improved communication between connected electronic devices;
  - Transferring data packets over a connected network saves time and money;
  - Automating tasks helps improve the quality of a business services and reduces the need for human intervention.

- Some disadvantages of IoT include:
  - As the number of connected devices increases and more information is shared between devices, the potential that a hacker could steal confidential information also increases;
  - Enterprises may eventually have to deal with massive numbers maybe even millions of IoT devices and collecting and managing the data from all those devices will be challenging.
  - If theres a bug in the system, its likely that every connected device will become corrupted;
  - Since theres no international standard of compatibility for IoT, its difficult for devices from different manufacturers to communicate with each other.

1.1.2 Volumetric Water Content

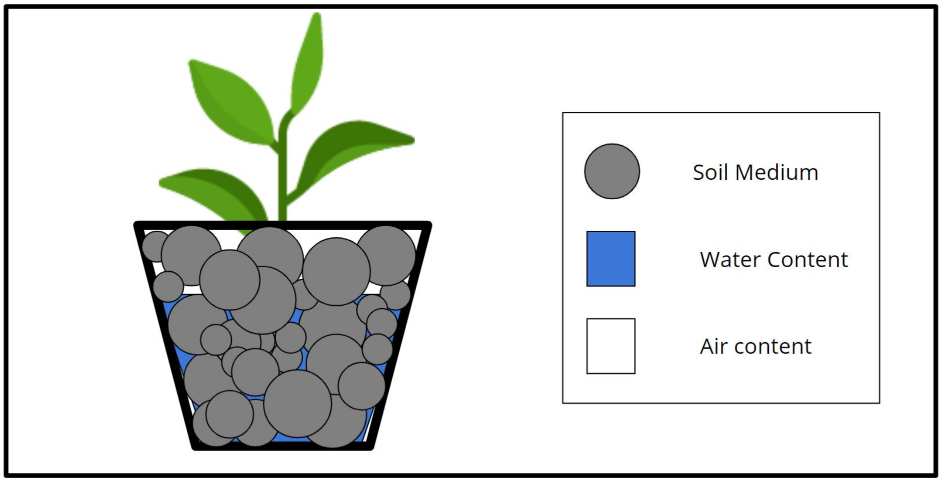


Figure 1.3 An example Soil structure

There is a need to know how wet the soil is in agriculture. This is a factor which directly affects the life of the plant. The measure of water within a limited sphere is used as a standard of measurement. The Figure 1.3 shows the general structure of the soil with respect to Volumetric contents. Soil has more sub-divisions but we will consider all solid material as medium in which the plant can grow. A well versed explanation for farmers is given in [4].

There are three kinds of states in this single example:

1. Solid : soil, nutrients, plant
2. Liquid : water, soluble compounds, etc.
3. Gaseous : Air pockets of different compounds.

A delicate balance of all three is required for any plant to grow. Water is the most important component for plant as its the medium it uses to take the nutrients present in the soil into itself. The air pockets also has their role in a plants life.

The way VWC [5] is defined is as follows:

"Percent of water content within a sphere of influence of soil. Measure either between 0 to 100 % or value between 0 and 1."

1.1.2.1 Direct measure of VWC:

The following is the steps to perform physical measurement of Volumetric Water Content on a soil sample:

- Obtain moist soil sample with known *volume*.
- Weigh moist sample.
- Dry sample at 105 °C for 24h.
- Weigh dry sample.

The Equation 1.1 shows the direct way to measure VWC with known volume.

$$\theta = \frac{M_{moist} - M_{dry}}{V_{sample}} \tag{1.1}$$

There are different sensors available in the market who measure soil moisture. The technology defines the metrics used to evaluate. The most scientific way to measure is using VWC. The accuracy, sphere of influence, resolution depends on the type of sensor and data logger used to perform the required measurement.

1.1.3 Data Acquisition Systems (DAQ)

Industries that make use of this technology include aerospace, medicine, wastewater services, and industrial manufacturing. Within these industries, operators of DAQ [6] software and instruments use them for tasks like data measurement, monitoring, and recording. In addition to allowing manufacturers to read data, data acquisition systems allow them to test a wide variety of technical products and make informed process adjustments. Operators can use DAQ software to input data about virtually anything, from gas pressure to voltage.

Typical data acquisition software applications include flight data acquisition, structural dynamics test systems, local electronic data capture systems, and data collection via PC-based systems and chart recorders. Another common data acquisition application is the use of central web-based systems for clinical trial data, such as the collection of wastewater toxicity and treatment response data. In general, operators most often use data acquisition systems to conduct measurement and testing for field studies, research, and product troubleshooting.

1.2 Research Objective

The advanced measuring tools existing in small to medium scale farming are mostly data-loggers with on-board storage. The current solution requires the user to be in proximity to measure and retrieve the data. This is executed by the user to setup a connection either via a wired connectivity or a short range wireless setup. The solution:

> **An economical and low-power reliable sensor node array system collects data and uploads it to the Cloud Database. Additionally a software at the user end allows them to scrutinize it remotely.**

1.3 Thesis Organization

The rest of the thesis is organized as follows:

- Chapter 2: A brief on the related areas of interest with related to Internet of Things, Precision Agriculture and Data logging systems.
- Chapter 3: The proposed architecture of CSR-DAQ explained in detail.
- Chapter 4: The phases of CSR-DAQ Implementation and results obtained.
- Chapter 5: Summarizing the results and explaining the possible areas of future research.

CHAPTER 2. LITERATURE AND INDUSTRY REVIEW

This chapter focuses on the various works of literature available on precision agriculture, data logging, and sensor deployment. All popular solutions for precision agriculture sensors and data loggers are analyzed. A research and industry gap was identified. Research constraints for the proposed architecture were parameterized based on the identified gap. During this process, various works presented in the literature are covered that improve precision agriculture.

The field of precision agriculture encompasses a variety of areas to address different issues faced by the farming community. Especially how the available technology can be applied to the farming practices to improve the quality and quantity of the produce. The major focus is on three categories shown in Figure 2.1:

1. Sensor integration.
2. Data collection/analysis.
3. Actuator control / feedback.

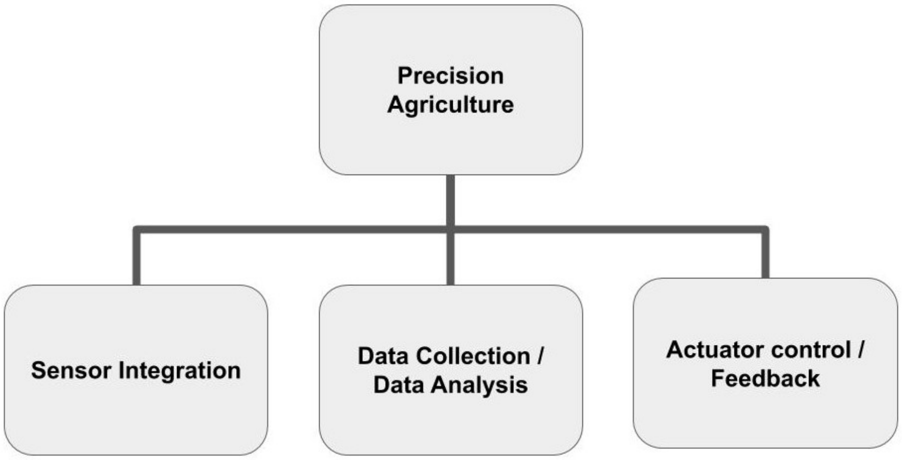


Figure 2.1 Categories of research Focus on Precision Agriculture

In the context of an IoT ecosystem, end-to-end data transmission with security essentially means how the data is collected, stored transmitted, and processed as information. The goal of the research is to provide a cost-effective solution for farmers to accumulate the required environmental data and visualize it remotely. Such a solution requires a study into existing solutions, factors that can affect sensors, methods, and areas of deployment, battery life, and durability.

2.1 Literature Review

The advantages of Wireless Sensor networks which form the core of the Internet of Things, allow users to measure, act and react to the physical world. The ease of creating a network with the available technologies have made communication and analysis of collected data. Much applied research has applied WSNs to multiple areas, such as environmental parameter monitoring, smart devices, smart cities, agriculture control, and other uses. The wide variety of technologies available allows us to cater to different range and data stream requirements. These benefits allow users to get the capabilities of the maximum of the available hardware.

Most of the papers out in the research domain are concerning the deployment of a wireless sensor network and how to improve transmission parameters in different environments. There are also a few which focus on the design model and securing the architecture to get optimum results.

The soil moisture measuring tool designed and developed in [7] allows the user to visualize moisture, temperature, and humidity at a user level. The custom hardware specified in the research paper proposes and implements the sensor node data collection based on the Arduino Mega Circuit and super long-range (LoRa) LPWAN network. This makes the overall system efficient and practical to the client. The weekly graphs shown has no proper standard metrics involved with not accuracy testing and was found that the soil moisture was a fixed value compared to the sensor readings from DHT-22.

In the sensor integration class, which focuses on what kind of sensors to be used and how they can be assimilated into the whole system as blocks. The production of custom solutions to propose a new network architecture is the most common model. TRIoT [8] is a classic example of multi-agent architecture. based on the Teleo-Reactive paradigm. They prove how remote IoT based nodes can work as Fog based computing to improve overall accuracy and sensor integration. A case study of precision agriculture was used to prove its application. Similarly, [9] also works on proving the application of the Teleo-Reactive paradigm. A clear cut discussion of TR based message payload and the translation process is explained. A robotic agent is used as a validation parameter. The paper [10] explains the overall concept of precision agriculture and proposes OPAIMS a two-tiered sensor network to collect and store data for remote applications.

The solution presented in [11] gives an overall architecture where multiple sensors such as environmental sensors, video feed, and GPS locations are fed into a monitoring system to get the results.The need for a highly efficient Wireless sensor network for precision agriculture and its current challenges is explained in [12]. The concept of LOS and obstructed view, transmission time and use of hybrid energy sources are explained in detail. The MAGPIE [13] shows how we can use a modified WSN to achieve the active monitoring of chronic diseases.

The application of an ant colony algorithm to improve issues raised from static sink nodes as mentioned in [14] allows us to find a shorter and optimal trajectory for message transmission. Coupling this with a robot for daily data collection will allow the data to be collected swiftly and easily. This is specifically applicable to banana plantations where the chain belt to transport the system can be used as a mobile sink node. Wireless sensor networks have issues with energy management. The paper [15] uses efficient short path data collection algorithm to reduce power consumption by the reduction in active hop transmission.

The paper [16] presents a systematic classification of systems and new CPS paradigms. A compilation of different viewpoints to applications at different levels of granularity. Such steps allow for a clear definition of cyber-physical systems with their specifications.

Water management is one of the primary concerns with any field of agronomy and is of special interest to research groups dealing with smart agriculture. There were many breakthroughs in this field over the years and the addition of the Internet of Things (IoT) has allowed us to get fascinating results. [17–19].We compare the existing literature with our proposed model in Table 2.1. These are some limited examples to showcase the use of WSN in other domains of interest. A few more examples are showcased with LoRaWan Technology [20–22].

Table 2.1 Comparison of proposed model with existing literature

Model	**Advantages**	**Disadvantages**
Smart Agriculture Using IoT Multi-Sensors [7]	- Remote data collection - Live data visualization	- No verification of collected data - System run under optimized use-case
Development of WSN [23]	- Reduce Cost -Increase Range of Transmission.	- Generates large amount of data. - Validation of data becomes difficult.
Smartphone Connectivity [17]	- Reduced real-time processing - Reduced battery life	- Generates noisy baseless data. - Limited Storage on units.
A life cycle framework of green IoT-based agriculture [18]	- Recognize and maintain ingredients quality - Improves yields.	- Requires supporting theory based on Financial ,operations and management issues in digitization of agriculture.
Proposed Model	- Reduced Cost - Plug and Play system - Runs for an entire harvest season - Feedback used to improve water management.	- Requires testing on other radio technologies to improve range. - uni-directional transmission. - node driven (no handshake).

2.2 Industry grade prototype review

The industry provides a good collection of solutions for collecting data from the field. They are generally referred to as Data Acquisition Systems(DAQ). It starts from a small minute data logger to super rugged backup on planes also referred to as a Black box. They are used in various roles from collecting data to making a backup of actions. This is a major requirement for the industry.

The industry for precision agriculture has two types of data loggers:

1. Active systems.

2. Passive systems.

2.2.1 Active Systems

There are more modern and intelligent systems who deal with data collection and data processing at the same time. They usually have a processing center and a method of transmitting the collected data to the internet. It could use a variety of technologies ranging from short-range low power Bluetooth to super long-range LoRaWAN networks of range 10+ miles.

The main characteristics of active systems are as follows:

- Collect data and process into information.
- Directly used by user.
- Wireless deployment.
- Ease of Use.
- Perform actuation.
- Cloud based Visualization tools.

Drawback:

- Expensive to set it up.
- Require assistance to maintain and upgrade.
- Limited no of sensor options (only those support actively by the manufacturer)
- Might require subscription to use the online tools.
- A premium solution.

To bring more clarity let us consider some of the more popular active systems in the industry.

2.2.1.1 Hobo Field Monitoring System

The HOBOnet Field Monitoring System [24] (Figure 2.2) from Onset provides scalable solution for web-enabled monitoring of field conditions for applications such as crop management, research, and greenhouse operations.

Onset's wireless sensors are ready to deploy and easily link to the network, and data is accessed through the new version of HOBOlink, Onset's innovative cloud-based software platform. Onset's cloud-based HOBOlink software makes it easy to view user data and manage HOBOnet Field

Figure 2.2 Onset Hobonet Field monitoring system

Monitoring System remotely. With new dashboards and Google Map integration, field monitoring data is more accessible and more meaningful.

Advantages:

a)Lowest cost-per-measurement point.

b) Wide coverage with wireless mesh technology that can communicate through vegetation.

c) Scalable, with up to 50 wireless sensors streaming back to a central, cloud-based weather station.

d) Remote access to data and current conditions with a customized dashboard for analysis.

e) Alarm notifications when conditions reach user-set thresholds.

Disadvantages:

a) Requires trained support staff.

b) High initial investment required.

c)Any additional sensor increases system cost exponentially.

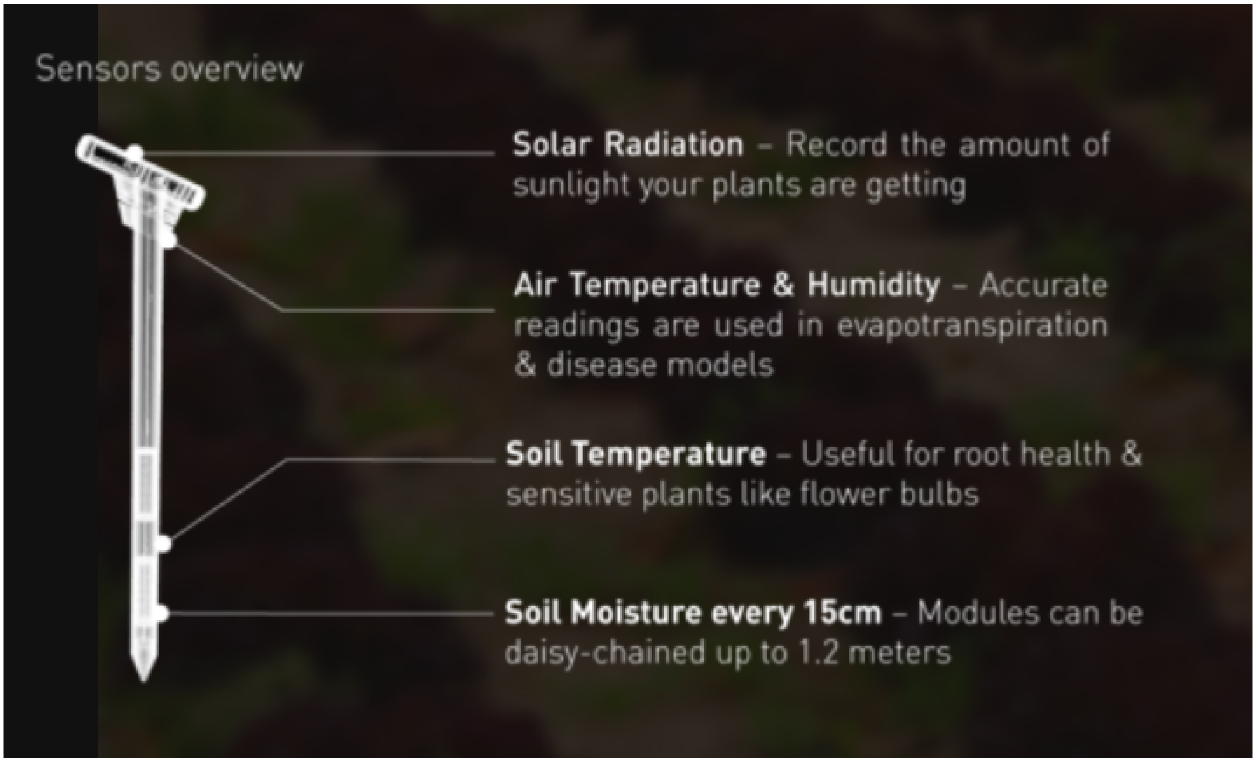


Figure 2.3 PYCNO Core - All in-one soil sensors

2.2.1.2 PYCNO

PYCNO [25] (Figure 2.3) is a cloud based plug and play senor array. It is self sufficient with solar based power charging circuit. It only has three components - Node sensor, master sensor, and cloud platform. The master node and sensor node both have 5 sensors on board such as solar radiation, air temperature & humidity, soil temperature and soil moisture (at 15cm depth interval). The master node collects all the data and pushes to the cloud using GSM network.

Advantages:

-Small all contained sensor units.

- Master node based mesh system.

- Online cloud platform is well developed.

- No additional sensors or set up is required.

Disadvantages:

- No modification to the system is possible.

