MONITORING AND FORECASTING SYSTEM FOR HYDROELECTRIC POWER PLANTS USING IoT

A

DISSERTATION

Submitted in partial fulfilment of the

Requirement for the award of the degree

Of

MASTER OF TECHNOLOGY

In

ALTERNATE HYDRO ENERGY SYSTEMS

By

Y RAM KUMAR

(17512016)

DEPARTMENT OF HYDRO AND RENEWABLE ENERGY

INDIAN INSTITUTE OF TECHNOLOGY, ROORKEE

ROORKEE- 247667 (INDIA)

June, 2019

CANDIDATE'S DECLARATION

I hereby declare the report which is being presented in this Dissertation, entitled, "**Monitoring and Forecasting System For Hydroelectric Power Plants using IoT"** in partial fulfilment of the requirement for the award of the degree of **Master of Technology** with specialization in "**Alternate Hydro Energy Systems**", submitted in **Department of Hydro And Renewable Energy, Indian Institute of Technology Roorkee,** is an authentic record of my own work carried out during the period from July 2018 to June 2019 under the supervision and guidance of **Dr. M. K. SINGHAL,** Associate professor, Department of Hydro And Renewable Energy, Indian Institute of Technology Roorkee.

I also declare that I have not submitted the matter embodied in this report for award of any other degree.

Date: June, 2019

Place: Roorkee. **(Y RAM KUMAR)**

CERTIFICATE

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Dr. M. K. Singhal

Associate professor,

Hydro And Renewable Energy Department,

Indian Institute of Technology,

Roorkee-247667.

ACKNOWLEDGEMENT

I hereby declare that the work which is presented in this project, entitled, "**Monitoring and Forecasting System For Hydroelectric Power Plants using IoT**", submitted in partial fulfilment of the requirement for the award of the degree of Master of Technology in **—Alternate Hydro Energy Systems** in **Department of Hydro and Renewable Energy, Indian Institute of Technology Roorkee**, is an authentic record of my own work carried out during the period from July 2018 to June 2019 under the supervision and guidance of Dr. M.K. Singhal, Associate professor, Department of Hydro And Renewable Energy, Indian Institute of Technology Roorkee, Roorkee (India), and special thanks to Mr. Yoshinobu Watanabe, Senior Technical Engineer and Mr. Hiroshi Nakayama, President of NAKAYAMA iron works Saga, Japan. For giving permission to test this project porotype in Shiraito Falls MHPP Japan.

Last but not the least I am also grateful to all faculty members and staff of Alternate Hydro Energy Centre, Indian Institute of Technology Roorkee.

I extend my thanks to all classmates who have given their full cooperation and valuable suggestions for my dissertation work. I also declare that I have not submitted the matter embodied in this report for award of any other degree.

Date: June 2019

Place: Roorkee **(Y RAM KUMAR)**

ABSTRACT

Electricity is one of the most important blessings that science has given to mankind. It has also become a part of modern life and one cannot think of a world without it. Electricity has many uses in our day to day life. In this report we mainly focus on hydro-electric power plants parameters.

A failure of a hydro power machine can lead to a catastrophic incident, like loss of power availability, loss of a machine, or loss of a complete plant. So, in this report we focus mainly on sensing the real time parameters of hydroelectric power plant like power generated, guide vane position, water pressure, and control panel temperature bearing temperature and so on. Using respective transducers.

Once the data sensing was done, the conversion of data to numerical digital levels using ADC, SDA, and SPI technologies are processed and store them in AWS RDS MySQL database to enable easy scalability. Building user friendly monitoring system using HTTP protocol and latest IoT web-technologies to facilitate Hydropower plant owners to keep eye on plant parameters and performance from anywhere globally in any device and operating systems.

Finally in this report a 3D printed IoT based data transmitter designing and manufacturing using Global System for Mobile Communications (GSM) dual band GPRS method was shown. R-Programming was used in this report to develop forecasting algorithm, in this algorithm Seasonal native model, Exponential smoothing model, and Autoregressive Integrated Moving Average shortly names as ARIMA model are used to predict the range of future values by recording data from SHIRAITO MHPP SAGA, JAPAN for 6 hours. From this predictions prevention measures to mechanical parts can be taken before big damage happen and create big economical loss.

Keywords: *Monitoring system, Forecasting system, Internet of thing (IOT), Relational database services (RDS), Global System for Mobile Communications dual band (GSM), and Autoregressive Integrated Moving Average (ARIMA).*

LIST OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

S no.	Symbols	Description
1	IoT	Internet of Things
2	ADC	Analog-to-digital converter
3	MHPP	Micro Hydro Power Plant
4	SPI	Serial Peripheral Interface
5	RDS	Relational Database Services
6	MySQL	My Structured Query Language
7	HTTP	Hypertext Transfer Protocol
8	GPRS	General Packet Radio Service
9	GSM	Global System for Mobile Communications
10	PLC	Programmable Logic Controllers
11	DHT	Digital Humidity and Temperature
12	LP	Linear Potentiometer
13	UART	Universal asynchronous receiver-transmitter
14	LDR	Light Dependent Resistor
15	AWS	Amazon Web Services
16	ETS	Exponential smoothing model
17	AR	Autoregressive model
18	MA	Moving Average model
19	ARIMA	Autoregressive Integrated Moving Average
20	ACF	Auto correlative Function
21	PACF	Partial Auto Correlative Function
22	AIC	Akaike Information Criterion
23	BIC	Bayesian information criterion

CHAPTER-1

INTRODUCTION

1.1 Background

Development of hydropower plants began toward the finish of the nineteenth century. The primary power plant depends on the rule of the substitute current concocted by Nikola Tesla was put into activity on the Niagara Falls in 1881. Since the occasion, the hydro control plant development innovation has been effectively spread all around the globe and today hydro control plants produce about 15.8 % of the complete worldwide age of the electric energy. The measure of units has been bitten by bit expanded from beginning little powers of a couple of kilowatts amid the opportunity to current forces of a few hundred megawatts. Restoration of gear empowers a duration of the age of the electric power from the water potential by existing force plants in a sheltered way at a satisfactory venture. Speculations into the restoration of existing plants are altogether lower than ventures into new hydro control plants since structural designing offices don't require any reinvestment and their esteem sums up to 80 % of the venture into another power plant.

Practically speaking, refurbishment comprises all the parts of a hydro power plant including civil engineering parts of the facility (dams, tunnels, power houses, input and output buildings, etc.), hydro-mechanical equipment and electric and mechanical equipment.

1.2 Renewable energy Power

By 2040, the renewable power source is anticipated to rise to coal and gaseous petrol power age. A few wards, including Denmark, Germany, the province of South Australia and some US states have accomplished high reconciliation of variable renewables. For instance, in 2015 breeze control satisfied 42% of power need in Denmark, 23.2% in Portugal and 15.5% in Uruguay. Interconnectors empower nations to adjust power frameworks by permitting the import and fare of a sustainable power source. Imaginative half breed frameworks have risen among nations and districts.

a) **Wind Power:** The 845 MW Shepherds Flat Wind Farm close Arlington, Oregon, US Airflows can be utilized to run wind turbines. Current utility-scale wind turbines run from around 600 kW to 5 MW of evaluated control, in spite of the fact that turbines with an appraised yield of 1.5-3 MW have turned into the most widely recognized for business use. The biggest generator limit of a solitary introduced coastal breeze turbine achieved 7.5 MW in 2015. The power accessible from the breeze is a component of the 3D shape of the breeze speed, so as wind speed builds, control yield increments up to the most extreme yield for the specific turbine Areas where winds are more grounded and increasingly steady, for example, seaward and high elevation destinations, are favoured areas for wind ranches. Regularly, full burden long periods of wind turbines shift somewhere in the range of 16 and 57 percent every year, except maybe higher in especially positive seaward destinations.

Wind-produced power met almost 4% of worldwide power request in 2015, with about 63 GW of new wind control limit introduced. Wind vitality was the main wellspring of new limit in Europe, the US and Canada, and the second biggest in China. In Denmark, wind vitality met over 40% of its power request while Ireland, Portugal, and Spain each met about 20%.