- High initial investment required.

- Limited no of sensors spread over the entire field.

2.2.1.3 NCD Remote Soil moisture sensor

Figure 2.4 Wireless Soil Moisture Sensor Long Range IoT Transmitter

NCD IoT Sensors [26](Figure 2.4) send a lot more than just sensor data. For instance, a unique serial number is sent so you can always identify a particular sensor on the network. A Node ID is also included, which a single byte of data that you can use for anything, such as the floor of a building or to help identify which group a sensor belongs to. Also included in the data packet is a firmware version, battery level, and sensor type. This allows your software to positively identify and manage the health of NCD IoT wireless sensors. While signal strength data is also available, it is handled in a different way, and is not included in the sensor data packet (but rest assured, signal strength data for each sensor is available). Please see the Resources tab to see detailed information on the data structure of this device.

Features:

- Wireless Soil Moisture Sensor for Indoor Outdoor Use.
- Industrial Grade 1 Channel IoT Soil Moisture Sensor.
- 10-bit Resolution 2 LSB Absolute Accuracy.
- Usable ADC Values of 0 (dry) to 830 (wet).
- Auto ADC Sample Transmission on Moisture Level Change.
- Configurable 10% Analog Voltage Change Detection.
- 2 Mile Line-of-Sight Range with On-Board Antenna..

Typical Applications:

Green House Soil Moisture Monitoring Application, Wireless Plant Health Monitoring, Smart Irrigation System, Low Soil Moisture Change Alarm Application, Create the perfect conditions for compost piles, Protect concrete foundations by maintaining adequate soil moisture, Push Soil Moisture Level To Cloud Services Like AWS and Azure, Wireless Soil Moisture Sensor for Indoor Outdoor use.

2.2.2 Passive Systems

These are plain data acquisition systems also called as data-loggers. They are robust devices whose main role is to collect the data and store it locally. It performs no other tasks such as processing or transmitting. They are black box devices used to collect logs or sensor information.

Advantages:

- Robust small packages.
- Have large memory to store data in most optimized format.
- Long battery life as its highly optimized.

Disadvantages:

- Requires manual reading or measurement of data.
- Post processing of collected data is required.
- Price exponentially increases based on ruggedness and portability.

2.2.2.1 Decagon Em5b

Figure 2.5 Decagon - em5b Data-logger

The Em5b [27] (Figure 2.5) is a 5-channel, self-contained data logger. It is housed in a white weather-resistant enclosure, making it suitable for outdoor operation in moderate weather climates. It is powered by 4 AAA size alkaline batteries. Your Em5b is designed primarily to make soil moisture measurements. It is the most cost effective logger in the ECH2O System. Decagon created the Em5b for researchers on a tight budget who want to measure large numbers of soil moisture sensors. The Em5b is a component of the ECH2O System, which comprises data loggers, sensors, telemetry, and software that help you measure soil moisture and other environmental parameters accurately and cost effectively. Decagons innovative sensors are the heart of the ECH2O System. Decagon also designed the system to be very easy to use (no programming needed).The ECH2O System gives you two software options for working with your hardware and collected data. Choose one or more packages to fit your needs.

Features:

- Controllable Measurement interval.
- Configurable to match sensors used.
- Internal storage with non-volatile flash up to 3,348 scans.
- No sensor detected will write port to 0.

2.2.2.2 Delta-T DL6 Soil Moisture Sensor

Figure 2.6 Delta T DL6 Soil Moisture Data Logger

The DL6 [28] (Figure 2.6) is a dedicated soil moisture logger optimized for use with soil moisture sensors. It can be used with combinations of Theta Probes and Profile Probes and also accepts rain gauge and soil temperature probe inputs. It is well suited to both research applications and irrigation monitoring.

DL6 Loggers include a novel accelerated logging feature to allow the tracking of wetting fronts. As you would expect from Dynamax, the DL6 is tough and passes a 1m drop test. It is environmentally sealed (IP68) and can be used in the field without any further protection. To minimize the need for opening the case, data is collected via a weatherproof external RS-232 socket, and the status of the logger can be checked using a vibration-activated LED.

Features:

- Ideal for Profile Probes, ThetaProbes and SM150T sensors.
- 8 channel logger with weather-proof case and battery power.
- Free use of Delta-LINK-Cloud data sharing service.

2.2.3 Comparison

We perform a comparison on the available systems and come to a decision on the hardware requirements based on what the industry fails to provide the farmers. This extensive comparison can be seen in Table 2.2.

This helps us correlate the issues with what the literature available and the solutions a user has available in the market. This further solidifies the problem statement for this thesis.

Table 2.2 Comparison table with industry standards and proposed hardware

Parameter	**HOBOnet**	**PYNCO**	**NCD Sensor**	**em5b**	**DL6**	**SCR-DAQ**
Cost for entire system	$1500+	$2649	$219.95 + base station	$300	>$200 (quote)	less than $200 (Need)
No of sensors per node	Base station up to 10. nodes 1-4	5 basic. Soil moisture sensor can vary.	1	4	8	4x
Connection to Cloud	Wi-Fi or cellular	Cellular	ZigBee Mesh	None	None	Depends on applied Technology
Power supply	Battery + Solar	Solar	Battery	AA	AA	??
Ease of install	Require installation	Plug and play	Plug and play	Plug and play	Plug and play	Plug and Play
Data Visualization	HOBOlink proprietary cloud	Centralized Cloud Console	User preference	User defined CSV	??	Centralized cloud Console - proposed
Sensor options	Varied and additional hardware	Fixed per module type.	3-4 options with ruggedized features	Commercial sensor from Decagon	Soil moisture sensors sold by manufacturer.	Any analog sensors added to calibration.

CHAPTER 3. PROPOSED ARCHITECTURE OF CSR-DAQ

IoT has been a trend for recent years and industry has been releasing a lot of smart devices ranging from smart light bulbs to smart vehicles. The whole scenario is based on the concept that the devices can use the collected sensor data clubbed with user input over the internet. This allows for automation or other forms of control of actuators.

The whole architecture is based on the general idea of the Internet of Things (IoT) where devices collect data, process it or transmit to a central control/ collector which will decide on the actions to be taken by the connected devices. This control can be over the internet wired control or any wireless technology. There has been growth in several industrial segments. Smart agriculture or Smart farming is one of the fastest areas of growth due to the rising need to produce by the growing population and application of AI and big data. This has resulted in a niche in the market with expensive senor and data logging systems. These are generally not affordable by farmers. This calls for the need for a cheaper and practical solution to farmers. Hence this architecture was proposed and implemented.

3.1 Design Goals

This section will provide a brief overview of the high-level considerations that were given before arriving at the proposed solution. Design goals such as the ones considered form the salient features of the proposed solution.

- **Cost-effectiveness:** The modules designed should be able to fit into the budget of the end-user.
- **Availability:** The immediate replacement parts should be available locally. Ex. AA batteries.

- **Accessibility and deployment readiness:** The solution should be easy to install without much assistance from expert technicians.
- **Scalability:** The solutions should be able to support multiple users, location, area, sensor types and so on. Any additional information should not cause any issues in the future.
- **Power efficiency and performance:**The solution should provide maximum field time without the need for the user to continuously replace the power supply.
- **Security:** The data transmitted should be secure and should be as stable and reliable as possible.
- **Visibility:** The collected data by the solution should be processed and be helpful to the end-users to make decisions.

3.2 Test bed Overall Architecture

Figure 3.1 tells the entire architecture of CSR-DAQ. There are four main modules.

1. **Node Array** - A physical system to measure the environmental conditions and translate them into digital format. It is designed to read the sensor data with the help of an analog to digital converter. Then find the value to a reference voltage. Pack the collected information into a packet and send to the aggregator.
2. **Aggregator** - A system to collect the data wirelessly from the sensor node array, store the information locally and also upload it to the cloud. The custom software is designed to read the required data into the system via a serial module and request the required information.
3. **Cloud DB** - Subscribe the MQTT channel and store data to access it in the future. The cloud storage allows multiple back-ups provided by a third person software solution to make sure that the database will available always. The data access allows for continuous read and writes with overwrite protection.

4. **Client Application** - Customized applications to read the latest data and provide insights based on the collected data. The data can be used in the required format as required. Graphs of different formats can be used to get required interpreted data.

Table 3.1 Modules of proposed CSR-DAQ

Module	**Description**
Node	• Physical device which collects data from sensor • Transmits data to the aggregator
Aggregator/ Base Station	• Collects the transmitted messages from the nodes • Strip data into information • Publishes to the internet.
Cloud based r DAQ Serve	• MQTT Channel for publishing and subscribing • Stores subscribed data in a Cloud based Relational Database
Client Application (customized App)	• Access cloud data to provide visualization • Subscribe and store data in a local system • Predict and react to information collected to perform automation.

The CSR-DAQ is further clarified in the Table 3.1. A more clear explanation of the same will be mentioned in the following sections.

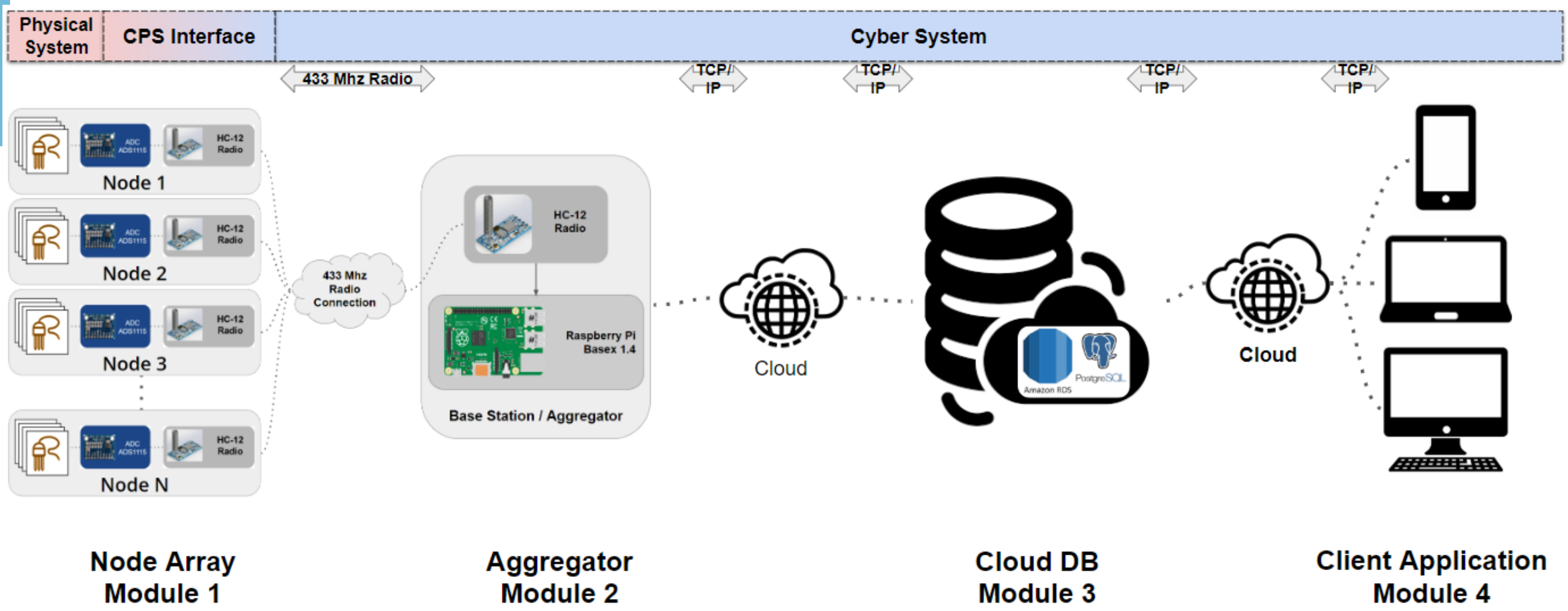


Figure 3.1 End-to-End Architecture of the CSR-DAQ

3.3 Node

The core portion of the CSR-DAQ design was the creation of the project. It was the area the design was focused on. The design was made to get maximum efficiency of transfer of collected data to the base station with the best technology possible.

The node was designed to perform the following functions:

- **Wake up** at set time interval based on user requirements. With options varying from 1 min to 60 minutes.

- **Collect** data from measured analog sensor values and transmit to the base station or aggregator.

- Effectively and wirelessly **transmit** the collected data via the implemented technology.

- Power down all the components on-board to **increase the battery life** to sustain for more than a harvest season.

These functions are severely restricted by the hardware requirements and environmental conditions. Due to extreme conditions such as high humidity, varying temperatures, and rough handling. This leads to a lot of pf practical issues which after theoretical calculations need to be considered. The variability of the battery consumption and capacity due to temperature and other factors also needs to be accounted to this model.

On consideration with all these parameters, we can bring about the following design requirements:

1. Battery Life-time.
2. Range of Transmission.
3. Resolution / Consistency.
4. Ruggedness.

3.3.1 Hardware Design

This section deals with an explanation of the considerations and issues put forward into the design of the hardware. A lot of consideration has been put into designing the hardware from conceptual design to version V1.5. Based on this the design was split into five separate modules:

1. Power Supply

2. Analog sensor input.
3. Analog-Digital Converter.
4. μcontroller and RTC.
5. Radio Module.

This has been effectively shown in figure 3.2. A deeper consideration and explanation are given in the following sections.

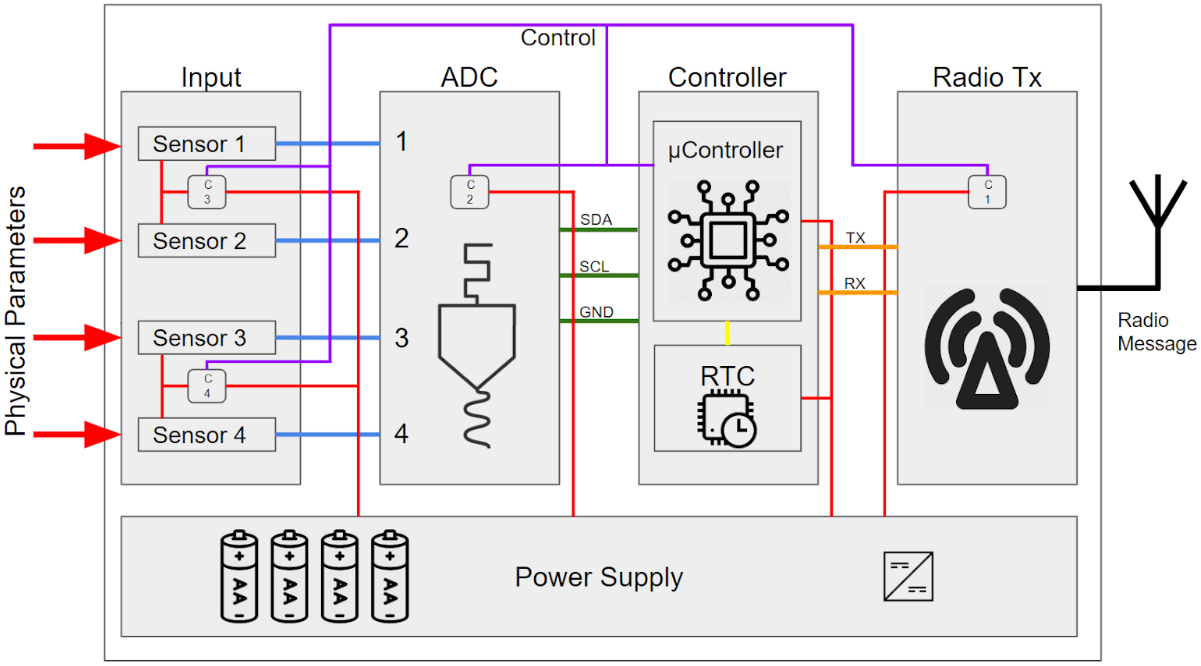


Figure 3.2 Overall Node Architecture

3.3.1.1 Power Supply

The main aspect of being an IoT device is its ability to stay alive for a long time frame without any additional change or power supply. This is a crucial requirement in the field. The specific conditions are:

- The device should be standalone. Powered by battery or solar power.

- Replacement battery should be easily available.
- The battery power conversions unit should be as efficient as possible with least voltage drop variation.
- Switch unwanted devices off when not required.

The power supply circuit had to designed in such a way it supplies the required theoretical requirement of 300 mA. The source should be based on the following constraints:

1. Good Capacity vs Size and weight.
2. Cost vs Availability.
3. μcontroller Voltage Requirements.

Based on the three constraints with the end solution to be designed for Farmers or in-field application. Priority was given to the easy availability of the batteries than specialized high capacity and bulky batteries. Also, the voltage in series should be greater than 5.5V which is satisfied by the use of the **4AA batteries**.

$$4 * 1.5V(AA) = 6V(theoretical) \approx 7.2V(practical) \tag{3.1}$$

This is then used to provide 5V using the buck circuit implemented using the AP1509SG-13 (Diodes Inc.) with a 2A current rating. and input voltage rating between 4.5V - 22V and fixed output at ≃5V.

3.3.1.2 Analog Sensor Input

The 3.5 mm jack port is the standard for the analog sensors, which is an industry-standard. This module is shown in Figure 3.4. This was used in the design with the option for a multiple of 4 sensors on each board. This is the interface between the physical values and the digital world. A proper equation to translate the given ADC values needs to be applied as taught in [29].

3.3.1.3 Analog Digital Converter

There was a need to increase the resolution of the read analog readings. The commercial data-logger is running a 12-bit resolution. This was an issue in the earlier prototype where the Arduino

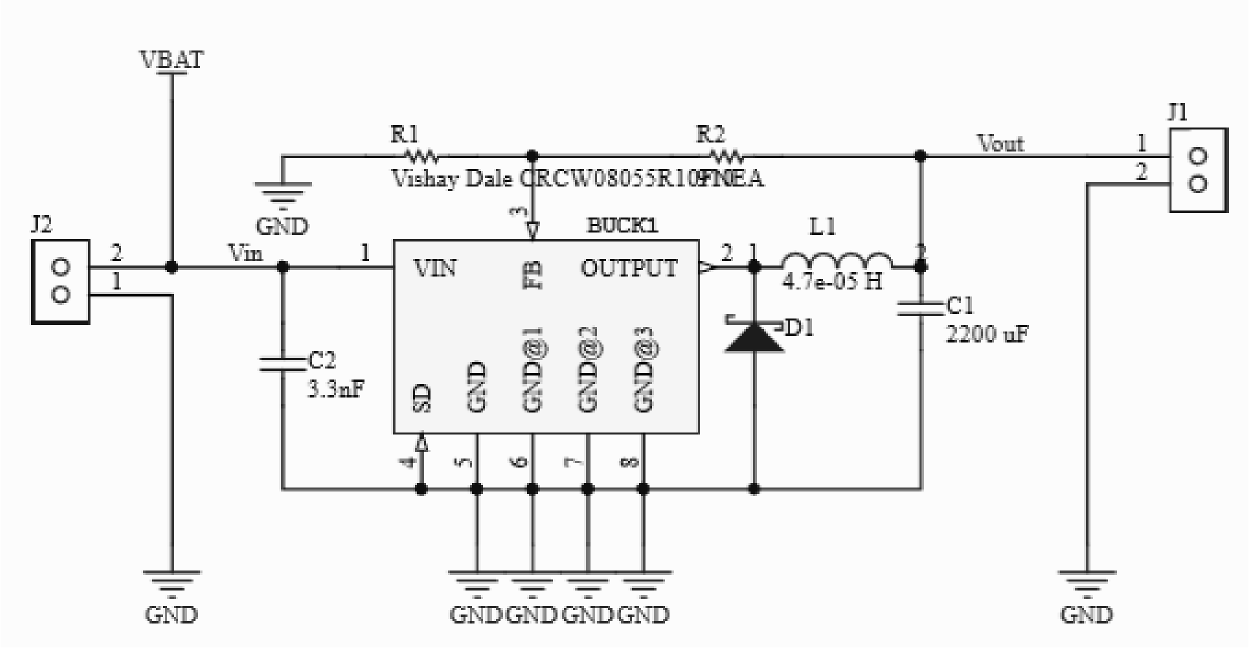


Figure 3.3 Buck Circuit Diagram

inbuilt ADC was only 10-bit. This reduced the accuracy at which we get the results from the data logger. This resulted in the addition of the ADS1115 (Figure 3.5) 16-bit i^{2}c device which has high accurate 4 input options. This also makes sure that even with the slight variations in the power voltage we will still get the required accurate values.

$$\frac{Resolution \quad of \quad the \quad ADC}{System \quad Voltage} = \frac{ADC \quad Reading}{Analog \quad Voltage \quad Measured} \tag{3.2}$$

3.3.1.4 μcontroller and RTC

The ATEMEGA328P (Figure 3.6) is the main controller in this design. It was based on a stable design called the Arduino UNO proto-typing board to run the required hardware. ATMEGA 328 has 1KB Electrically Erasable Programmable Read-Only Memory (EEPROM). This property shows if the electric supply supplied to the micro-controller is removed, even then it can store the data and can provide results after providing it with the electric supply. Moreover, ATMEGA-

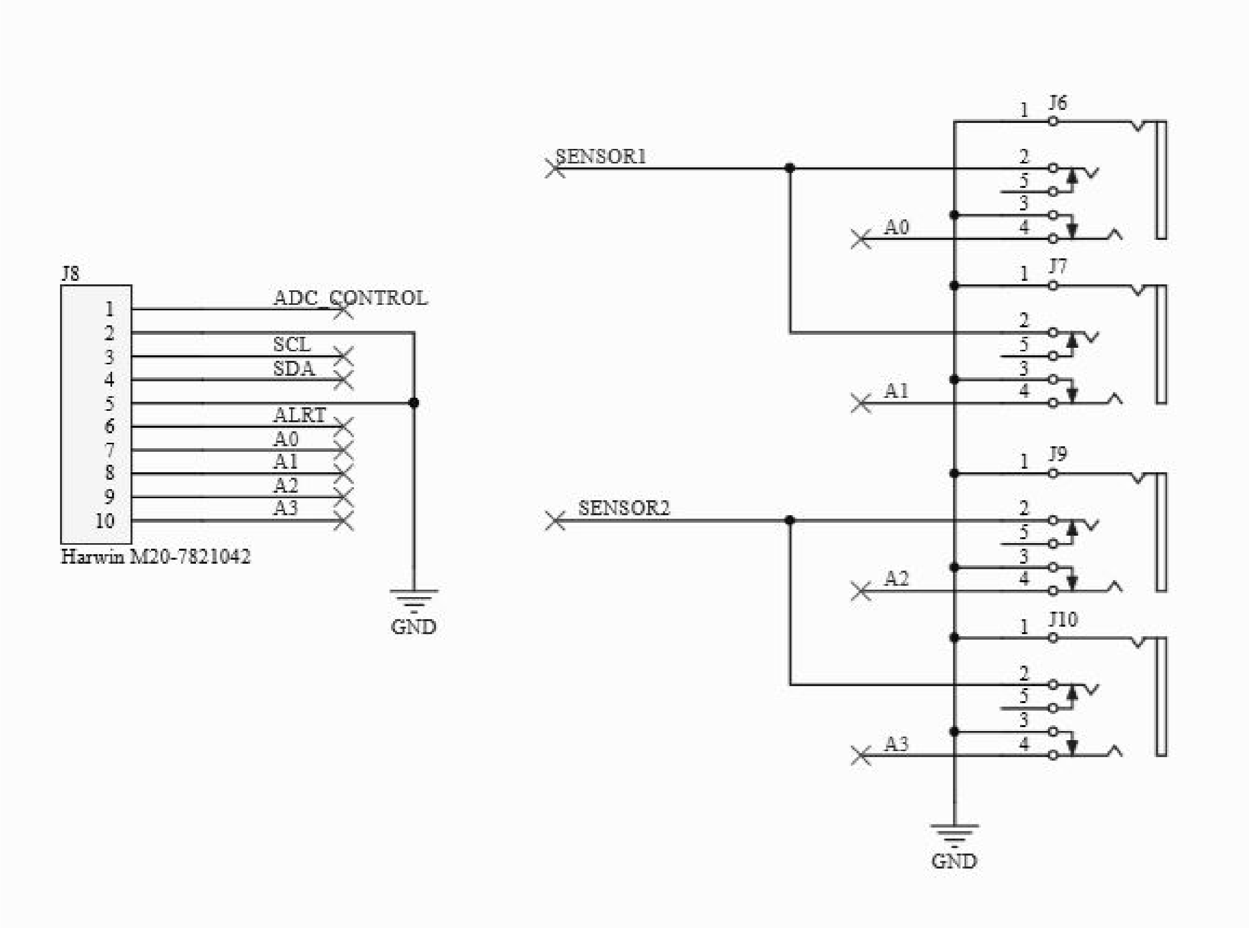


Figure 3.4 3.5 mm analog jack design and ADC socket

328 has 2KB Static Random Access Memory (SRAM). Other characteristics will be explained later. ATMEGA 328 has several different features which make it the most popular device in todays market. These features consist of advanced RISC architecture, good performance, low power consumption, real timer counter having a separate oscillator, 6 PWM pins, programmable Serial USART, programming lock for software security, throughput up to 20 MIPS, etc. ATMEGA-328 is mostly used in Arduino. The further details about ATMEGA 328 will be given later in this section.

The following features were crucial on designing the hardware:

- Low power sleep mode.

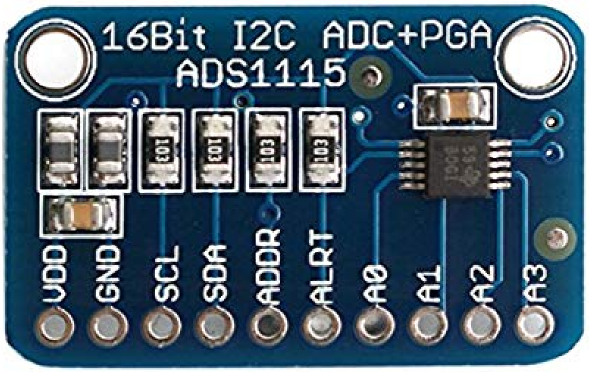


Figure 3.5 ADS1115 4 input ADC unit

- Accurate control and extensive small device communication such as i^2c and SPI.
- Easy to interface with varying technologies.
- Run at a fixed clock frequency and 5V power rail to easily merge with the sensors in the field.

Figure 3.6 shows the pinouts on the microcontroller and the no of available pins for use. The device in full sleep mode requires an external interrupt to wake it up. The external interrupt is a power supply signal with the wake condition being a Rising Edge of the signal. This is shown in Figure 3.7. Here the external interrupt is provided by the RTC clock running on the unit.

The current RTC we are using is the DS3231 RTC Precision Clock with a skew of 1 sec in a year. This is taken as an external system module which allows us to update the clock to the required time easily.

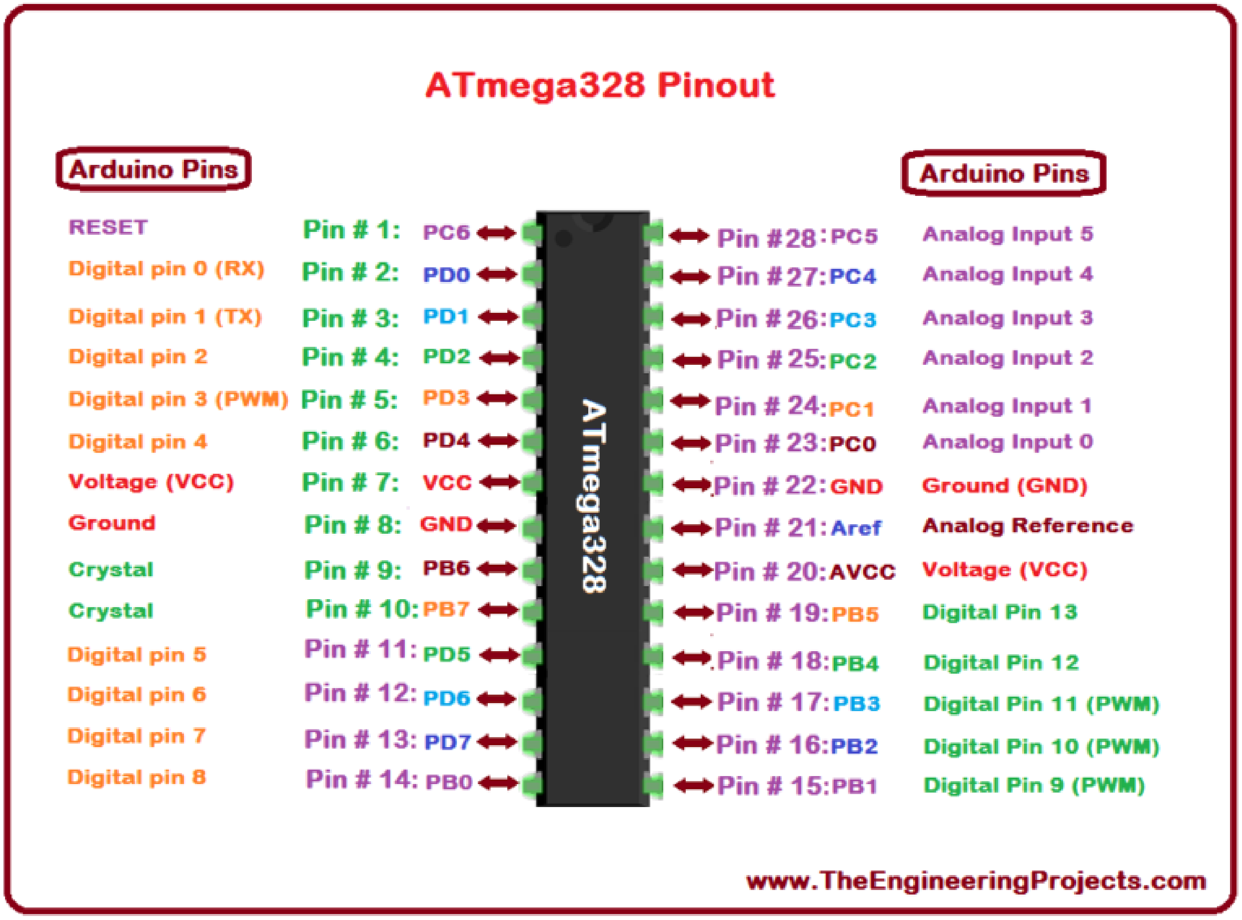


Figure 3.6 ATMEGA328P Pinout [1]

Features:

- Can be connected directly to the microcontroller IO ports.
- Standard 2.54 mm pins for input and output connections.
- Two calendars and alarm clock.
- Two programmable square-wave outputs.
- Real-time clock generator for seconds, minutes, hours, day, date, month, and year timing.
- Valid until 2100 with leap year compensation.
- Can be cascaded with other I2C devices.
- The address can be set using the pins A0/A1/A2 (the default address is 0x57).

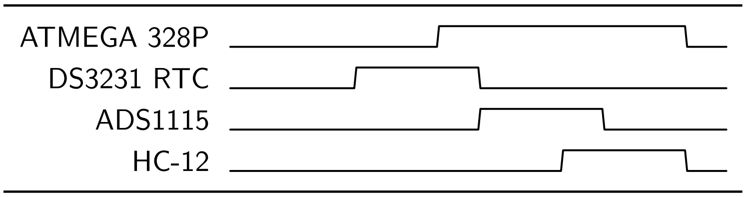


Figure 3.7 Node Wake up Cycle

- Battery socket compatible with LIR2032 batteries.
- I2C interface.

Specifications:

- Operating voltage: 3.3 V to 5.5 V.
- Real-time clock chip: DS3231.
- Clock accuracy: 2 ppm.
- Memory chip: AT24C32 (32 Kb storage capacity).
- On-chip temperature sensor with an accuracy of 3 .
- I2C bus interface maximum speed: 400 kHz.
- Size: 38 x 22 x 14 mm.

3.3.1.5 Radio Module

The design has to be in such a way it can be interfaced with multiple technologies. As per the client's needs and requirements. The radio will be working on a node identifier and will transmit the data to the aggregator or the base station.

3.4 Aggregator / Base station

The main function of the aggregator is to collect the radio messages actively and record the data transmitted serially. The base station began with a raspberry pi zero with Arduino connected to it to act as an intermediate to read serial messages from the radio. This has been replaced with Raspberry pi 3B for the need of better processing power and RJ45 Ethernet connector.

Table 3.2 Available Compatible Technology

Radio Technology	**Payload**	**Range**
Long-Range (LoRa)	0.3 kbps- 50kbps	upto 15km
Sigfox	approx. 100 bytes	30-50km
Ingenu	proprietary	approx. 30mile
Satellite	?	upto 36,000 kilometres above Earth
ZigBee	20 kbit/s (868 MHz band)-250 kbits/s (2.4 GHz band)	10-100 m
Z-wave	9600 bits/s - 40 kbits/s	100m
Wi-Fi	upto 10 GB/s	-
Bluetooth	up to 24 Mbits/s	up to 25m
RFID	-	up to 5m

In the later generation, an auto serial port detection OS detection script has been written to work with multiple Operating systems. A sample screenshot of the Software is also attached. The function starts with creating a .csv file with the pre-recorded settings and waits for serial data to come in. This allows keeping the local log of all collected data. This allows making sure no data is lost. The network upload using the MQTT allows for secure transmission to the digital channel. The subscribers can use the data from the channel for the required solutions. The base station software is customized to run at the required specification for each use case scenario.

3.4.1 Firmware

The Node has BasexV1.3 Transmission was written in Arduino IDE. It controls the wake-sleep and other peripherals in the PCB design. This uses a low-power package to put the Microcontroller to Full sleep. This makes sure that the micro-controller is asleep until an external event wakes it up.

The radio network is a unicast broadcast model. Data can be lost if two nodes transmit at the same time. This issue was solved using a simple collision avoidance algorithm (Algorithm 1).