All inclusive, the long haul specialized capability of wind energy is accepted to be multiple times complete flow worldwide vitality creation, or multiple times flow power request, expecting every single handy boundary required were survived. This would require wind turbines to be introduced over enormous territories, especially in zones of higher breeze assets, for example, seaward. As seaward wind speeds normal ~90% more prominent than that of land, so seaward assets can contribute generously more vitality than land positioned turbines in 2014 worldwide breeze age was 706 terawatt-hours or 3% of the world's absolute power.

b) **Solar Power:** Solar energy, radiant light and heat from the sun, is harnessed using a range of ever-evolving technologies such as solar heating, photovoltaics, concentrated solar power (CSP), concentrator photovoltaics (CPV), solar architecture and artificial photosynthesis. Solar technologies are broadly characterized as either passive solar or active solar depending on the way they capture, convert and distribute solar energy. Passive solar techniques include orienting a building to the Sun, selecting materials with favourable thermal mass or light dispersing properties, and designing spaces that naturally circulate air. Active solar technologies encompass solar thermal energy, using solar collectors for heating, and solar power, converting sunlight into electricity either directly using photovoltaics (PV), or indirectly using concentrated solar power (CSP).

A photovoltaic system converts light into electrical direct current (DC) by taking advantage of the photoelectric effect Solar PV has turned into a multi-billion, fast-growing industry, continues to improve its cost-effectiveness, and has the most potential of any renewable technologies together with CSP. Concentrated solar power (CSP) systems use lenses or mirrors and tracking systems to focus a large area of sunlight into a small beam. Commercial concentrated solar power plants were first developed in the 1980s. CSP-Stirling has by far the highest efficiency among all solar energy technologies.

In 2011, the International Energy Agency said that "the development of affordable, inexhaustible and clean solar energy technologies will have huge longer-term benefits. It will increase countries' energy security through reliance on an indigenous, inexhaustible and mostly import-independent resource, enhance sustainability, reduce pollution, lower the costs of mitigating climate change, and keep fossil fuel prices lower than otherwise. These advantages are global. Hence the additional costs of the incentives for early deployment should be considered learning investments; they must be wisely spent and need to be widely shared". Italy has the biggest extent of solar electricity on the planet, in 2015 solar provided 7.8% of electricity request in Italy in 2016, after one more year of quick development, solar generated 1.3% of worldwide power.

c) **Geothermal energy:** High-Temperature Geothermal energy is from thermal energy produced and put away in the Earth. Thermal energy is the energy that decides the temperature of issue. Earth's geothermal energy begins from the first development of the planet and from radioactive decay of minerals (in as of now unsure yet perhaps generally equivalent (extents). The geothermal inclination, which is the distinction in temperature between the centre of the planet and its surface, drives persistent conduction of thermal energy as the warmth from the centre to the surface. The descriptive word geothermal begins from the Greek roots geo, which means earth, and canteen, which means heat.

The heat that is utilized for geothermal energy can be from profound inside the Earth, right sensible centre 4,000 miles (6,400 km) down. At the centre, temperatures may reach more than 9,000 °F (5,000 °C). Warmth conducts from the centre to encompassing rock.

Amazingly high temperature and weight cause some stone to liquefy, which is ordinarily known as magma. Magma changes over upward since it is lighter than the strong shake. This magma at that point warms shake and water in the hull, once in a while up to 700 °F (371 °C).

From hot springs, geothermal energy has been utilized for washing since Palaeolithic occasions and for space warming since antiquated Roman occasions, however, it is currently better known for power age.

Low-Temperature Geothermal refers to the use of the outer crust of the earth as a Thermal Battery to facilitate Renewable thermal energy for heating and cooling buildings, and other refrigeration and industrial uses. In this form of Geothermal, a Geothermal Heat Pump and Ground-coupled heat exchanger are used together to move heat energy into the earth (for cooling) and out of the earth (for heating) on a varying seasonal basis. Low-temperature Geothermal (generally referred to as "GHP") is an increasingly important renewable technology because it both reduces total annual energy loads associated with heating and cooling, and it also flattens the electric demand curve eliminating the extreme summer and winter peak electric supply requirements. Thus Low-Temperature Geothermal/GHP is becoming an increasing national priority with multiple tax credit support and focus as part of the ongoing movement toward Net Zero Energy. New York City has even just passed a law to require GHP anytime is shown to be economical with 20-year financing including the Socialized Cost of Carbon.

d) **Growth of renewables:** From the end of 2004, worldwide sustainable power source limit developed at rates of 10–60% yearly for some advancements. In 2015 worldwide interest in renewables rose 5% to $285.9 billion, breaking the past record of $278.5 billion of every 2011. 2015 was likewise the principal year that saw renewables, barring enormous hydro, represent most of all new power limit (134 GW, making up 53.6% of the aggregate). Of the renewables all out, wind represented 72 GW and solar photovoltaics 56 GW; both record-separating numbers and forcefully from 2014 figures (49 GW and 45 GW separately). In monetary terms, solar made up 56% of absolute new investment and wind represented 38% as shown in Figure 1.1.

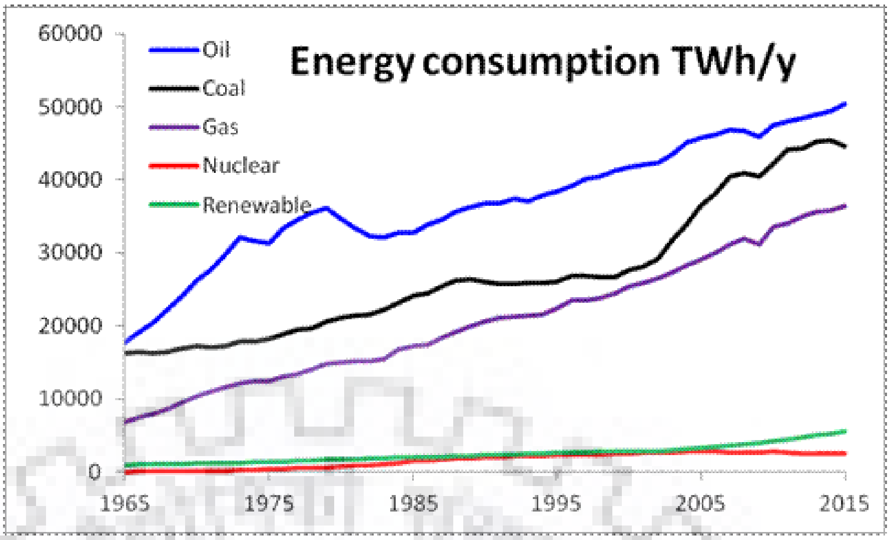


Figure 1.1 Growth of Renewable Energy Consumption [8]

Projections change. The EIA has anticipated that just about 66% of net augmentations to power capacity will originate from renewables by 2020 because of the joined policy advantages of neighbourhood contamination, decarbonisation, and energy expansion. A few examinations have set out guides to power 100% of the world's energy with wind, hydroelectric and solar continuously 2030.

As per a 2011 projection by the International Energy Agency, solar power generators may create the greater part of the world's power inside 50 years, diminishing the outflows of ozone depleting substances that hurt the earth. Cedric Hilbert, senior examiner in the renewable energy division at the IEA said: "Photovoltaic and solar-thermal plants may meet most of the world's demand for electricity by 2060 – and half of all energy needs – with wind, hydropower and biomass plants supplying much of the remaining generation". "Photovoltaic and concentrated solar power together can become the major source of electricity", Hilbert said.

In 2014 worldwide wind power capacity extended 16% to 369,553 MW. Yearly wind energy generation is likewise developing quickly and has stretched around 4% of overall power use, 11.4% in the EU, and it is generally utilized in Asia, and the United States. In 2015, overall introduced photovoltaics capacity expanded to 227 gigawatts (GW), adequate to supply 1 percent of worldwide power requests. Solar thermal energy stations work in the United States and Spain, and starting at 2016, the biggest of these is the 392 MW Ivanna Solar Electric Generating System in California. The world's biggest geothermal power establishment is The Geysers in California, with an evaluated capacity of 750 MW. Brazil has one of the biggest renewable energy programs on the planet, including the generation of ethanol fuel from sugar stick, and ethanol presently gives 18% of the nation's car fuel. Ethanol fuel is additionally broadly accessible in the United States.