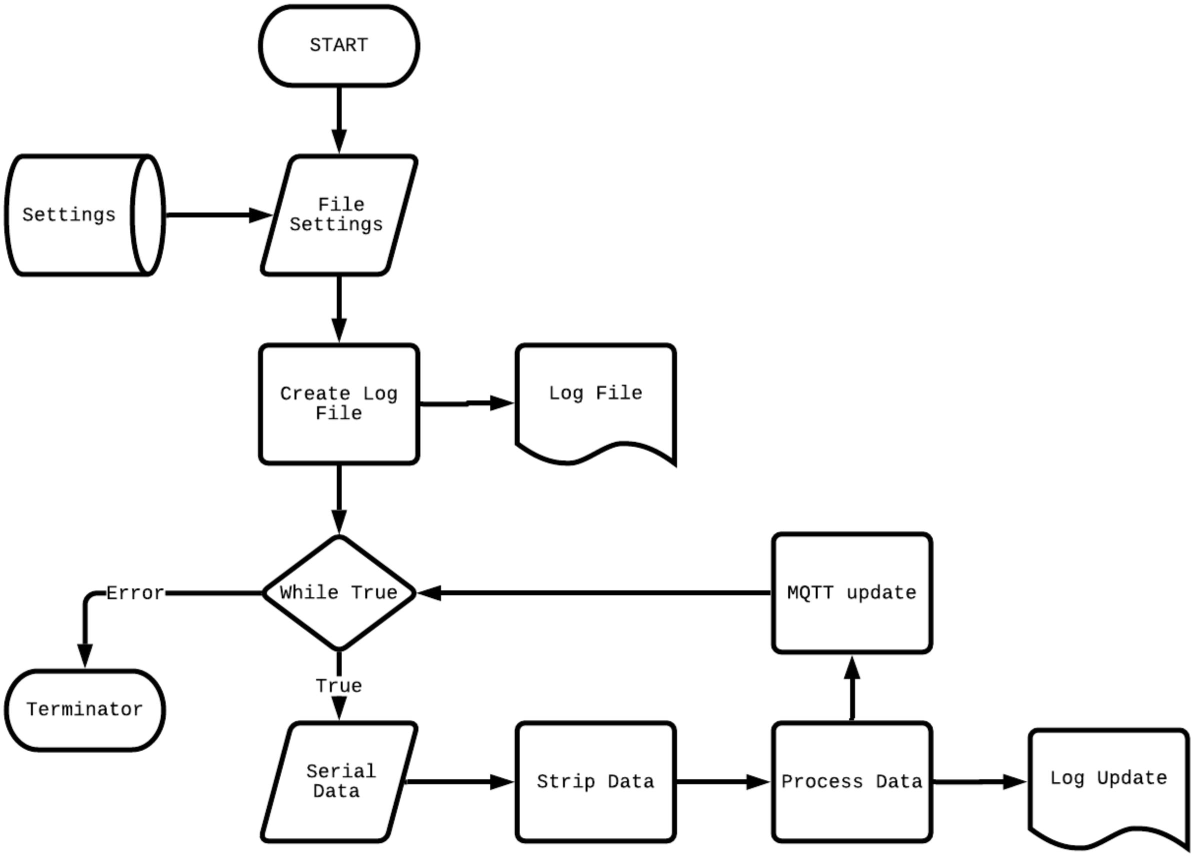


Figure 3.8 Basex function architecture

3.5 Cloud Database

The entire database is structured so that it is in a cloud-based service. This allows us to retrieve data from anywhere in the world as long as the proper security settings are configured. The hardware i.e the base station has to be configured as an IoT Core to enable this feature.

3.5.1 MQTT

MQTT (Message Query Telemetry Transport) [30] is an open OASIS and ISO standard (ISO/IEC PRF 20922) lightweight, a publish-subscribe network protocol that transports messages between devices. The protocol usually runs over TCP/IP; however, any network protocol that provides

```
Data: DateTime from RTC
Result: When to Transmit Data
initialization;
Node id (unique to each node in network);
Transmit time = 5 ;
wake up minute = ((Node - 1)* Transmit Time) / 60 ;
wake up second = ((Node - 1)* Transmit Time) % 60 ;
while True do
    read Date-Time;
    if Minute = wake up minute then
        if Second = wake up second then
            Collect sensor readings;
            Transmit Readings;
            Go to sleep;
        else
            wait for wake up ;
        end
    else
        go to sleep;
    end
end
```

Algorithm 1: Basex anti-collision Algorithm

ordered, lossless, bi-directional connections can support MQTT. It is designed for connections with remote locations where a ”small code footprint” is required or the network bandwidth is limited.

The MQTT protocol defines two types of network entities: a message broker and a number of clients. An MQTT broker is a server that receives all messages from the clients and then routes the messages to the appropriate destination clients.[11] An MQTT client is any device (from a microcontroller up to a full-fledged server) that runs an MQTT library and connects to an MQTT broker over a network.[12]

Information is organized in a hierarchy of topics. When a publisher has a new item of data to distribute, it sends a control message with the data to the connected broker. The broker then distributes the information to any clients that have subscribed to that topic. The publisher does not need to have any data on the number of locations of subscribers, and subscribers, in turn, do not have to be configured with any data about the publishers.

If a broker receives a topic for which there are no current subscribers, it will discard the topic unless the publisher indicates that the topic is to be retained. This allows new subscribers to a topic to receive the most current value rather than waiting for the next update from a publisher.

When a publishing client first connects to the broker, it can set up a default message to be sent to subscribers if the broker detects that the publishing client has unexpectedly disconnected from the broker.

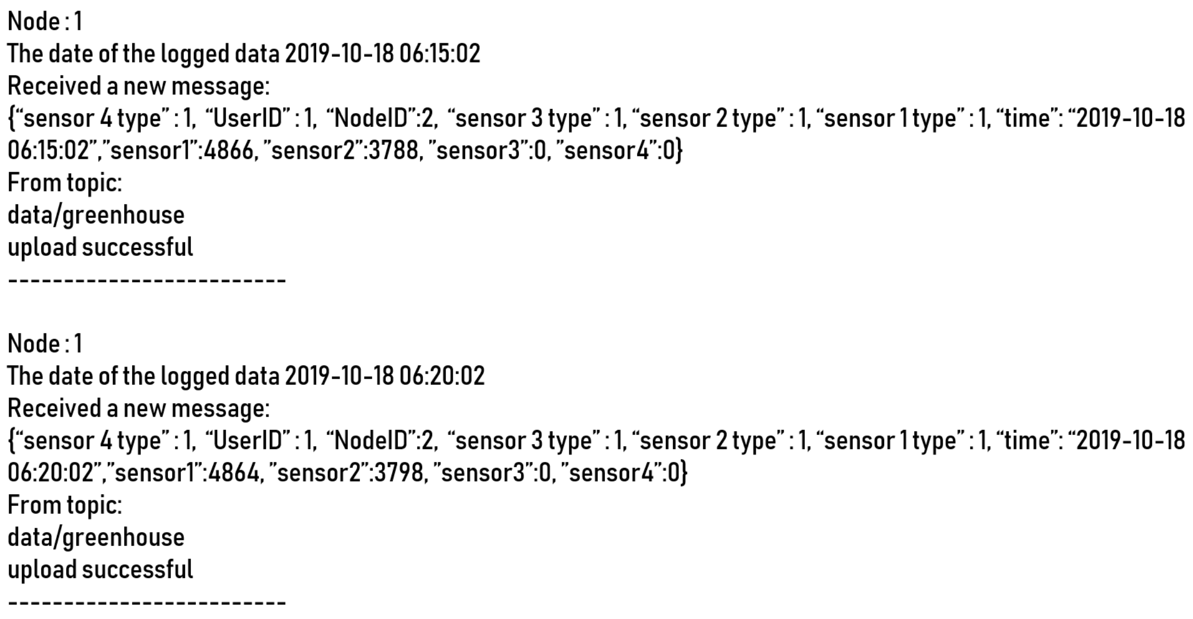
Clients only interact with a broker, but a system may contain several broker servers that exchange data based on their current subscribers' topics.

A minimal MQTT control message can be as little as two bytes of data. A control message can carry nearly 256 megabytes of data if needed. There are fourteen defined message types used to connect and disconnect a client from a broker, to publish data, to acknowledge receipt of data, and to supervise the connection between client and server.

MQTT relies on the TCP protocol for data transmission. A variant, MQTT-SN, is used over other transports such as UDP or Bluetooth. MQTT sends connection credentials in plain text format and does not include any measures for security or authentication. This can be provided by the underlying TCP transport using measures to protect the integrity of transferred information from interception or duplication.

3.5.2 Amazon AWS RDS

Amazon Relational Database Service (or Amazon RDS) is a distributed relational database service by Amazon Web Services (AWS). It is a web service running "in the cloud" designed to simplify the setup, operation, and scaling of a relational database for use in applications. Administration processes like patching the database software, backing up databases and enabling point-in-time recovery are managed automatically. Scaling storage and compute resources can be performed by a single API call as AWS does not offer an ssh connection to RDS instances.



```
Node : 1
The date of the logged data 2019-10-18 06:15:02
Received a new message:
{“sensor 4 type” : 1, “UserID” : 1, “NodeID”:2, “sensor 3 type” : 1,“sensor 2 type” : 1,“sensor 1 type” : 1,“time”: “2019-10-18
06:15:02”,”sensor1”:4866, ”sensor2”:3788, ”sensor3”:0, ”sensor4”:0}
From topic:
data/greenhouse
upload successful
-------------------------

Node : 1
The date of the logged data 2019-10-18 06:20:02
Received a new message:
{“sensor 4 type” : 1, “UserID” : 1, “NodeID”:2, “sensor 3 type” : 1,“sensor 2 type” : 1,“sensor 1 type” : 1,“time”: “2019-10-18
06:20:02”,”sensor1”:4864, ”sensor2”:3798, ”sensor3”:0, ”sensor4”:0}
From topic:
data/greenhouse
upload successful
-------------------------
```

Figure 3.9 MQTT payload format

3.5.3 Client Application

The client application is used as an extension of the live data collected from the channel. The server-client connection to the cloud database allows to retrieve older data collected on the server and run required test and solutions custom made for the clients. These could include predictions, analysis and visualization of the collected data or information.

CHAPTER 4. TESTING AND EVALUATION

This chapter focuses on the hardware parameters used to test the functionality of the testbed. It also has an explanation of the field test to verify the veracity of the compiled data.All theoretical solutions discussed in chapter 3 was implemented. An end-to-end test case is explored and then later on the security and authenticity of the collected data are verified

The deployment was done in two places as requested by the client Dr. Ajay Nair, Professor, Department of Horticulture:

1. Horticulture Hall Green House- Iowa state University - Test-bed
2. Iowa State University Horticulture Research Station - Field

4.1 Implemented Hardware

The following section explains the hardware used to implement the proposed architecture. A limited range of radio solutions was considered for implementation. The decision was based on the following conditions:

- Field range: 200m
- Plants under study: Vegetable patches.
- Temperature range : $-2\,°C$ - $44\,°C$.
- Sensors used: Decagon 10HS [31]

Based on available condition and resources the following was decided as test case scenarios.

- **Node:**
  - Microcontroller :ATMEGA328P [32].
  - Architecture : Arduino UNO.
  - RTC : DS3231 [33].

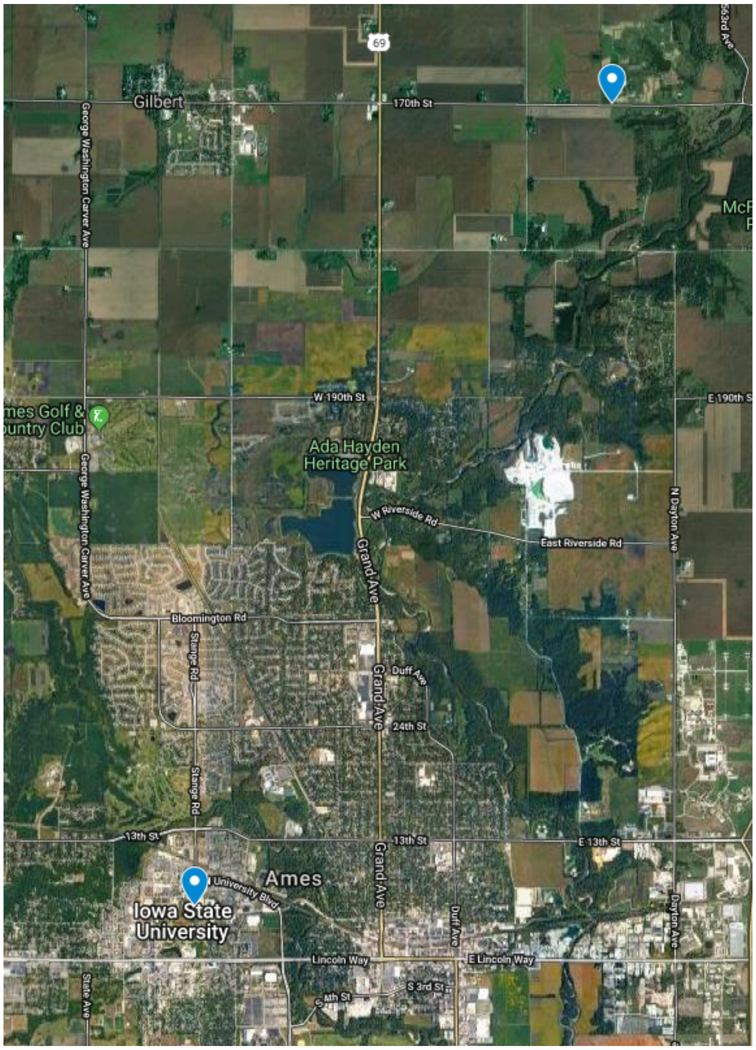


Figure 4.1 Test-bed and Field Location

  - ADC : ADS1115 [34].
  - Communication: HC-12 433Mhz Radio [35].

- **Base Station:**
  - SoC : Raspberry Pi 3B. [36].
  - Communication: HC-12 433MHz Radio.
  - Internet : Ethernet / Wi-Fi.
  - Firmware : Python based Basex.

- **Cloud DB:**
  - Cloud Service Provider: Amazon AWS [37].
  - SQL format : PostGres [38].
  - MQTT Platform : AWS IoT-Core.

These are used in the extensive testing and calculations.

4.2 Testing

The implementation was done in four phases to test and validate the performance of the collected data. The simple split of the phases are as follows:

Table 4.1 Testing Phases with Location and Duration

Phase	Location	Duration
1.Hardware and range Test	Horticulture Research Farm	Summer 2018
2.Greenhouse Data Collection	Horticulture Hall	January 2019 - September 2019
3.Outdoor Field Test	Horticulture Research Farm	September 31 - November 01 2019
4.Greenhouse Regression Test	Horticulture Hall	November 19 - 31 2019

4.2.1 Hardware and range test

The main goal of this test was to get perform trial and error on the hardware prototype V2.0 and analyze all the issues and defects. This test was performed in the Horticulture Research Farm in Story county.

The proposed hardware design was engineered to the specification and was deployed in a controlled area i.e Greenhouse. The plant understudy was ficus (Ficus Benjamina). This has good water absorption and allows us to replicate a soil saturation and dry setting in a span of a few days. Testing was done in a small pot to limit the area of coverage and test. Test conditions:

- Controlled soil and air temperature.
- Water added at recorded intervals
- within accessible range of the radio modules.

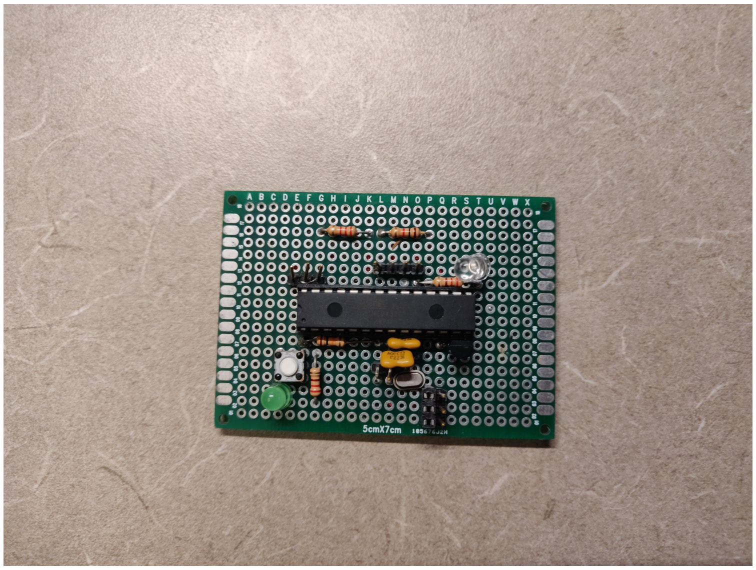


Figure 4.2 Phase 1 Prototype

- Limited access to external elements due to naked radio modules.

The following issue was observed and the readings with a single sensor module gave skewed readings

- Clock time was skewed.
- Battery was unable to run the device more than a few days.
- Used external voltage reference resulting in variation in recorded values.

The following were the readings observed over a period of a few days. A single sensor was used and no commercial sensors were used as a comparison at this stage where a direct 1-to-1 comparison is impossible.

From this graph, a smoothed graph can be made of the required readings but the interval of measurement is too small to make a difference in soil water content. The code written was to support the measurement at every 2 minutes but this was found to be a futile effort as the internal clock of the Arduino is skewed and needs additional processing power and controls to get the right time and additionally no delay or interrupt can be used while performing the same to maintain the quality and performance.

Additional insights:

- HC-12 is easily obstructed by thick items such as trees planted to break wind ex coniferous trees.
- Arduino is not a suitable solution for clock control and analog sensor readings.
- The Battery converter used is highly inefficient even though the voltage was maintained constant. A lot of power is wasted in the form of heat.

4.2.2 Greenhouse Data collection

The location of this test was in the open greenhouse at Horticulture hall at Iowa state university campus. The goals for this experiment are as follows:

- Collect actual sensor data.
- Prove a working prototype.
- Commercial Sensor graphing.
- Range Test.

This was a long-time test and improvement cycle with new hardware introduced and modified to get the required results. A few separate tests were conducted during this time frame. They include the following:

- Range Test:

In this test, a working battery-powered prototype was carried by an intern while the base station was placed outside running on the power bank. This allowed the device to see at what range the hardware was able to see the transmitted message without any loss. This test proved that the radio on Line-of-sight(LOS) gave a range of 0.5 miles (approx. 2500 ft) in crowded structural areas. Also, the base station was in an always-on state so it can receive data as soon as the message from the node was transmitted.

- Continuous message transmission:

The node was stress-tested to see if a continuous stream of data at regular intervals can be transmitted without collision with the base station. This makes sure multiple nodes can simultaneously

Figure 4.3 Test Bed set at the greenhouse

transmit data and no data is lost.This makes use of the anti-collision algorithm (Algorithm 1).

- Multi-sensor data collection:

This test is to make sure that the collected data is the same as the commercial data logger. This was proved to be an exact value as the sensor gave the same reading irrespective of the excitation voltage.

4.2.3 Outdoor Field Test

The Final prototype has all issues rectified and was designed to be placed in an IP66 outdoor accepted hardware case. This keeps all the hardware to safe from environmental factors. The PCB has been redesigned for easy re-programming using micro USB connection and battery facing the top so farmers can change the battery easily.

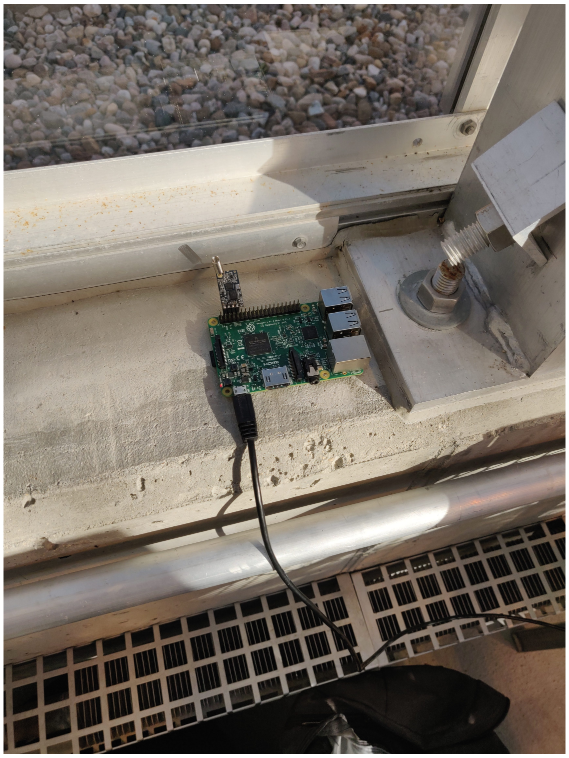


Figure 4.4 Base station Deployed

Advantages:

- Enclosed outdoor casing.
- Customized hardware.
- USB debugging.
- Wake-up test button.

This was deployed at the horticulture research station with three working sensors. This was deployed on two trees next to the building to see the value of the sensors rising and falling. This was also verified with the Equation for VWC and was run on battery power with the prototype 4.

Figure 4.5 Phase 3 Field Site with deployed sensors

4.2.4 Greenhouse Regression Test

This test is run on the final prototype built towards the project. This has the upgraded hardware with optimized firmware for maximum battery efficiency. The goals of this test set-up are as follows:

- Measure power consumption.
- Regression on collected data.
- Prove ease of deployment.

This is a week old stress test on the collected data to make the analysis of the results collected and see if the VWC measurements defined are close to the expected values. Figure 4.6 shows the base station placed in the testbed location. It is kept in a fully enclosed case to protect from the elements.

Figure 4.6 Phase 4 base station

The Figure 4.7 shows the Proposed node and the commercial node placed right next to each other. This is arranged in such a way that there are total of 8 sensors connected to each other in alternating format.

The whole project was set up to confirm the accuracy and the resolution of the new hardware designed and deployed. The fifth version of the prototype (v6) was deployed on 11-26-2019 13:50 pm at the Horticulture Hall Research Green House. A week-long stress test set with 8 Decagon 10HS sensors was deployed. Two plants were taken into consideration:

1. Plant 1 - ficus (Ficus Benjamina) - 2 sensors each(4 total).
2. Money Plant (Epipremnum aureum) - 2 sensors each (4 total).

Figure 4.7 Phase 4 Node

2 of each sensor were connected to the Proposed Datalogger and Commercial Datalogger. This was done to make sure that no data was lost due to bad sensors. And also helps in improving the regression values. Due to the hardware design fault, the components onboard was always in ON state. An exception is for the microcontroller who is always in low power full sleep mode and RTC running on any available power supply. The Figure 4.8 shows the complete setup. The plants have been properly labeled for clear reference.

The Reading was consistent with what was expected. The ADC on both the commercial and proposed sensors read the same on multiplying by their factors of 0.75 and 0.1875. The values were found to be having a R^2 value of 96% to 98%. This proves that the designed hardware is working as designed.

$$VWC = 2.97 * 10^{-9} mV^3 - 7.37 * 10^{-6} mV^2 + 6.69 * 10^{-3} * mV - 1.92 \tag{4.1}$$

The equation 4.1 on testing was found to give the most closest data to VWC as per the commercial data logger. A simplified version (Equation 4.2)was also found based on the Data Sheet of the

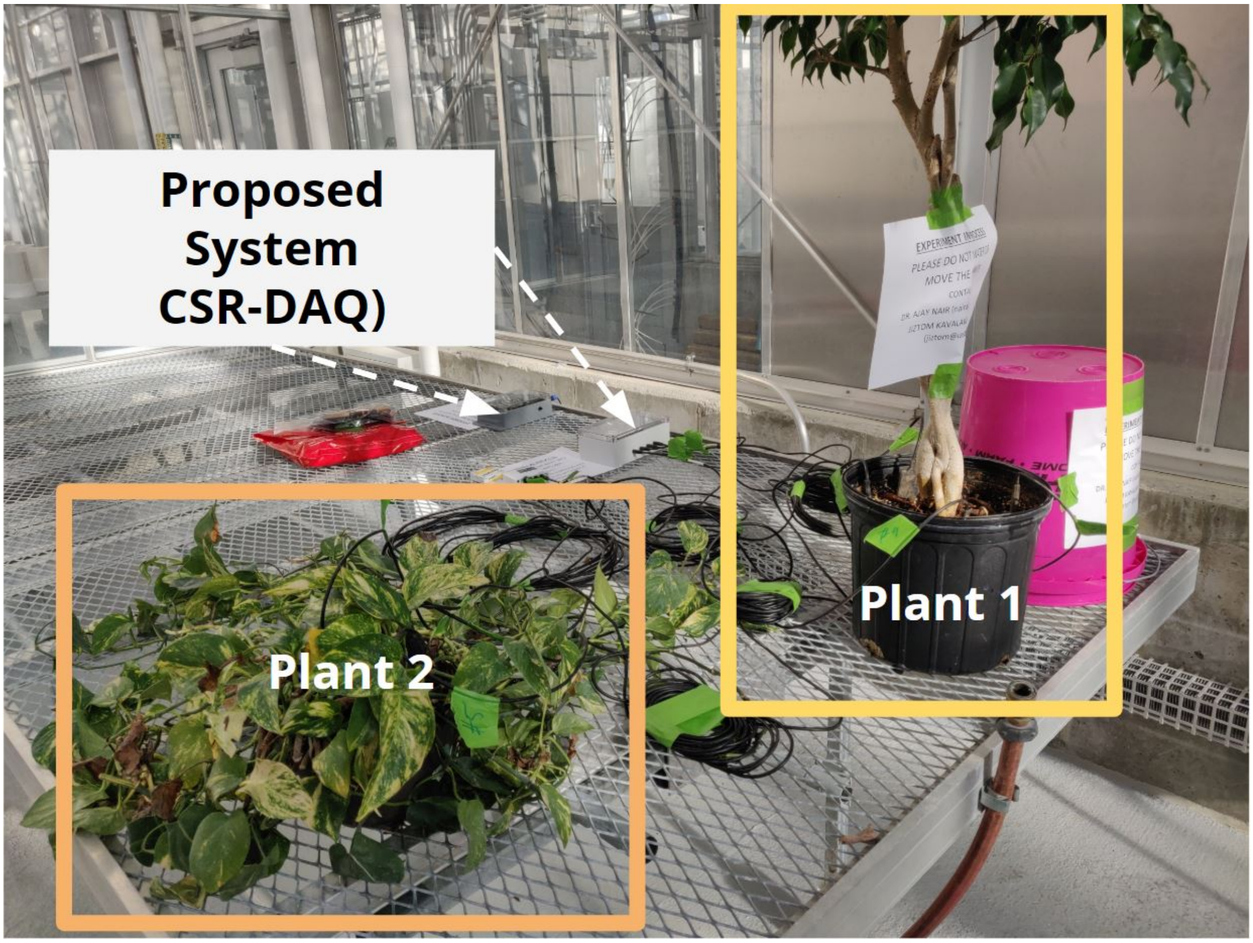


Figure 4.8 Phase 4 Test Bed Setup

Decagon sensors with the following parameters.

- $mV_{min} = 300mV$
- $mV_{max} = 1250mV$
- $VWC_{min} = 0.0\%$
- $VWC_{max} = 57.0\%$

$$VWC = \frac{mV - (mV_{min})}{mV_{max} - mV_{min}} * (VWC_{max} - VWC_{min}) + VWC_{min}$$

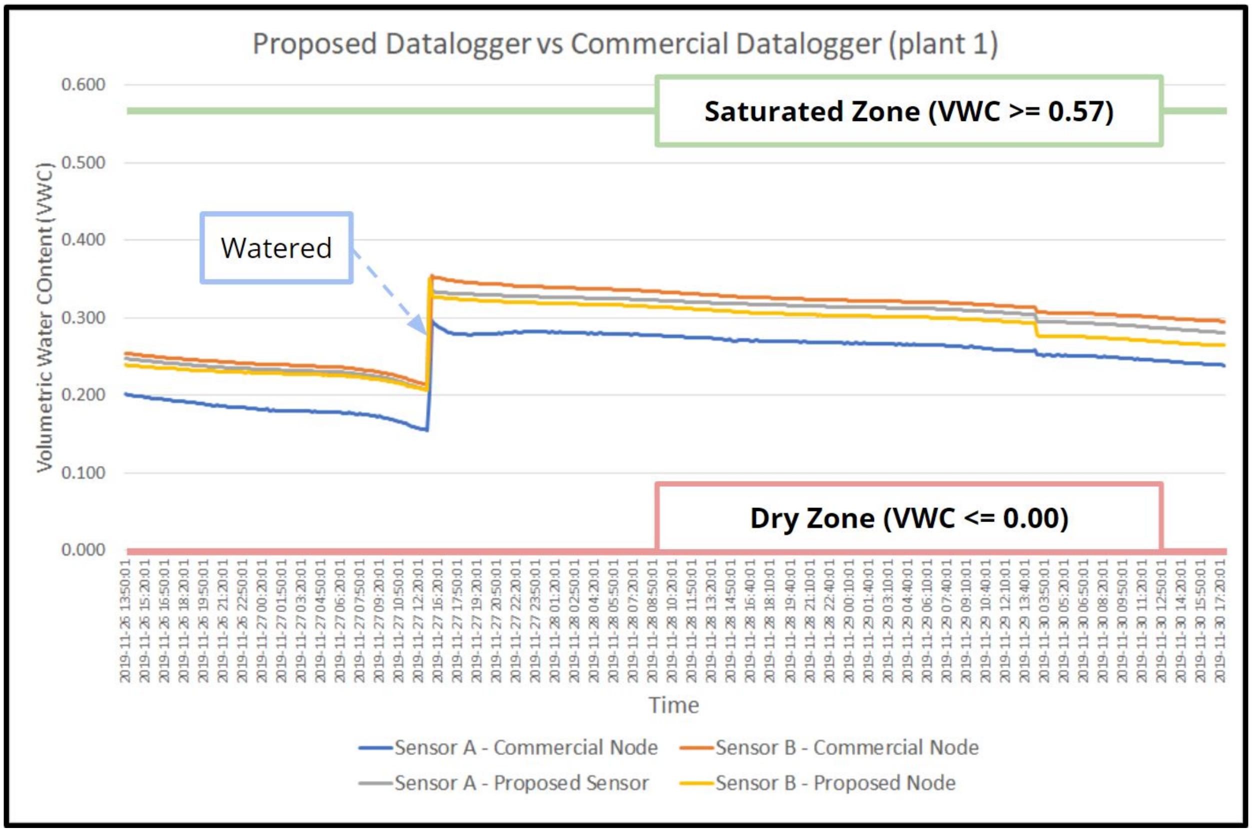


Figure 4.9 VWC Calibration of Proposed Data logger with Commercial Data logger (Plant 1)

$$VWC = \frac{mV - 300}{950} * (57.0) + 0.0 \tag{4.2}$$

Based on the collected reading and applying Equation 4.1 we get the following graphs Figure 4.9 and Figure 4.10. These have the following watering cycle.

- Plant 1: Watered only once on 11-27-2019.
- Plant 2: Watered twice on 11-27-2019 and 11-28-2019.

This was performed as the money plant has a smaller pot and was decided as a variation to the sensor readings. Another Dip is also observed in the graphs, this was due to a battery failure at the time of transmission. This resulted in an incomplete transmission that satisfies the criteria but fails data extraction in the Basex software. As the software at the moment does not have a self restart function it crashed.

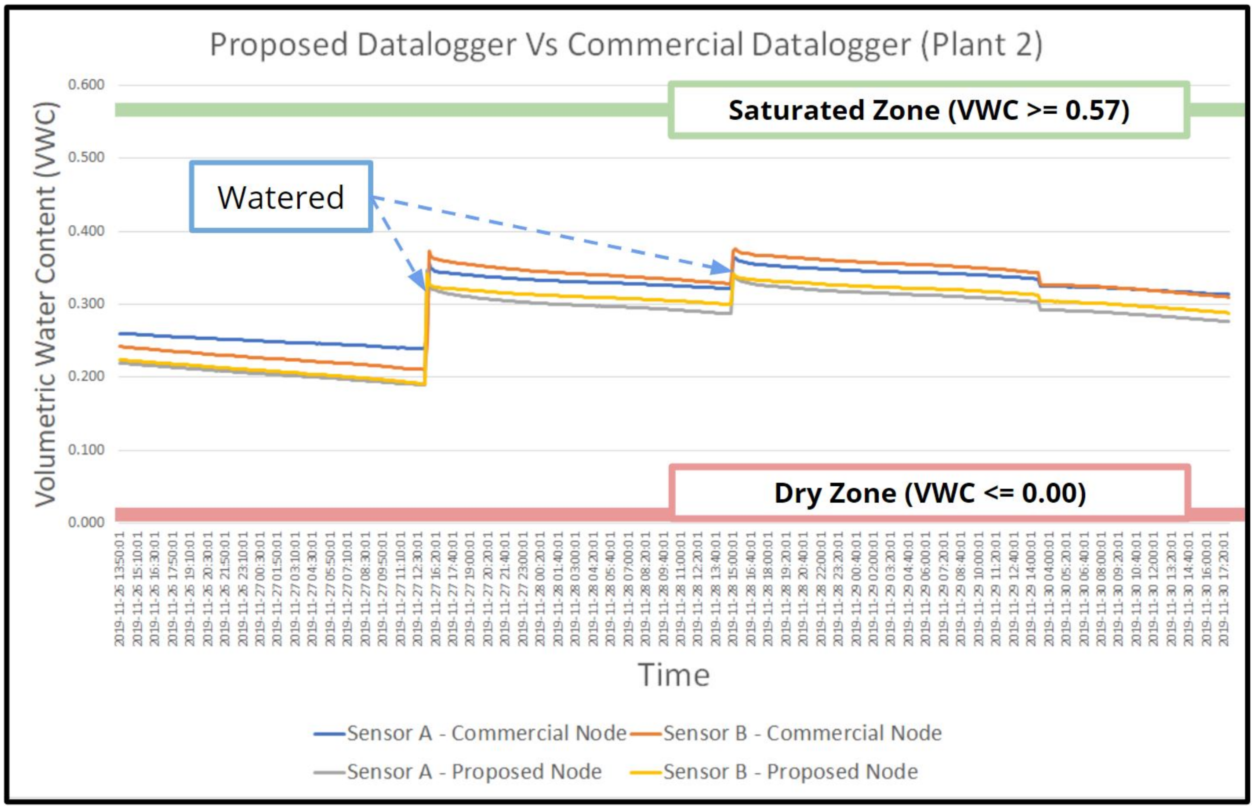


Figure 4.10 VWC Calibration of Proposed Data logger with Commercial Data logger (Plant 2)

4.2.4.1 VWC Plant 1

The four lines as shown in Figure 4.9 show the 4 sensors inserted into plant 1. 2 connected to the proposed Datalogger and the other two connected to the Commercial Em5b Datalogger. The results of the sensor output millivolts were measured in both the Data loggers and are used to compare and observe similar results. The Proposed data logger ADC values need to multiplier of 0.1875 and the commercial sensor a multiplier of 0.75. Once this value is computed, it can be used to find the VWC readings using the Equation 4.1. This on plotting is shown to have a reading where the commercial sensor was higher and the one sensor was lower. The variance in Values was very small with a limit of 10-30 mV which is within the tolerance range of scientific computation.

The plant was only watered once within the measurement cycle to reset the sensors to its maximum water saturation possible in the pot. Now to understand how accurate the values are we compared all the sensors with each other using a scatter plot. This can be seen here in Figure 4.12. The diagram shows that crowd along the middle trend line which tells a high R^2 value.

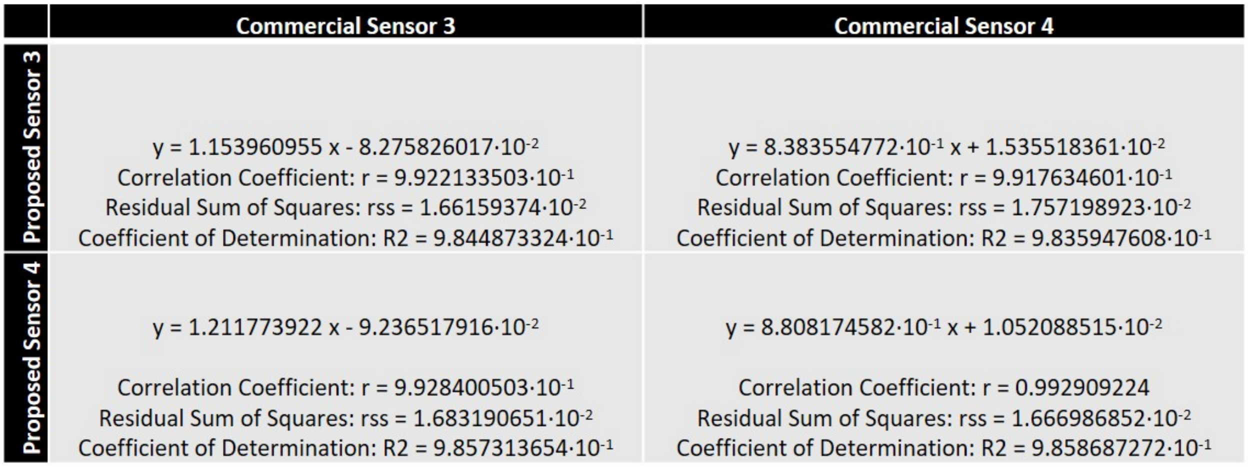
4.2.4.2 VWC Plant 2

The four lines as shown in Figure 4.10 show the 4 sensors inserted into plant 1. 2 connected to the proposed Datalogger and the other two connected to the Commercial Em5b Datalogger. The results of the sensor output millivolts were measured in both the Data loggers and are used to compare and observe similar results. The Proposed data logger ADC values need to multiplier of 0.1875 and the commercial sensor a multiplier of 0.75. Once this value is computed, it can be used to find the VWC readings using the Equation 4.1. This on plotting is shown to have a reading where the commercial sensor was higher and the one sensor was lower. The variance in Values was very small with a limit of 10-30 mV which is within the tolerance range of scientific computation.