1.3 Smart hydroelectric systems

Smart Hydro Power began in 2010 as a gathering of kinetic turbine developers. The experience on establishments and contact with the requirements and difficulties of remote or off-the-network areas conveyed us to the present position of offering items and bundles that pursue the standard of being basic, focused, and complete. These days, we run various decentralized electrification projects, offering a scope of renewable energy arrangements which are upheld by a wise electrical administration framework. New thoughts and motivated experts are constantly welcome here. We have confidence in the feasible improvement and empowering individuals to have the option to characterize their very own decisions and to shape their own lives.

1.4 Monitoring and Broadcast systems

To identify and prevent failures, it is advantageous to have a decent monitoring tool, that is the reason you need a monitoring system. Monitoring systems are in charge of controlling the technology utilized by an organization (hardware, operating systems, networks, and communications or applications) so as to dissect their activity and execution and to recognize and alarm about possible errors. A decent monitoring system can monitor devices, foundations, applications, infrastructure, and even business process.

1.4.1 Advantages of Monitoring and Broadcast systems

1. It improves the utilization of the equipment of the organization, through the control of its great activity. On the off chance that, for instance, a PC isn't working appropriately, the monitoring system will distinguish it, pull out of it and the choice, to fix or replace it, might be made.
2. It prevents incidents and when these incidents occur, they are recognized quicker, which spares time and money.
3. By maintaining a strategic distance from falls in the administration or limiting the season of goals of it, the picture of the organization will improve just as the client administration.
4. Less time is spent on controlling the correct working of the systems, for the most part on the grounds that the monitoring system deals with it.

5. This way, your certified staff will almost certainly dedicate more opportunity to different assignments, realizing that, if an issue emerges, you will get the relating cautions. This will likewise cause an expansion in your efficiency, which is most likely a significant motivation behind why you need a monitoring system.
6. Generate more revenue.

Challenges in Monitoring and Broadcast systems

1. Machines' actions in unpredictable situations.
2. Information security and privacy.
3. Machine interoperability.
4. Mean-reverting human behaviours.
5. Slow adoption of new technologies.

1.5 Internet of Things (IoT) Technology and Applications in Hydroelectric Power Plants

An IoT ecosystem comprises of web-enabled smart devices that utilization embedded processors, sensors and correspondence equipment to gather, send and follow up on information they obtain from their surroundings. IoT devices share the sensor information they gather by associating with an IoT portal or another edge device where information is either sent to the cloud to be broke down or analysed locally. Some of the time, these devices speak with other related devices and follow up on the data they get from each other. The devices do the greater part of the work without human intercession, despite the fact that individuals can collaborate with the devices, for example, to set them up, give them directions or access the information.

The availability, systems administration and correspondence conventions utilized with these web-enabled devices to a great extent rely upon the particular IoT applications deployed.

Hydroelectric energy has more than 1,200 GW of installed capacity worldwide, a number that according to projections – is expected to exceed 1,600 GW by 2025. Plants powered by water now supply 70% of renewable energy worldwide, guaranteeing clean energy to over a billion people in 152 countries. Numbers and trends reveal that hydroelectric offers an essential contribution to the decarbonisation of the electricity and energy sectors, and not just because of the mega dams expected in in China, India and Brazil or the great development potential in Africa. Mature markets, where this is no more room for large plant projects and common efforts are focused on efficiency and performance, are turning out to be an important laboratory for the experimentation of innovative solutions capable of channelling the opportunities offered by so-called smart technologies.

1.6 Different Micro Processors and Controllers for IoT

Difference between a Microcontroller and a Microprocessor: Microcontrollers likewise contrast from microprocessors. Microprocessors contain just a CPU, and consequently, require added peripherals to perform tasks. MCUs, then again, contain RAM, ROM, and similar peripherals, which allow them to perform (simple) tasks independently as show in Figure 1.2.

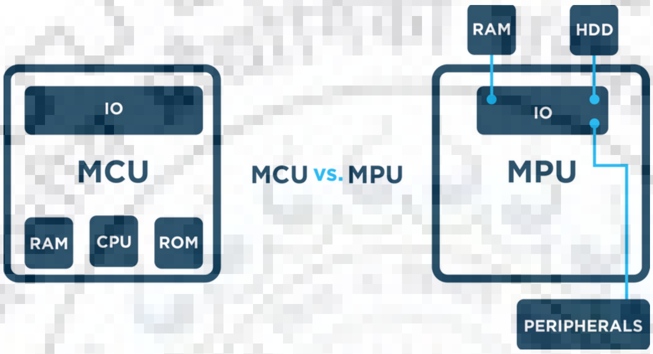


Figure 1.2 Difference between micro-controller and micro-processor [17]

List of Processors and controllers:

a) Arduino and PCDuino
b) BeagleBone Black
c) Raspberry pi

The Arduino, Raspberry Pi, BeagleBone and PCDuino may look very comparable for you, however, they are in reality altogether different devices. The Arduino is a microcontroller. A microcontroller is only one small piece of a PC. The Arduino can be programmed in C, however, can't run an operating system. Then again, the Raspberry Pi and PCDuino are PCs. Those devices can run an operating system alone.

a) **Arduino and PCDuino:** The Arduino is essentially ideal for electronics projects and prototyping. You can simple associate some LED's, transducers, motors into the board directly. It is easy to understand board, it is simple to program the Arduino, and you need their software (that can be download for free). Essentially with that product you can transfer your source code directly into your Arduino through USB.

The heart of the Arduino is ATMega3280P microcontroller. We use Arduino in this project work.

b) **BeagleBone:** It is like a Raspberry Pi yet its all the more powerful, in light of the TI SITARA AM335x, an application processor SoC containing an ARM Cortex-A8 centre. You have more pins to control.

c) **Raspberry pi:** In this project we utilize this processor, The Raspberry Pi is a finished minicomputer. It needs a working framework to work. All the Storage is given from a SD card. You can interface this to your system with an Ethernet Cable. The brain of the Pi is an ARM1176JZF-S 700 MHz it has designs it has a HDMI yield. You can connect a console and screen, load up Linux, and the less actually smart may do not understand how tiny the machine driving everything truly is. The Pi is an incredibly powerful platform in a very small package its credit card sized and perfect for embedded systems, or projects requiring more interactivity and processing power.

CHAPTER-2

LITERATURE REVIEW

A literature review has been conducted for knowing the importance of smart systems, different existing technologies, protocols used in different monitoring systems are studied as below.

2.1 Importance of Smart monitoring systems

M. Mustafa Music et al. [17] recommended that, as a result of the administrative necessities for the quality of electricity supply being forced in numerous nations, power quality was should be a significant part of smart distribution grids. In that unique situation, there was a requirement for the Integrated Power Quality Monitoring System, which would coordinate all the power quality data accessible from different systems of smart distribution matrix, for example, Power Quality Monitoring System, Automated Meter Reading/Advanced Metering Infrastructure, Supervisory Control and Data Acquisition, Electric Vehicle Management System etc.

Salvucci et al. [22] proposed that regarding the time before the most contemporary observing systems, the data base with programmed reports speaks to a major advance forward in the innovation of account of machine activity parameters. Just individual conditions of the intrigue must be recorded before and the acquired data were accordingly analysed. That diligent work was done as a back ground process on a server PC today. One need to tap the mouse a few time on the web application you will acquire pertinent data, for example, number of working hours of the generator, number of begins and quits, working routines of the machine, recurrence of appearance of the vibrations, etc. based on which the present state of the machine can be surveyed just as its remaining life time, timespan by the accompanying update or the need of a fix or refurbishment.

2.2 Research on smart systems in power generation and distribution

Handziski and Vlado [8] said for our existing society, electricity was extremely basic and to precisely maintain and expand this power distribution system, it was expected to understand and screen the system conduct. Self-configuration of sensor hub which are comparable to DHCP, should be possible. In this different sorts of flaw was happen in transmission system single line to ground line issue, line to ground deficiency, line to line issue, and balance three stage shortcoming. This four flaw permit to understand transmission line systems.

Fang X and Misra S [5] recommended that from smart network, electricity was circulated to our present society with the assistance of internet of things (loT), smart grid innovation is the significant specialized help for the age, transmission, substation, distribution. Smart grid depends on internet of things. IoT innovation is advantageous of smart network.

2.2.1 Applications of smart systems

Jain et al. [10] proposed a new SHEMS dependent on the IEEE802.15.4 and ZigBee and Develop Routine Protocol called "DMPR" (Disjoint Multi Path Routine) to upgrade the performance of ZigBee sensor arranges that provides intelligent service to consumers.

2.2.2 Reliability of smart systems

Ramakrishnan et al. [21] said that the reliability and prediction of the electronics disappointment rates are frequently identified with the thermal stress actuated amid different operating conditions at the board. To get data about the environmental conditions and to aid the prediction of an item's disappointment rate an observing system could be utilized. A temperature observing capacity executed in a FPGA utilizing VHDL. The proposed system is made out of various temperature sensors together with a non-volatile MRAM memory for long-term storage. The arrangement offers an adaptable system that could be scaled to a subjective number of sensors and be adjusted to any FPGA or processor innovation. To verify the concept, a climate chamber was used to simulate a number of different environmental settings followed by a reading of the data logged.

2.3 Intermittent Nature of Hydroelectric power plants

Anees and Ahmed Sharique [2] said that Photovoltaic (PV) generations where relied upon to have a significant job role rather than later. However, the power yield from the PV was random and intermittent in nature. In this manner, the PV generation presents numerous difficulties to the power system task. To evaluate impacts of the behaviour of PV on the power system operation, we have so far proposed effective utilization of PUMPED STORAGE HYDRO POWER PLANT (PSHPP). In the past research, it sets aside an excessive amount of calculation effort to get an ideal planning of PHSPP.

2.4 ANN used for prediction of power production in HPP

Kawadia et al. [13] said that in developing countries, the power production was properly less than the request of power or load, and sustaining a system stability of power production is a trouble quietly. Sometimes, there is a necessary development to the correct quantity of load demand to retain a system of power production steadily. Thus, Small Hydropower Plant (SHP) includes a Kaplan turbine was verified to explore its applicability. On applying on Artificial Neural Networks (ANNs) by approaching of Feed-Forward, Back-Propagation to make performance predictions of the hydropower plant at the Himreen lake dam-Diyala in terms of net turbine head, flow rate of water and power production that data gathered during a research over a 10-year period. The model studies the uncertainties of inputs and output operation and there's a designing to network structure and then trained by means of the entire of 3570 experimental and observed data. Furthermore, ANN offers an analysing and diagnosing instrument effectively to model performance of the nonlinear plant.

2.5 Various Monitoring and control Techniques used in different countries

Nema et al. [18] Mentioned that in Myanmar, stand-alone micro-hydro power system, water turbine will vary in speed due to the variation of consumer load. This speed variation will cause in fluctuation in both voltage and frequency output from a generator. To solve this problem, electronic load controllers were invented and used in micro-hydro power system. The objective of this paper is to monitor the electrical parameters such as voltage, current, power and frequency of electronic load controller (ELC) by using microcontroller and liquid crystal display (LCD). This explains how to monitor and sense the above parameters and isolate between the power line and microcontroller. The voltage is sensed by using the step down transformer and voltage divider circuit. The current was measured with the help of an ACS 712 current sensor. Frequency signal was obtained by using a frequency signal converter circuit. Microcontroller 16F887 and liquid crystal display are used as the main devices to monitor the above parameters according to the values obtaining from the sensing circuits.

Alves et al. [1] said that in Brazil the control, protection and supervisory systems of hydro power plants faced significant changes in recent years. With the evolution of digital processors and high capacity communication systems, the cluster of panels containing electromechanical and static devices were substituted by IEDs (Intelligent Electronic Devices). However, most part of digital systems currently in operation are composed by independent subsystems, with a limited level of communication between them. As result, each subsystem in a hydro power plant (SCADA, protection, voltage regulators, and speed governors, among others) interacts with the process entirely independently from the others. This situation leads to a multiplication of wired signals (currents, voltages, temperatures, equipment status and alarms, commands etc.) that are brought from the process up to the IED location, entailing considerably the installation complexity (amount of cables and panels, conduits, ducts, passing boxes) and the interface hardware needed. The final result is an increased overall cost of the project.

Xiaohui et al. [27] recommended that recently, the monitoring service for significant users in the distribution arrange is moderately lacking in China, which enormously impacts the effectiveness of power distribution. In light of the implanted MCU, remote sensor system and Ethernet correspondence innovation, a sort of keen checking system for distribution organize is displayed by the qualities of significant power users. The gathered data could be imparted to the electric power company through Ethernet, which understands the computerization and remote power data checking.

Wang et al. [26] referenced that the Broadband remote access system utilized for observing power transmission line. smart matrix innovation guarantee data trade and asset management rapidly and proficiently smart framework innovation will include different advances, including detecting, correspondence, organizing, security, data mining examination, control and management etc. half and half remote system is utilized for checking power transmission line. Here Wi-Fi and wi- max innovation is utilized for detected data transmission in transmission of data.

Sauter et al. [23] said that Smart grids vigorously rely upon correspondence so as to arrange the generation, distribution, and utilization of energy much more so whenever conveyed power plants dependent on sustainable power sources are considered. Given the assortment of correspondence partners, a heterogeneous system infrastructure comprising of IP-based and reasonable field-level systems is the most proper arrangement. A two-level infrastructure and conceivable field-level systems with specific consideration regarding metering and supervisory control and data acquisition applications. For the issue of system reconciliation, a mix of door and burrowing arrangements is proposed which permits a semi-straightforward start to finish association between application servers and field hubs. The achievability of the methodology and execution subtleties were examined at the case of power line correspondence and IP-based systems explored in the European research venture on constant vitality management through power lines and web. By and by, it is demonstrated that the correspondence engineering is sufficiently flexible to fill in as a nonexclusive answer for smart grids.

2.6 GPRS Based Remote monitoring system for MHPP

Lidula et al. [15] recommended that the displayed distributed system comprises of three microcontrollers and is proposed for the control and observing of the little hydroelectric power station. The little power station is situated in a remote region and works in self-governing mode. Particular programming is developed, acknowledging "Pinnacle POWER TRACKING" calculation for programmed control. Like clockwork, reports of estimated data are sent to the PC (base station) by means of GPRS/Internet. The utilized correspondence protocols are: TCP/IP; GPRS; SMS; ZigBee; USB.

2.7 PLC for sensor data controlling

Sung et al. [24] said that power line correspondence (PLC) is a proficient strategy in cost viewpoint since it was correspondence system utilizing power line without additional correspondence links. If there should be an occurrence of DC-PLC, it sends a correspondence sign of the high recurrence band to the observing system through the DC power line and was broadly utilized for DC framework or the dispersed generation system utilizing DC source. Particularly, the management system which assesses the presentation and productivity of PV module by estimating PV power through checking system dependent on the DC-PLC has been expanded. Here, apply the DC-PLC technique for minimal effort PV module checking system and establishes an estimating gadget and correspondence modem on a MCU. For the correspondence execution, we utilize the advanced channel and apply the multi-transporter correspondence. The proposed DC-PLC Modem was confirmed through reproduction and examination.

2.8 Web application based Remote monitoring system

Y. Ram Kumar and K Selvajyothi [28] depicts an Internet of things (IoT) based observing of distribution transformer to expand its life and execution. Here a technique has been proposed by transferring the continuous sensor data to database utilizing IoT idea and showing it in the online live gateway. In this manner empowering to break down the data and henceforth, to distinguish the deficiency. The design and execution of the plan are expounded. As the demand for power was expanding and transformers are the significant hardware at the distribution side, a checking system will help in limiting the power interrupts in our everyday life.

2.9 Gaps Identified

Based on the literature review it was found that there are number of methods available for monitoring and broadcasting parameters of hydroelectric power plant. Most of them are short ranged, local hosted systems and have its own disadvantages. Gaps identified are listed below.

1. Big data analysis and machine learning using existing monitoring system are complicated in existing monitoring systems, further it is difficult to develop effective forecasting system (used to predict future requirement of inventory) in existing available systems due to fixed storage memory space, historical data storing in single volume is difficult.
2. Existing methods are assembly of Sensor network, PLC and SCADA which are very costly for maintaining or to afford by micro hydro power plants.
3. In existing monitoring system, easy global access of power plant data are not available as they are locally hosted using fixed dedicated servers.
4. Some industries maintain more than one power plants in such cases, organising all these power plants data in one platform feature is not available in existing monitoring system.

2.10 Objectives

1. To Build compact embedded sensor network circuit board for signal conditioning and sensor network data flow connections into ADC before connecting to microprocessor, for monitoring parameters in micro hydro power plants (MHPP).
2. To Design and build monitoring panel using web-technologies and with high firewall security using AWS technology.
3. To Design and Build 3D printed IoT data transmitter device using GSM communication module in GPRS mode, for sending data to server from MHPP.
4. To Install IoT data transmitter in existing MHPP for enabling remote monitoring and forecasting system.