Specifically, plant 2 was a smaller pot with a lowered amount of water it can hold. Hence the watering cycle was performed twice. As seen clearly in the graph. Also, to clarify the exact values an in-depth analysis of the sensor data was done. To approve its resolution and accuracy a linear regression was performed on the collected data. Figure 4.11 is a scatter matrix with all the linear regression results with 499 samples. As the produced R^2 values are around 0.98 value proves the authenticity of the system. This is again re-verified by the visual representation shown in Figure 4.13.

4.3 Power Analysis

The following gives the approximate statistics of the power consumption on the board. Due to a hardware issue with reverse diode in the p-channel MOSFET even though the control signal works



	Commercial Sensor 3	Commercial Sensor 4
Proposed Sensor 3	$y = 1.153960955\,x - 8.275826017\cdot10^{-2}$ Correlation Coefficient: $r = 9.922133503\cdot10^{-1}$ Residual Sum of Squares: $rss = 1.66159374\cdot10^{-2}$ Coefficient of Determination: $R2 = 9.844873324\cdot10^{-1}$	$y = 8.383554772\cdot10^{-1}\,x + 1.535518361\cdot10^{-2}$ Correlation Coefficient: $r = 9.917634601\cdot10^{-1}$ Residual Sum of Squares: $rss = 1.757198923\cdot10^{-2}$ Coefficient of Determination: $R2 = 9.835947608\cdot10^{-1}$
Proposed Sensor 4	$y = 1.211773922\,x - 9.236517916\cdot10^{-2}$ Correlation Coefficient: $r = 9.928400503\cdot10^{-1}$ Residual Sum of Squares: $rss = 1.683190651\cdot10^{-2}$ Coefficient of Determination: $R2 = 9.857313654\cdot10^{-1}$	$y = 8.808174582\cdot10^{-1}\,x + 1.052088515\cdot10^{-2}$ Correlation Coefficient: $r = 0.992909224$ Residual Sum of Squares: $rss = 1.666986852\cdot10^{-2}$ Coefficient of Determination: $R2 = 9.858687272\cdot10^{-1}$

Figure 4.11 VWC Regression analysis for plant 2

the devices remain at always-on state. This has caused the hardware to be running in the worst possible case with just the microcontroller going to sleep mode. The following is the theoretical considerations for the calculations:

- Required or regulated voltage: 5V
- Practical time the node was awake: 23 hours and 20 minutes at the full wake.
- Measurement interval: 10 min.
- Average time awake per cycle: 10 sec.
- Time sleep in an hour: 59 min.

Now based on Figure 4.14 we see the drain by the battery as per the four changes within the short time frame. This is insufficient data as it only gives the first 20 accurate readings. Additional information is required to make sense of the battery and the drain has been proved not to be a linear equation but more of a logarithmic, polynomial or exponential signal.

Now based on the collected data we will get the following calculations:

$y = -0.0328x + 6.2229$

Now as we find that the device is alive for 23hr20min we can make the following calculations

$count = (23 * 6) + 2$

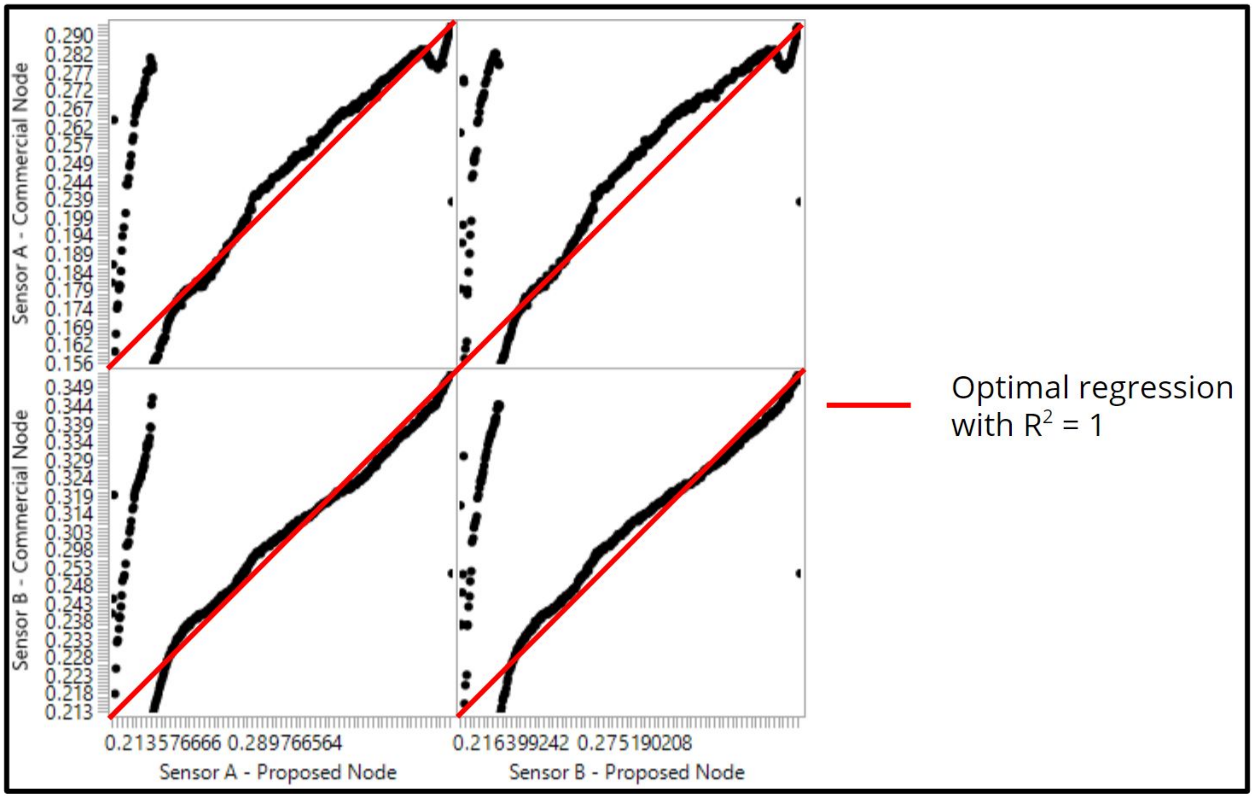


Figure 4.12 VWC Scatter Plot with Plant1

$count = 144$

Now based on this we can get the following:

$y = -0.0328 * 144 + 6.229$

$y = 1.560V$

This is not the best equation of the battery drain. A more clear battery drain equation can only be found based on longer data regression. As per this value we get the following results for 4.6V

$$\text{if } y = 4.6V \quad \text{count} = -(y - 6.229)/0.0328$$

$$\text{count} = -(4.6 - 6.229)/0.0328 = 49.4786$$

This is for each count being 10min. Now on expanding we get the following:

49.4786 count = 494.786 min

The Equation 4.3 is derived based on the design parameters to calculate the approximate lifetime

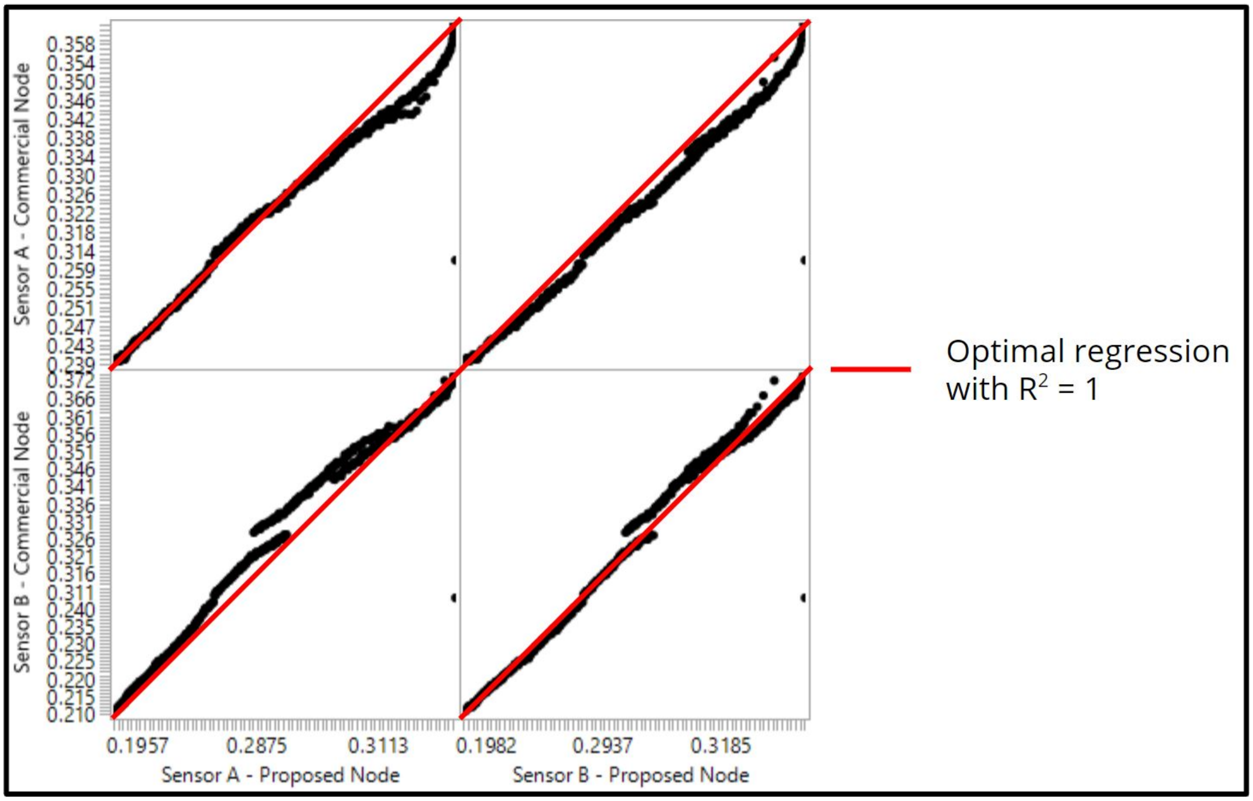


Figure 4.13 VWC scatter Plot with Plant 2

of the battery.

$$\text{battery lifetime}_{minutes} = \text{awake time}_{minutes}(61 - (\text{wake interval}_{minutes} * \text{wake count})) \tag{4.3}$$

Now converting we have the following theoretical results:

We conducted a rigorous test without sleep-mode for the duration of 5 days. Figure 4.14 shows the battery consumption for the period of 17 counts where each count elapses for 10 min. As shown in the figure, the battery voltage was 5.7 V at a count of 17. From the characteristics, we derived the curve-fitting function as $y = -0.0328x + 6.2229$, where y is the voltage and x is the time. We substituted the minimum voltage required for the buck converter, i.e., $y = 4.6$ V and observed the maximum time $x = 419$ min approximately. As stated in Equation 4.3 , we considered sleep-mode algorithm and obtained the expected battery lifetime as $x = 24813.5179$ min or 17.2316 days.

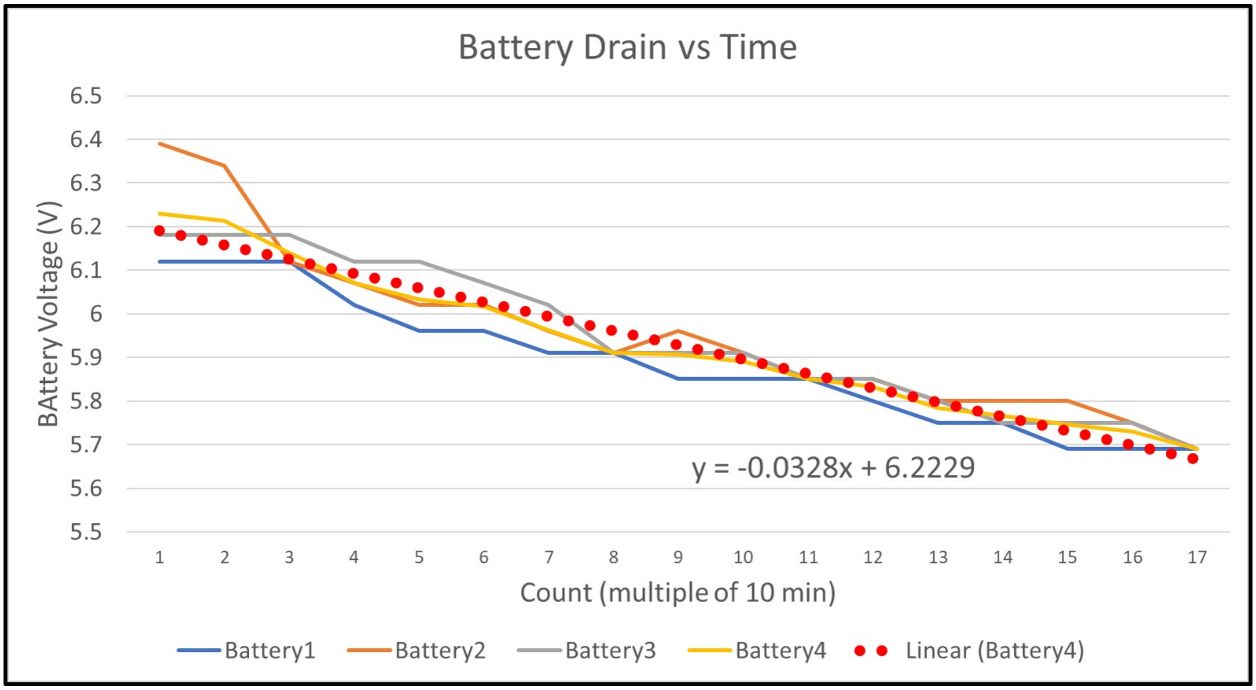


Figure 4.14 Power Drain vs Count

4.4 Cost Evaluation

This section looks deeper into the hardware costs and peeks into the future productivity of the project as a product to customers. The hardware mentioned here is a cost borne by the maker at MRP rates than procurement rates due to low volume. The analysis shows that the larger the number of products and more reduced components will reduce the price drastically.

We will compare the cost at two levels. In Table 4.2 we discuss the cost of building a single node at the manufacturing level. This is compared with making a prototype, 10 prototypes and also a small product line of 500 units. These show that cost per node can be reduced to less than a 100$.

Now we are assuming that a small scale farmer with a 2-acre plot, which has 8 sensors already installed. So, in order for the farmer to install such a system, it will require an exaggerator and

Table 4.2 Cost of Node

Component	Cost - Prototype	Cost - 10 prototype	Cost - production
PCB	33	3.4	0.94
Components	46.12	38.791	23.7349
HC-12 (2)	25.8	25.8	25.8
ADS1115	14.95	14.95	4
DS3231	13.95	13.95	6
Battery AA (4)	2	2	2
Protective case	17.3	15.78	14.04
Labour	0	0	10
Total	**153.12**	**114.671**	**86.5149**

two nodes. So defining the cost of an entire product at system range is discussed in Table 4.3. Now adding on to the cost of the whole system with 2 nodes and a base station:

Table 4.3 Cost of Entire System

Component	Cost - Prototype	Cost - 10 prototype	Cost - production
Basex Node(2)	306.24	229.342	179.04
Base Station	115	115	100
System Total	**421.24**	**344.342**	**279.04**

The entire analysis shows the feasibility of manufacture this product while still maintaining the target price that was observed in advance. Thus satisfying the client's need for being practical and functional.

CHAPTER 5. CONCLUSION

5.1 Summary

We proposed a CSR-DAQ architecture and developed a testbed-based embedded prototype. The prototype is an indigenous implementation of a cost-effective data acquisition system for precision agriculture. We conducted multiple tests in 4 phases for the VWC calibration and battery consumption characteristics. We observed the VWC under regression with the benchmark datalogger to have a R^2 value of over 98 %. This was visually confirmed in Figure 4.12 and Figure 4.13. We observed the battery consumption as 419 min without a sleep-mode algorithm and approximately 17 days with the sleep-mode algorithm. The modularity of the CSR-DAQ system allows for easy adaptation to different sensors and networking technologies such as Wi-Fi, Bluetooth, Zigbee, LoRa, etc.

CSR-DAQ used the radio of the 433 MHz to provide broadcast communication to transfer technology. The hardware used specifically is HC-12 433 MHz Radio module. The extremely popular prototyping hardware architecture - Arduino allows for easy collection and transmission of sensor values. MQTT was implemented using the IoT-core options from cloud services such as Amazon AWS / Google Cloud to securely transmit data from sensor aggregator to the cloud storage. Cloud-based RDS solutions from Amazon AWS / Google Cloud were used as solutions to deal with a large amount of collected data and provide statistics for future use. Customized Software provides results of the query is used to generate graphs and other visualization tools to help the user to understand the environment scientifically. These information are tabulated in Table 5.1.

The proposed system was implemented both in a controlled environment and out in the field. The tests successfully verified the proposed architecture. The results of the final phase showed a R^2

value greater than 98% which deems the system to be field worthy in comparison to a commercially available data logger. The results are clearly explained in Chapter 4.