2.11 Methodology

1. The study has been carried out in four stages. Flow chart of this proposed study was shown in Figure 2.1.
    a. Stage one is to design a data flow architecture, schematics, and 3D printing the whole data transmitter device. This device uses GPRS mode to access internet connectivity.
    b. Second stage is to host monitoring panel in AWS and creating tables in RDS database to store the sensor data came from data transmitter.

c. Third stage is to decrypt the data followed by displaying values and time series plots in monitoring panel.
d. Final stage is to observe the trend of time series plots and forecasting data in monitoring panel. Additionally, a notification system also built in this study to prevent big damage and maintain good efficiency of power plant.

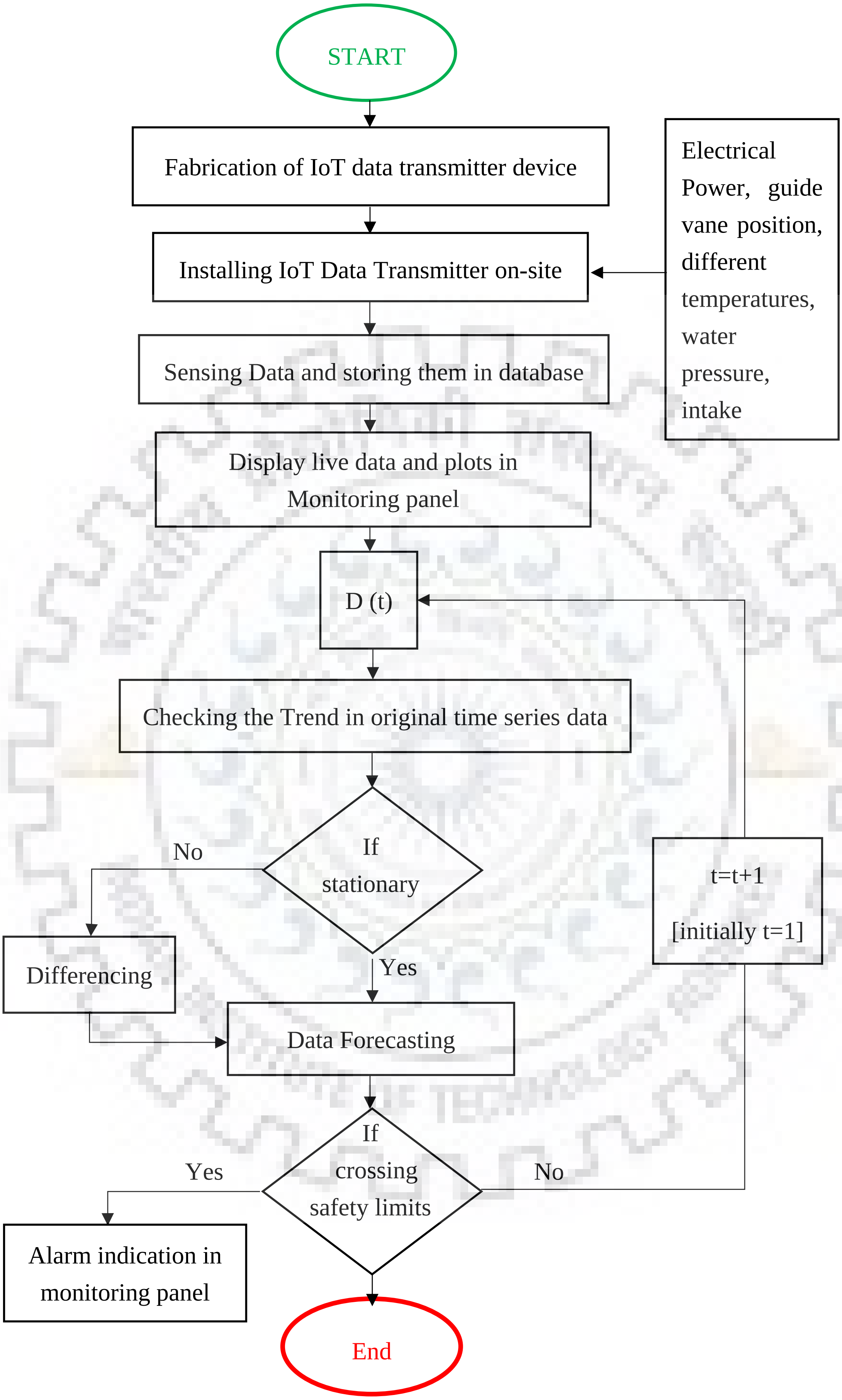


Figure 2.1: Flow chart of the proposed study.

CHAPTER-3

A Comparative Analysis of IoT with existing monitoring system

While designing and researching on IoT system and the feasibility in various power generation industrial companies, it is been observed that there is one common question that these industries repeatedly rise was, “Currently in power plants monitoring and forecasting is already featured using SCADA monitoring systems, them how it is going to change or improve by using IoT?”. So, here in this chapter a comparative analysis of IoT with existing monitoring system and how IoT is going to be next industrial revolution are explained by using these following technical categories as shown in table 3.1

Table 3.1: Different categories to do comparison with IoT over existing systems.

S.no	Industrial Features
1	Device interoperability
2	System ownership cost
3	Insights from data
4	Scalability
5	Standards and protocols

Before doing comparison, it is important to understand SCADA is widely used for monitoring and control different applications in industries, so the basic functions of SCADA is explained below.

SCADA stands for supervisory control and data acquisition, this is a method which uses computers graphical user interface programmable logic controllers discrete PID controllers and networked data communication to connect power plant data parameters. The key feature of SCADA is its capability to do supervisory operation on multiple devices working at same time. Figure 3.1 shows SCADA process in different functional levels [18].

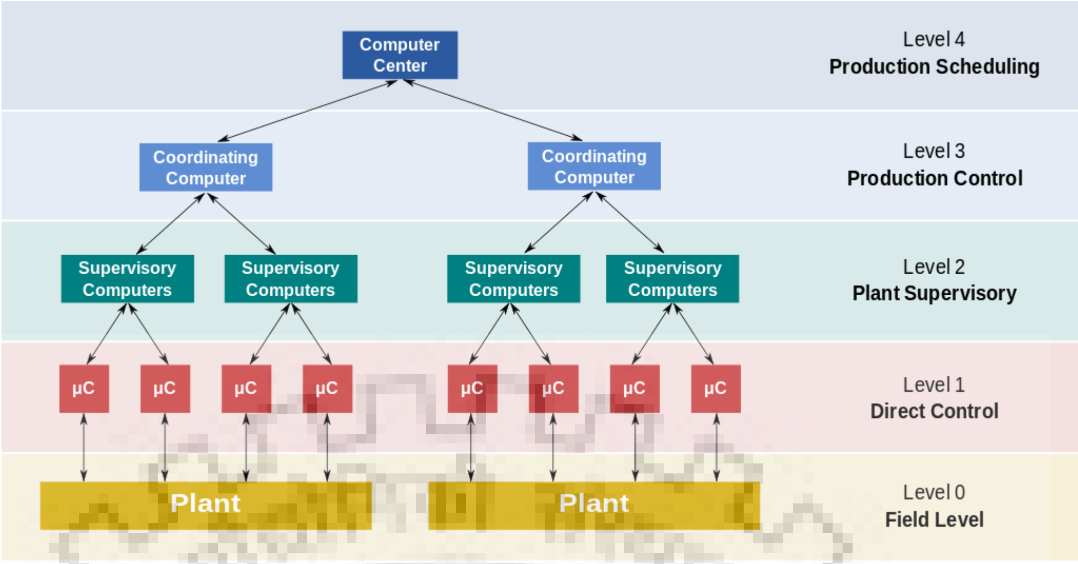


Figure 3.1 SCADA functional levels of monitoring and control.

1. Level 0: here all on site parameters like control panel temperature, guide vane position, bearing temperature, power generated after generator, rate of discharge sensing transducers are involved.
2. Level 1: μC stands for micro controller, so here all I/O device modules like micro controllers, ADC/DAC modules are involved.
3. Level 2: here supervisory computers present which gather data information from previous level and display them in monitoring panel.
4. Level 3: this is production control level which will not directly control the process, but this is targeting the monitoring and target the data collected by sensors.
5. Level 4: is energy production scheduling zone.

The three main features of SCADA system are:

1. Supervisory control
2. Reporting data periodically through Simple Mail Transfer Protocol (SMTP)
3. Alarms when any immediate breakdown or abnormal functioning of any parameter and Events

3.1 SCADA monitoring system and IoT monitoring system

A comparative analysis between SCADA monitoring system with IoT monitoring system are listed based on 5 important parameters which are shown in table 3.2

Table 3.2: SCADA monitoring system and IoT based monitoring system differences

S no	Parameter	SCADA monitoring system	IoT monitoring system
1	Device interoperability	Integration of devices that are created by different manufacturers is not simple. Sometimes, devices from same manufacturer is also complicated, it is a very big challenge to use them interchangeably if those device version varies. In this lacking much needed interoperability, which is mandatory for developing programmability for devices and transducers. In addition, they require a horizontal platform that can do operation across wide range of devices without depending on vendors or versions.	The main target of this method is to enable communication across wide range of devices regardless of model number or versions. Internet of Things uses Message Queuing Telemetry Transfer (MQTT) protocol to make communication across different devices throughout the entire system.
2	System ownership cost	Industries can store data only up to certain extent, after which the data will be overwritten by the most recent received data. This means that in order to store more date sensed from transducers industries must invest more for extra servers with high capacities. Also, in the case of software licensing or getting additional features, this system owners should buy	This system can considerably reduce the hardware and system ownership cost for industrial owners, while also avoiding the extra software licensing and upgradation by using relational cloud services

		separate licenses for additional servers and have to pay periodically for system upgradation	
3	Insights from data	Industrial owners have limitations in analysing historic data and the make sense out of it. It is not focusing on collecting or analysing the full historic data as they are only doing daily basis data analysis. So, the data from this system does not provide any help for decision making on working sensor parameters, and making difficult to take decisions when to do maintenance and services	This system has very big advantage while speaking about data storage than SCADA. This system sense and store the data from every industrial process and then uses big data analytics and machine learning algorithms to predict the efficiency and potential outcomes. With easy understandable insights from this system, industrial officers gain ability to minimize sudden equipment failures, predict maintenance needs, and get higher outcomes from their current assets
4	Scalability	Sometimes, this system may leave out very important information from wide range of devices, as system is not storing or analysing data. The main reason behind is not connecting devices system performance and data security, as the number of users increases, the bandwidth requirement to function also parallel increases, finally this results in increasing the overheard cost for the company. Mainly, using this systems it is difficult to get perceptive reports from centralized system for devices that are located in remote locations.	Here all the sensors data are stored in relational database services cloud and can be easily accessed from anywhere. Also, the advantage of connecting extra devices provides access to all the data that are playing important role in making decisions in industries.
5	Standards and Protocols	OPC is used in this system for data collection, this standard proved its reliability in most of the	This brings standardization of networks concept in networks, sensors and data collecting to data base tables. This system standards

		cases. However, this depends on Distributed Component Object Model (DCOM) technology that only runs in Windows operating systems, which is a very big drawback.	like OPC are used to enable real time device communication within power plant floors which uses different devices and transducers from different manufacturers. In IoT system data security is also maintained using protocols such as SSL and HTTPS

The IoT is an assembly of various modern technologies that are working together in harmony to form a network of devices that are always constantly connected.

The impact of IoT will definitely bring to the industry can be referred as the next big industrial revolution. It is been observed that in every stage of a industrial revolutions are built by some innovating and improving from the previous stage technologies and methods. So, IoT is definitely an innovative concept after SCADA monitoring system.

For the people who are currently using SCADA monitoring system they can integrate this IoT data transmitter devices to their SCADA monitoring system in data acquisition stage and expand the features of their monitoring and control system.

CHAPTER-4
DEVELOPMENT OF IoT DATA TRANSMITTER DEVICE

IoT data transmitter is an electronics system which does data handling from different types of transducers, data conversion if required and finally encrypting and sending this digital format data to database. Data travel path from transducer to server was shown in Figure 4.1

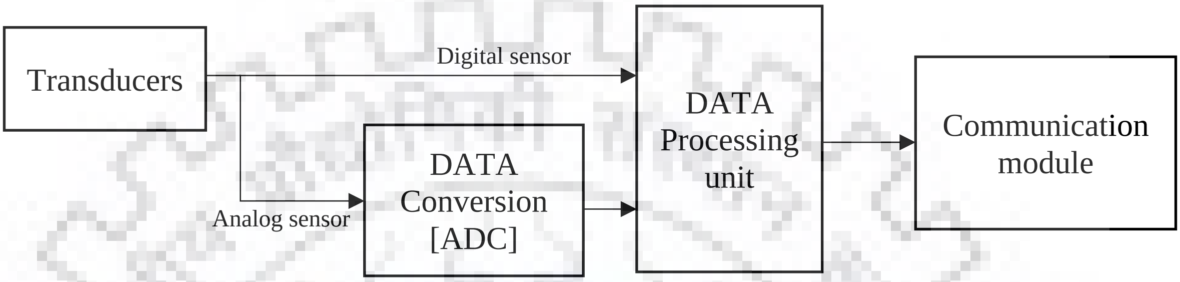


Figure 4.1 Block Diagram view of IoT Data Transmitter

In this chapter, two data transmitters are designed and developed, in Table 4.1 processor and communication method used in these two data transmitters are shown.

Table 4.1: processor and communication module used in data transmitter

Method 1	Micro processor	Raspberry pi
	Communication mode	Ethernet connectivity
Method 2	Micro processor	Arduino UNO
	Communication mode	GSM GPRS mode

4.1 Method 1: Data transmitter using Raspberry pi as Micro processor

Sensors-based systems are nowadays an extended technology for many markets due to their great potential in the collection of data from the MHPP and the processing of such data for different purposes. The requirements that are necessary for many such applications (real-time processing, low-power consumption, reduced size, reliability, and data security) means that research on advanced architectures of Microprocessors and System-on-Chips (SoC) is needed to design. Here we choose Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4GHz, and Raspberry Pi B as micro-processor to process the sensor data and send them to remote data-base. The data transmission technique is shown in Figure 4.2.

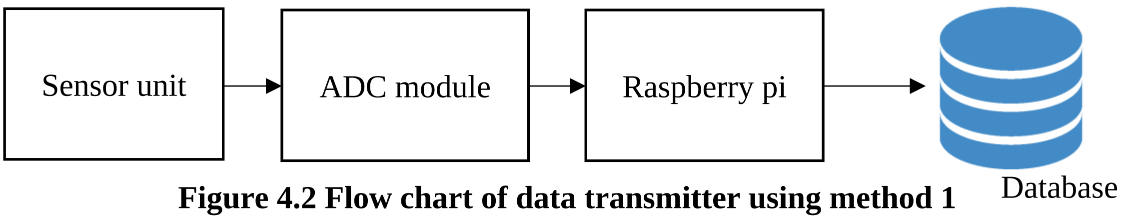


Figure 4.2 Flow chart of data transmitter using method 1

The following list of process are followed for building this data transmitter.

a. ADC module design

b. Sensor data transmission and server side algorithms

4.1.1 ADC module design

When an analogue sensors output is converted to a digital output with an analogue to digital converter (ADC), conversion resolution, conversion accuracy, conversion speed or bandwidth, inherent system noise levels, and power consumption are all ADC come to picture. Errors due to temperature, supply voltage, linearity, quantizing, and other factors may reduce the accuracy of an ADC by several bits. 7 bit flash ADC simulation schematic in Figure 4.3

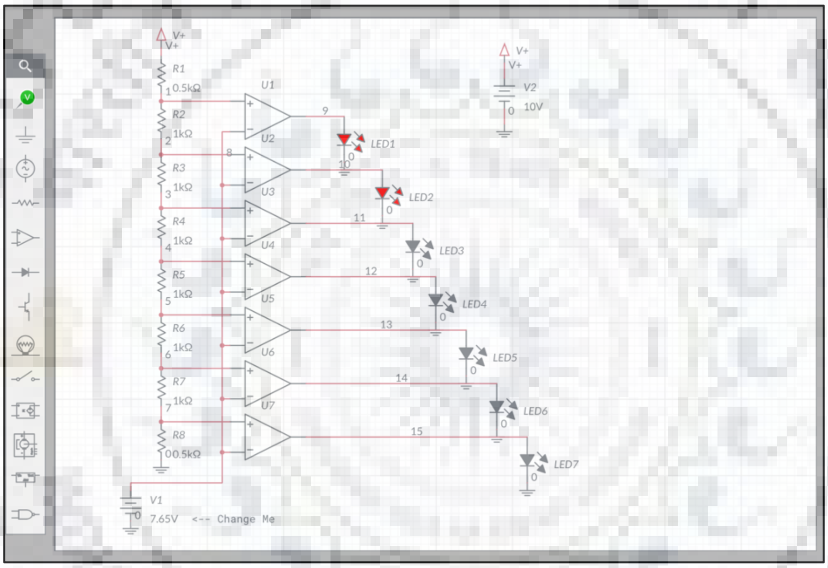


Figure 4.3 7-Bit Flash ADC Schematic in MATLAB

Calculations that can be made to determine the adequacy of the ADC for a specific application.