The solution will allow the user to automate the water management system in an outdoor environment, Improve the quality and quantity of the crops by proper resource management and achieve Economic efficiency. These can also be used for an in-depth study of the soil quality and its erosion to time.

The salient feature is the result of trial and error over the hardware design as shown in chapter 4. It can also be confirmed with the results that CSR-DAQ works as designed in the field. And on calibrating more sensors we can effectively use it in the field as a cost-effective alternative for the current industrial and literature solutions.

Table 5.1 Updated Comparison table with industry standards and proposed hardware

Parameter	**HOBOnet**	**PYNCO**	**NCD Sensor**	**em5b**	**DL6**	**SCR-DAQ**
Cost for entire system	$1500+	$2649	$219.95 + base station	$300	>$200 (quote)	less than $250
No of sensors per node	Base station up to 10. nodes 1-4	5 basic. Soil moisture sensor can vary.	1	4	8	4x
Connection to Cloud	Wi-Fi or cellular	Cellular	ZigBee Mesh	None	None	433 Mhz Radio Wi-Fi
Power supply	Battery + Solar	Solar	Battery	AA	AA	4 AA
Ease of install	Require installation	Plug and play	Plug and play	Plug and play	Plug and play	Plug and Play
Data Visualization	HOBOlink proprietary cloud	Centralized Cloud Console	User preference	User defined CSV	??	Centralized cloud Console - Amazon AWS / Google Cloud
Sensor options	Varied and additional hardware	Fixed per module type.	3-4 options with ruggedized features	Commercial sensor from Decagon	Soil moisture sensors sold by manufacturer.	Any analog sensors added to calibration.

5.2 Future Work

The following are the scope for future improvements for CSR-DAQ:

- **Multi Sensor support:** CSR-DAQ can be calibrated to support a wider range of analog sensors. This will allow more users for the faster switch from traditional data loggers.
- **Cloud DB dynamic update:** A more steady and robust connection to the cloud services with a scalable architecture. The past data set can be used as an input stream for predicting future environmental characteristics. On linking with climate data, the tool can become a more efficient and effective solution for precision agriculture.
- **Web apps :** A customized set of web apps to suit user needs and trigger mechanisms to alert or react to certain parameters. A personalized use space with graphs of different locations, predictions, and results will enable an easier understanding of the field to the user.
- **Network Technologies:** Apply CSR-DAQ architecture with different networking technologies to calculate the range and its effectiveness.
- **Solar Power :** Adding a solar panel and related circuitry to make the system self-sustainable. This will allow for prolonging the field deployment.

Based on the proposed CSR-DAQ architecture as a platform the following can be developed:

- **Research solutions:** A more robust and ruggedized version can be used in volatile areas such as volcano, frigid areas, etc to monitor its conditions.
- **Predict Environmental characteristics :** On applying machine learning with other information sets such as weather, wind, humidity, etc a more precise prediction of local parameters to optimize practices in an outdoor environment.

BIBLIOGRAPHY

[1] S. Z. Nasir, "Introduction to arduino uno," May 2019. [Online]. Available: https://www.theengineeringprojects.com/2018/06/introduction-to-arduino-uno.html

[2] I. S. of Precision Agriculture, "Precision ag definition." [Online]. Available: https://www.ispag.org/

[3] M. Rouse, L. Rosencrance, M. Rouse, and M. Rouse, "What is internet of things (iot)? - definition from whatis.com." [Online]. Available: https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT

[4] "Soil moisture sensor tutorial." [Online]. Available: https://www.vegetronix.com/TechInfo/Soil-Moisture-Sensor-Primer.phtml

[5] D. R. Cobos, "Why does my soil moisture sensor read negative?" [Online]. Available: http://manuals.decagon.com/RetiredandDiscontinued/Slicksandcontent/Presentations/SoilMoisture301.pdf

[6] "What is data acquisition?" [Online]. Available: http://www.ni.com/data-acquisition/what-is/

[7] T. Khoa, M. Man, T.-Y. Nguyen, V. Nguyen, and N. Nam, "Smart agriculture using iot multi-sensors: A novel watering management system," *Journal of Sensor and Actuator Networks*, vol. 8, p. 45, 08 2019.

[8] D. Fernández, P. Sánchez, B. Álvarez, J. A. López, and A. Iborra, "Triot: A proposal for deploying teleo-reactive nodes for iot systems," in *Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection*, Y. Demazeau, P. Davidsson, J. Bajo, and Z. Vale, Eds. Cham: Springer International Publishing, 2017, pp. 70–81.

[9] P. Snchez, B. lvarez, E. Antolinos, D. Fernndez, and A. Iborra, "A teleo-reactive node for implementing internet of things systems," *Sensors (Basel, Switzerland)*, vol. 18, no. 4, April 2018. [Online]. Available: http://europepmc.org/articles/PMC5948476

[10] Y. Wang, Y. Wang, X. Qi, and L. Xu, "Opaims: Open architecture precision agriculture information monitoring system," in *Proceedings of the 2009 International Conference on Compilers, Architecture, and Synthesis for Embedded Systems*, ser. CASES '09. New York, NY, USA: ACM, 2009, pp. 233–240. [Online]. Available: http://doi.acm.org/10.1145/1629395.1629428

[11] J. Hwang, C. Shin, and H. Yoe, "Study on an agricultural environment monitoring server system using wireless sensor networks," *Sensors (Basel, Switzerland)*, vol. 10, pp. 11 189–211, 12 2010.

[12] M. Pande, N. K. Choudhari, and S. Pathak, “Energy efficient hybrid architecture and positioning of sensors in wsn for precision agriculture,” in *Proceedings of the CUBE International Information Technology Conference*, ser. CUBE ’12. New York, NY, USA: ACM, 2012, pp. 198–203. [Online]. Available: http://doi.acm.org/10.1145/2381716.2381754

[13] A. Brugués, S. Bromuri, and M. Schumacher, “Demonstration of magpie: An agent platform for monitoring chronic diseases on android,” in *Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection*, Y. Demazeau, P. Davidsson, J. Bajo, and Z. Vale, Eds. Cham: Springer International Publishing, 2017, pp. 299–302.

[14] J. Wang, J. Cao, R. S. Sherratt, and J. H. Park, “An improved ant colony optimization-based approach with mobile sink for wireless sensor networks,” *The Journal of Supercomputing*, vol. 74, no. 12, pp. 6633–6645, Dec 2018. [Online]. Available: https://doi.org/10.1007/s11227-017-2115-6

[15] S. Biswas, R. Das, and P. Chatterjee, “Energy-efficient connected target coverage in multi-hop wireless sensor networks,” in *Industry Interactive Innovations in Science, Engineering and Technology*, S. Bhattacharyya, S. Sen, M. Dutta, P. Biswas, and H. Chattopadhyay, Eds. Singapore: Springer Singapore, 2018, pp. 411–421.

[16] P. Hehenberger, B. Vogel-Heuser, D. Bradley, B. Eynard, T. Tomiyama, and S. Achiche, “Design, modelling, simulation and integration of cyber physical systems: Methods and applications,” *Computers in Industry*, vol. 82, pp. 273 – 289, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0166361516300902

[17] A. Tzounis, N. Katsoulas, T. Bartzanas, and C. Kittas, “Internet of things in agriculture, recent advances and future challenges,” *Biosystems Engineering*, vol. 164, pp. 31 – 48, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1537511017302544

[18] J. Ruan, Y. Wang, F. T. S. Chan, X. Hu, M. Zhao, F. Zhu, B. Shi, Y. Shi, and F. Lin, “A life cycle framework of green iot-based agriculture and its finance, operation, and management issues,” *IEEE Communications Magazine*, vol. 57, no. 3, pp. 90–96, March 2019.

[19] A. Khanna and S. Kaur, “Evolution of internet of things (iot) and its significant impact in the field of precision agriculture,” *Computers and Electronics in Agriculture*, vol. 157, pp. 218 – 231, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168169918316417

[20] R. Almeida, R. Oliveira, M. Lus, C. Senna, and S. Sargento, “A multi-technology communication platform for urban mobile sensing,” *Sensors*, vol. 18, no. 4, 2018. [Online]. Available: https://www.mdpi.com/1424-8220/18/4/1184

[21] M. Cerchecci, F. Luti, A. Mecocci, S. Parrino, G. Peruzzi, and A. Pozzebon, “A low power iot sensor node architecture for waste management within smart cities context,” *Sensors*, vol. 18, no. 4, 2018. [Online]. Available: https://www.mdpi.com/1424-8220/18/4/1282

[22] . Lozano, J. Caridad, J. F. De Paz, G. Villarrubia Gonzlez, and J. Bajo, “Smart waste collection system with low consumption lorawan nodes and route optimization,” *Sensors*, vol. 18, no. 5, 2018. [Online]. Available: https://www.mdpi.com/1424-8220/18/5/1465

[23] J. Muangprathub, N. Boonnam, S. Kajornkasirat, N. Lekbangpong, A. Wanichsombat, and P. Nillaor, “Iot and agriculture data analysis for smart farm,” *Computers and Electronics in Agriculture*, vol. 156, pp. 467 – 474, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0168169918308913

[24] “Hobonet field monitoring system.” [Online]. Available: https://www.onsetcomp.com/hobonet

[25] “Pycno core - all-in-one soil sensors.” [Online]. Available: https://get.pycno.co/collections/pycno-core-all-in-one-soil-sensors

[26] “Industrial iot wireless soil moisture sensor transmitter.” [Online]. Available: https://store.ncd.io/product/wireless-soil-moisture-sensor-greenhouse-iot/

[27] “Em5b-datasheet.” [Online]. Available: http://library.metergroup.com/Manuals/10697_Em5b\Data\Collection\System_Web.pdf

[28] “Data loggers.” [Online]. Available: http://www.dynamax.com/products/data-loggers/dl6-soil-moisture-data-logger

[29] S. Nathan, “Adc conversion sparkfun.” [Online]. Available: https://learn.sparkfun.com/tutorials/analog-to-digital-conversion/relating-adc-value-to-voltage

[30] T. H. Team. [Online]. Available: https://www.hivemq.com/blog/mqtt-essentials-part-1-introducing-mqtt/

[31] “Decagon 10hs datasheet.” [Online]. Available: http://publications.metergroup.com/Manuals/20426_10HS_Manual_Web.pdf

[32] “atmega 32x datasheet 2018,” 2018. [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/ATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061A.pdf

[33] “Ds3231,” Mar 2015. [Online]. Available: https://datasheets.maximintegrated.com/en/ds/DS3231.pdf

[34] A. Industries, “Ads1115 16-bit adc - 4 channel with programmable gain amplifier.” [Online]. Available: https://www.adafruit.com/product/1085

[35] “hc-12 wireless serial port communication module datasheet 2012,” Oct 2012. [Online]. Available: https://www.elecrow.com/download/HC-12.pdf

[36] “raspberry pi 4 model b specifications raspberry pi.” [Online]. Available: https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/

[37] “Aws documentation.” [Online]. Available: https://docs.aws.amazon.com/index.html?nc2=h_ql_doc_do

[38] “The world’s most advanced open source relational database.” [Online]. Available: https://www.postgresql.org/

APPENDIX. ADDITIONAL MATERIAL

Schematics

The following are the hardware circuit designs for the PCB. These were designed using Circuit Maker software from Altium.

1. Power control components - High Side driver.
2. Micro controller circuit.
3. Battery power circuit.
4. ADC and senor control.
5. USB Debugger.

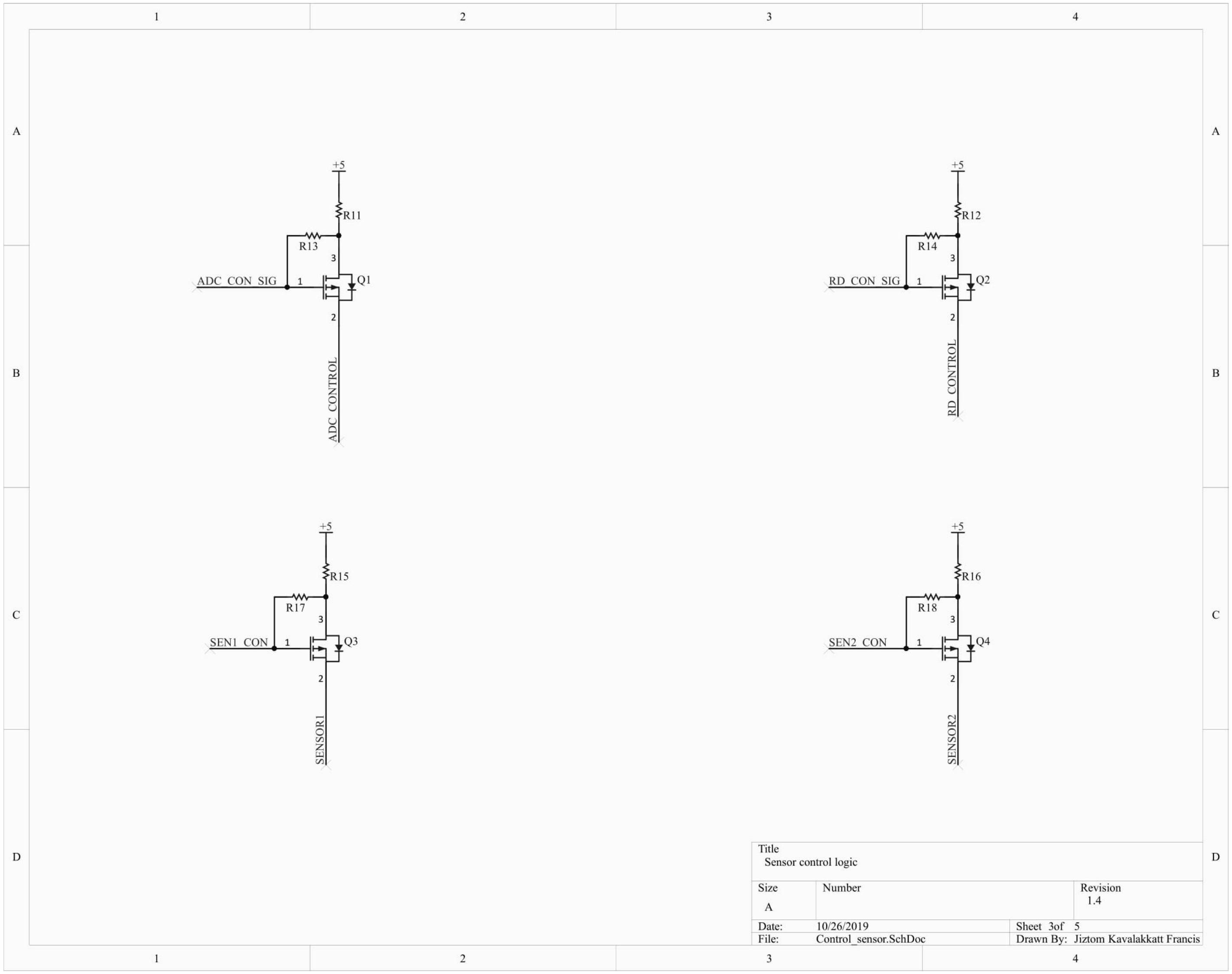


Figure .1 Power control components - High side driver

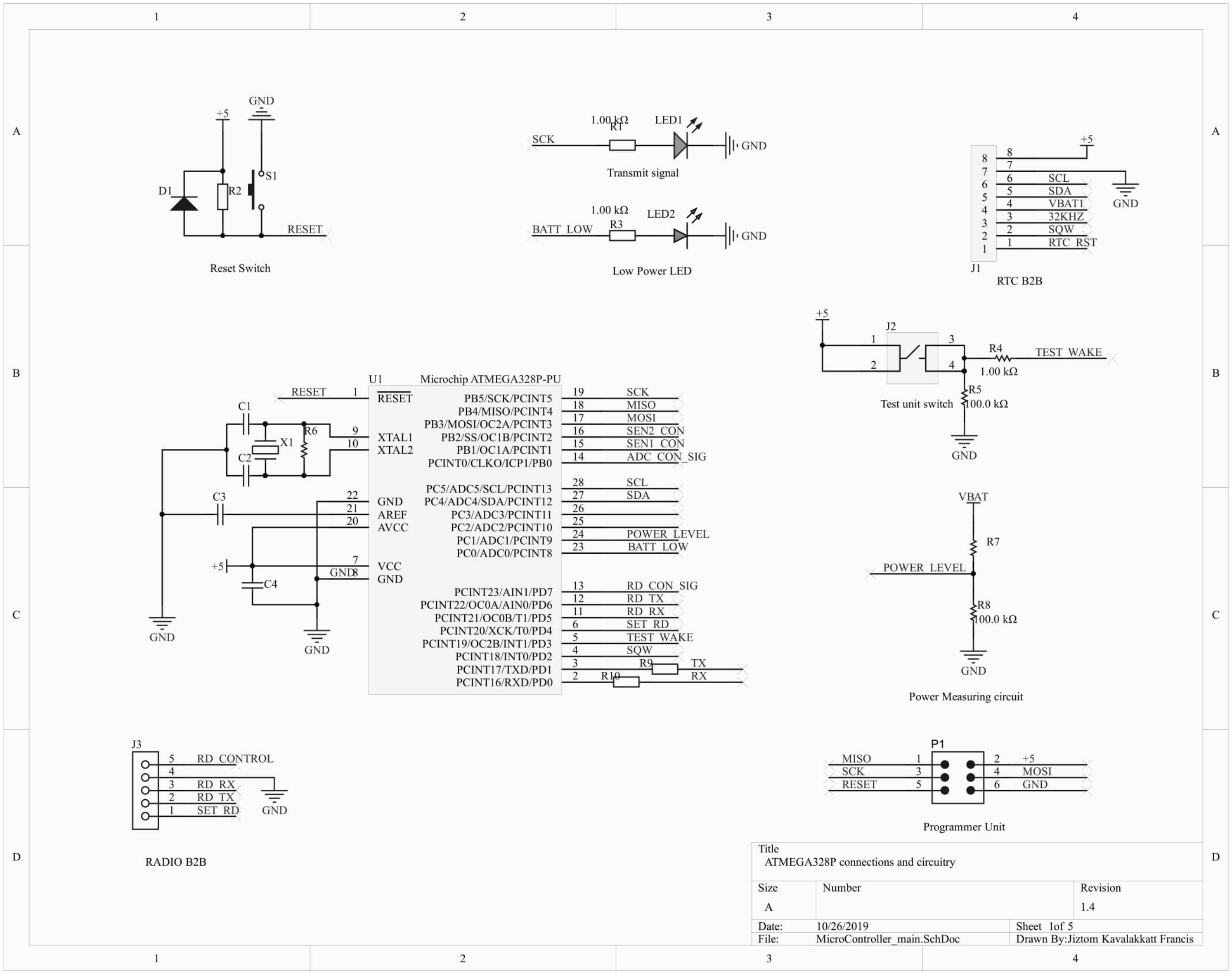


Figure .2 Micro controller Circuit

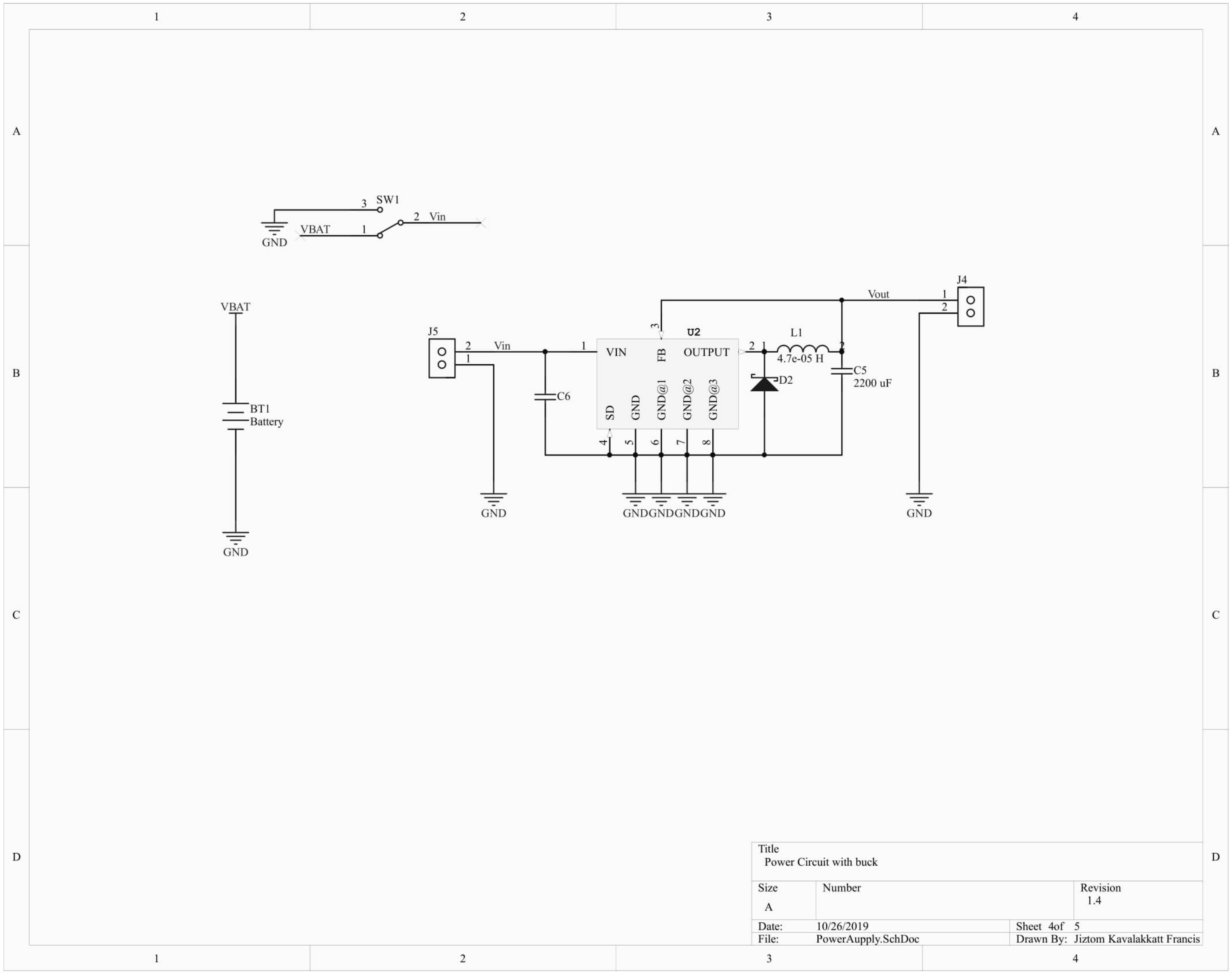


Figure .3 Battery power circuit

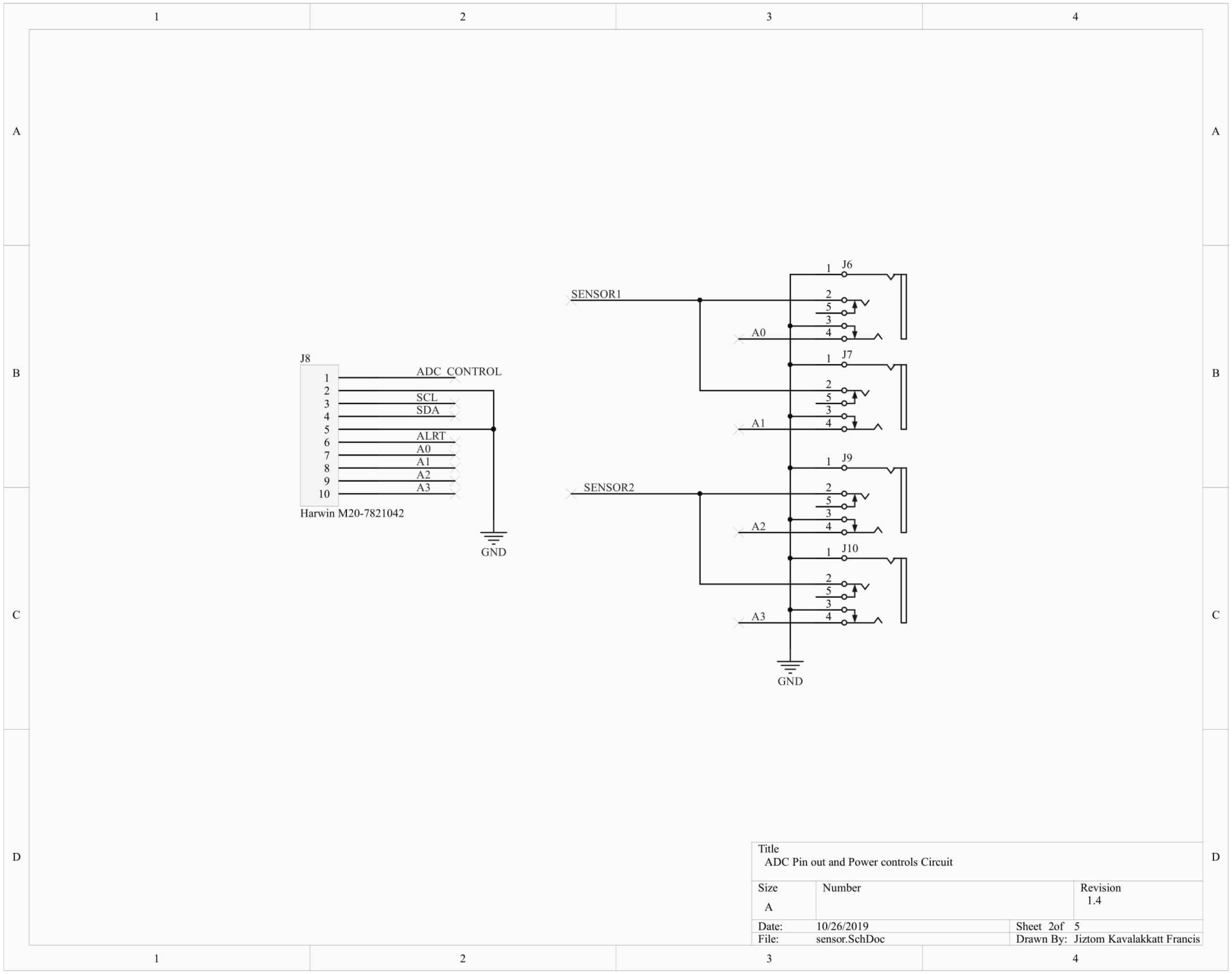


Figure .4 ADC and sensor controls

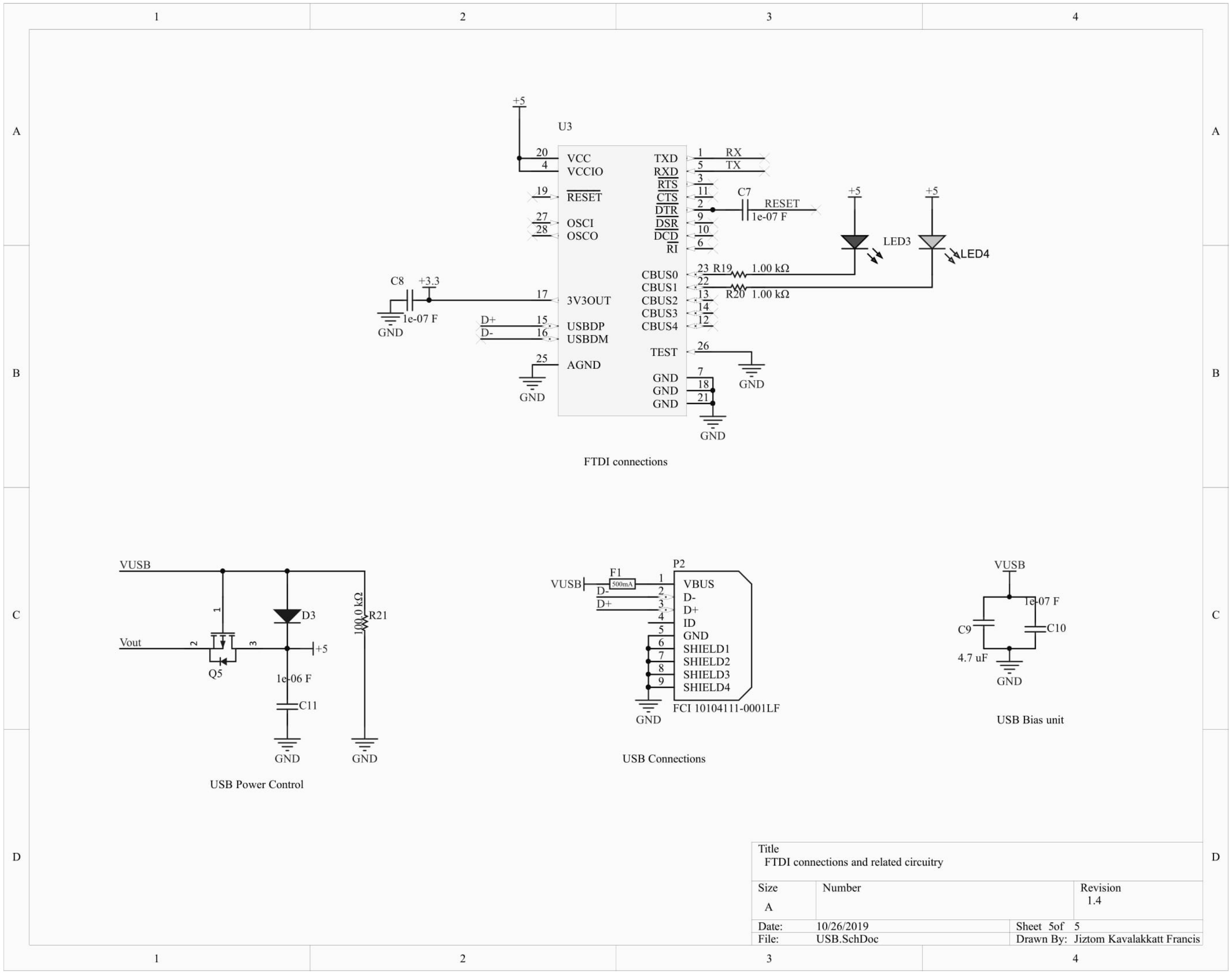


Figure .5 USB Debugger

PCB models

The following are the 3D for the generated PCB. These were designed using Circuit Maker software from Altium.

1. PCB 3D model.

2. PCB 2D model.

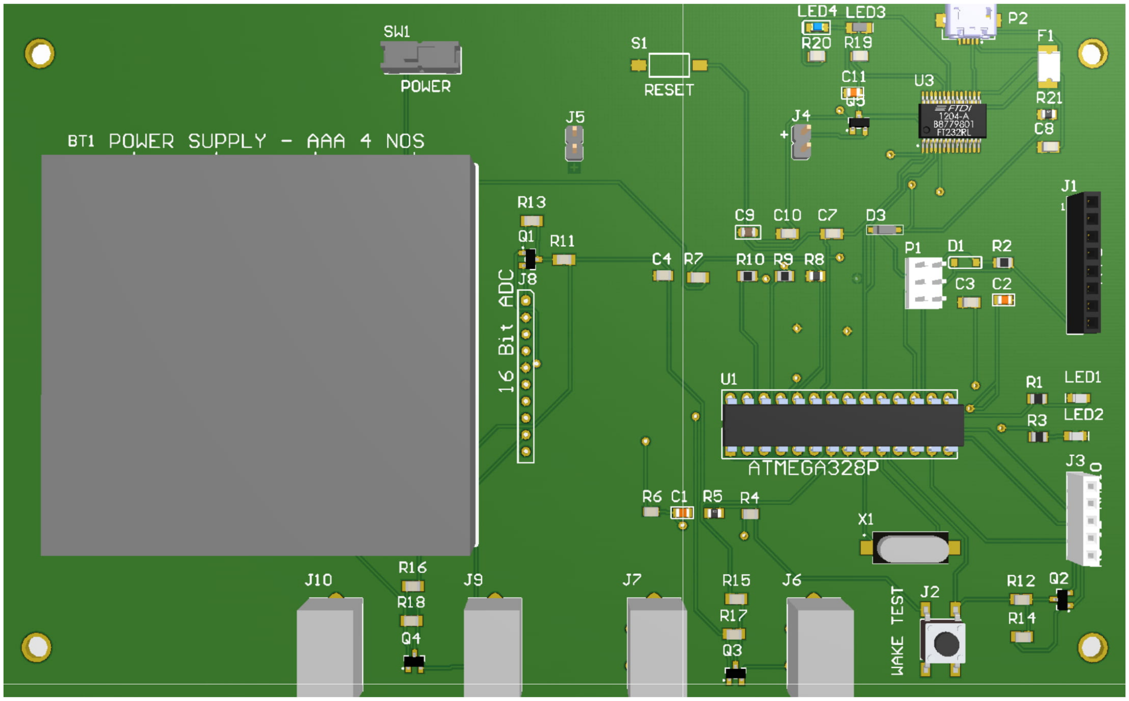


Figure .6 PCB 3D model

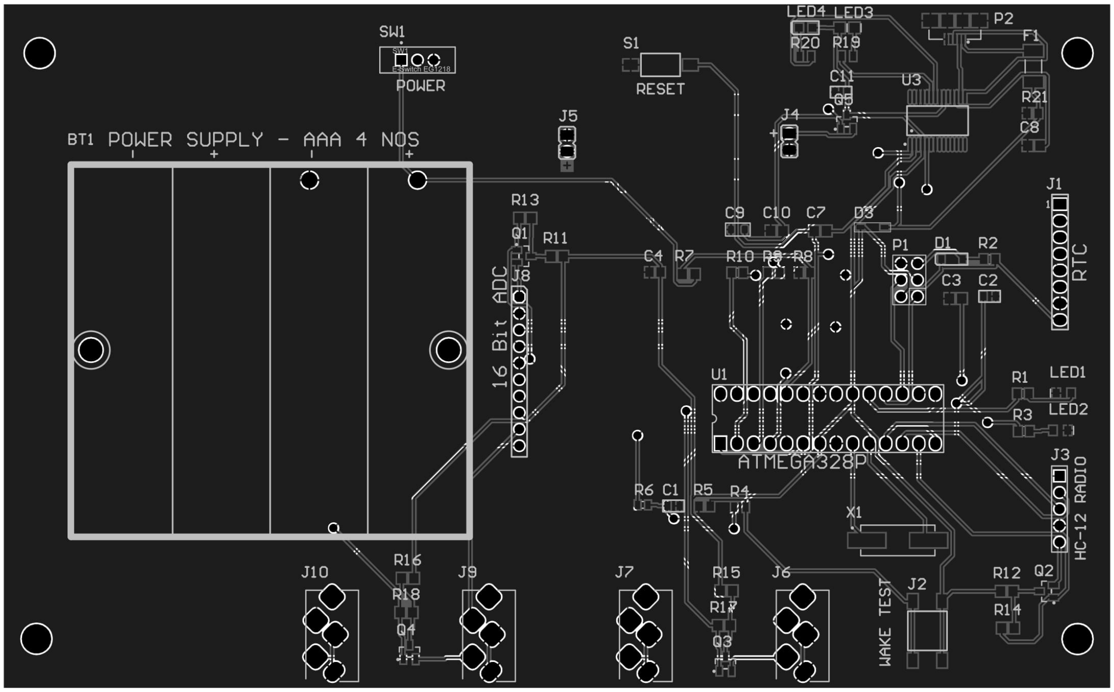


Figure .7 PCB 2D model