1. The resolution is the least significant bit (LSB) % full scale/100.
2. The quantization error (as a % of full-scale range) is [±1/2 * 1/ (2n– 1) * 100] which is also ±1/2 LSB.

The theoretical root mean square (RMS) signal-to noise ratio (SNR) for an N-bit ADC is shown in equation 4.1.

$$SNR = 6.02 * N * 1.76dB \quad (4.1)$$

Where:

N = number of bits

Sensors with millivolt-level outputs are well below the values required for use in a system, so amplification is necessary. This amplification plays an important role in the converted digital value. An amplified pressure signal supplied to a 7-bit ADC provides an example of the combined capability of the amplifier and the ADC. The A/D conversion is related to the pressure input as shown in equation 4.2 which is for 127 digital count as maximum limit.

$$count = [V_{FS} - V_{offset}] * 127/[V_{RH} - V_{RL}] \quad (4.2)$$

Where:

$V_{FS} - V_{offset}$ is sensor full scale span voltage

$V_{RH} - V_{RL}$ is 5V should be the same as the MCU.

A sensor with a 0.25 to 4.75V output, the maximum number of counts available at the output register will be: *count* (full scale) = 127.

Therefore, a full-scale pressure of 15 psi with 5.0V supply results in a system resolution of: 15 psi/127= 0.118 psi/*count*.

As 4 sensors so are required to be used, more than one ADC channel is required [20]. So, an 8-channel 24-bit ADC module was built, since 24-bits are there the sensing accuracy and resolution is very high as every minute change can be sensed. this system sends data to MCU using SPI protocol need to be design an on-board clock to ADC module so a built in clock of 7.68 MHz clock to this ADC module is included as shown in Figure 4.4.

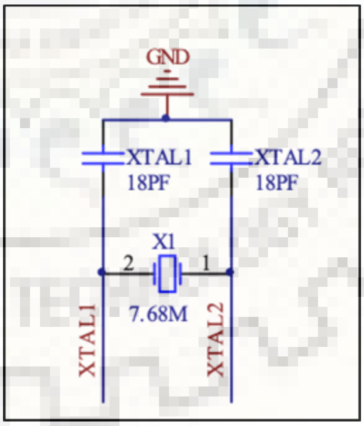


Figure 4.4 On-board clock schematic crystal oscillator 7.68MHz

ADS1256 IC is used as main ADC processor in this processing unit. It has 24 bits input extremely low-noise, 24-bit analogue-to-digital (A/D) converters. They provide complete high-resolution measurement solutions for the most demanding applications. Electronic specification is as shown in Table 4.2.

Table 4.2 ADS1256 IC Parameters

S no	IC Parameter	Values
1	Resolution (Bits)	24
2	Sample Rate (Max) (kSPS)	30
3	Number of input channels	8
4	Interface	SPI
5	Operating temperature range (C)	-40 to 85
6	Power consumption (mW)	36
7	Multi-channel configuration	Multiplexed
8	Reference mode	Ext
9	Input range (Max) (V)	5.25
10	Input range (Min) (V)	0
11	Analog voltage AV (Min) (V)	4.75
12	Analog voltage AV (Max) (V)	5.25

On-board PCB schematic for signal conditioning and ADC1256 IC design is shown in Figure 4.5.

In some cases, we need to change the state from 0 to 1 or from 1 to 0. In either case, we need to hold the digital pin either 0 and then change the state to 1 or we need to hold it 0 and then change to 1. In both cases, we need to make the digital pin either 'High' or 'Low' but it cannot be left floating. A Pull-up resistor is used to make the default state of the digital pin as High or to the logic level and a Pull-Down resistor does exactly opposite, it makes the default state of the digital pin as Low for this purpose pull up resistors are designed for ADC module as shown in Figure 4.5.

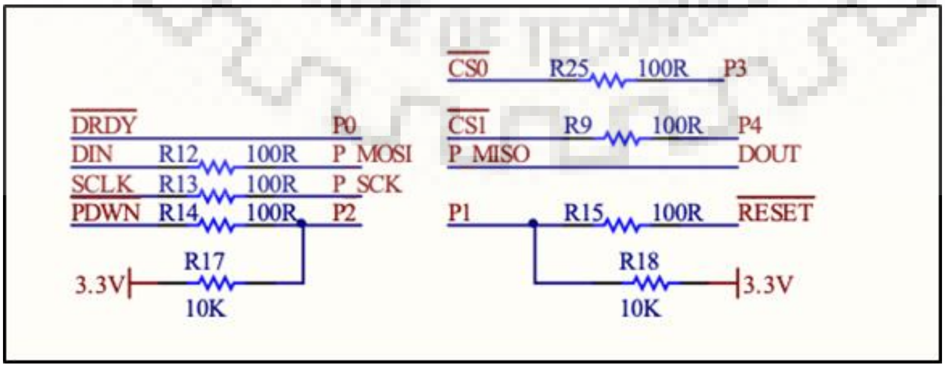


Figure 4.5 Pull up resistors in on-board schematic

For Analog 0(CS0) and Analog 1(CS1) input pins with multiple input and single out (MISO) feature for Analog 1 pin as shown in Figure 4.6.

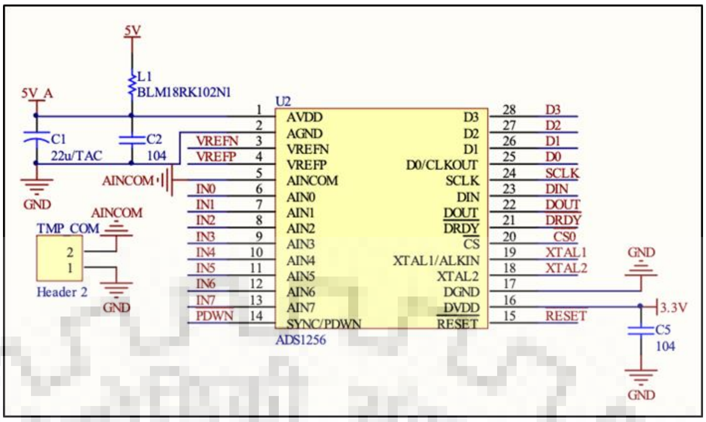


Figure 4.6 ADS1256 IC pin assigning to peripherals

IC peripherals are clock, Analog pin 0 to 7, clear signals, filters, we installed a potentiometer and LDR for module checking purpose before using for main purpose. Reference voltage for activating these peripherals, 5V reference voltage was designed to this on-board PCB design. All these peripherals are shown in Figure 4.7 and Figure 4.8

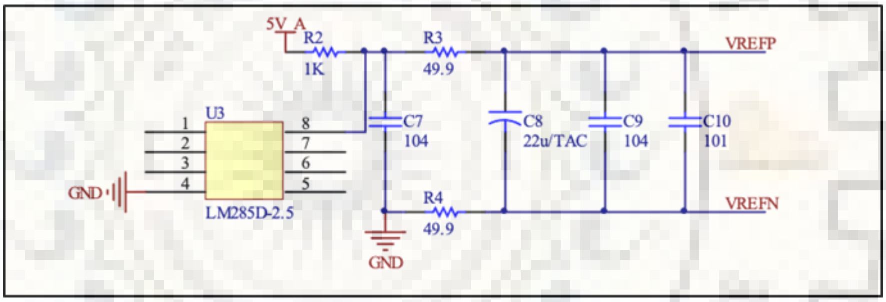


Figure 4.7 Generating reference voltage for ADC

An RC filter is designed to filter out higher frequencies and surge spikes during working conditions because these spike voltages will destroy MCU and analog sensor data pin connection I/O ports are designed and named as AIN0 to AIN7 are designed Filter design in MATLAB is shown in Figure 4.9.

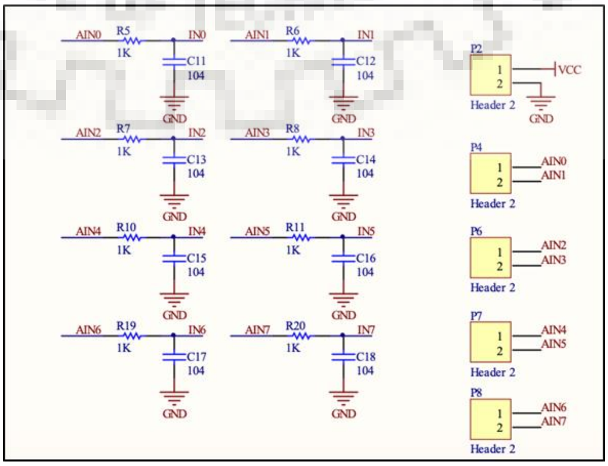


Figure 4.8 RC-Filter for each ADC in port

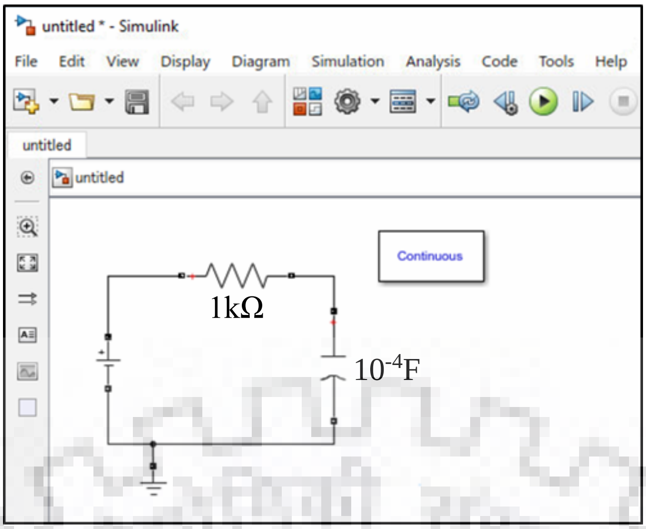


Figure 4.9 RC-Filter low-pass with R=1KΩ and C=10^{-4}F using MATLAB

Filter transfer function:

$$G(s) = \frac{10}{s + 10} \tag{4.3}$$

Rise time:

Time taken to reach 90% of original data when unit step U(s) is given as input gives H(s)

$$H(s) = G(s)U(s) \tag{4.4}$$

$$H(s) = \left[\frac{10}{s + 10}\right]\left[\frac{1}{s}\right] \tag{4.5}$$

$$H(s) = \frac{1}{s^2 + 10S} \tag{4.6}$$

Comparing this with standard second-order unit response shown in equation 4.7

$$\frac{\omega_n^2}{S^2 + 2\delta\omega_n S + \omega_n^2} \tag{4.7}$$

Then rise is as shown in equation 4.8

$$tr = \frac{\pi - tan^{-1}\frac{\sqrt{1-\delta^2}}{\delta}}{\omega_n\sqrt{1-\delta^2}} \tag{4.8}$$

Where:

Tr = rise time

ω_n = natural frequency

δ =damping ratio

Tr = 0.2302585092994 sec, δ=5 and cut-off frequency is fc =1.591 Hz

Bode plot is shown in Figure 4.10 From this plot as phase margin and gain margin are within 0 to -180^{o} this system is stable.

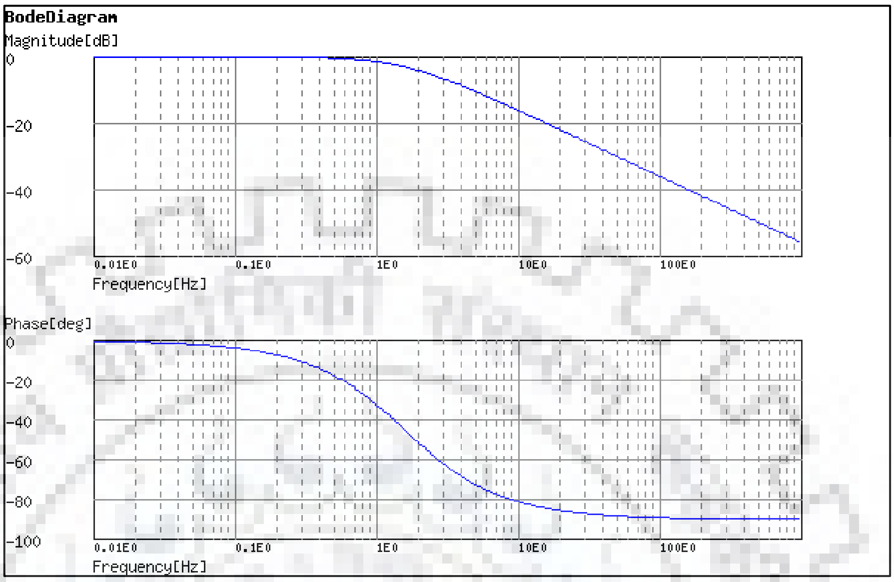


Figure 4.10 Bode plot of the designed RC-Filter

The following are the list of on-board features implemented in this ADC module. Their explanations are dissipated below.

a. On-board indicators

b. On-board LDR

c. On-board potentiometer

a. On-board indicators: LED PWR1 is for indicating 3.3V ref voltage is chosen, LED A1 is for indicating usage of on-board LDR, and LED A2 is for indicating on-board potentiometer as shown in Figure 4.11.

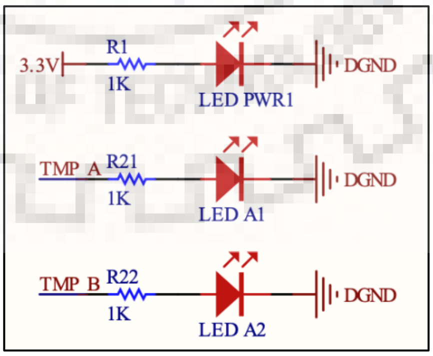


Figure 4.11 ADS1256 on-board indicators

b. On-board LDR (light depending resistor): Schematic for installing LDR photocell in ADC module is shown in Figure 4.12. For safety operation 10kΩ of pull-up resistor is connected between VCC and LDR input pin i.e., collector port. And configured to Analog 1 in ADC module.

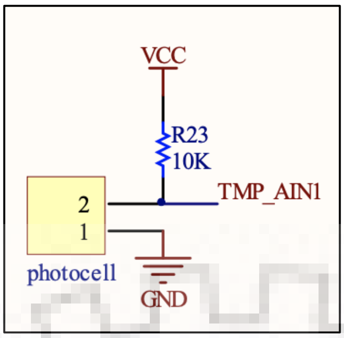


Figure 4.12 LDR photocell in ADC module

c. **On-board potentiometer:** Schematic for installing Potentiometer in ADC module which is used to test the ADC module working is shown in Figure 4.13. And configured to Analog 0 port in ADC module.

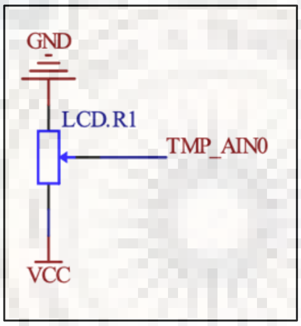


Figure 4.13 Schematic for installing Potentiometer in ADC module

Finally list of things used to build ADC module was printed and soldered all components. Module is shown in Figure 4.14.

List of components used to build this 24 bit ADC module:

1. AD/DA input/output: screw terminals
2. AD input: pin headers, the pinout is compatible with sensor interface standard, easy to connect various analogue sensor modules
3. 7.68MHz crystal
4. LM285-2.5: provides reference voltage for the ADC chip
5. Photo resistor
6. LED output indicator
7. 10K potentiometer
8. Power indicator
9. ADS1256: 24bit high-precision ADC, 8ch (4ch differential input)

10. ADC testing jumper
11. Power selection jumper

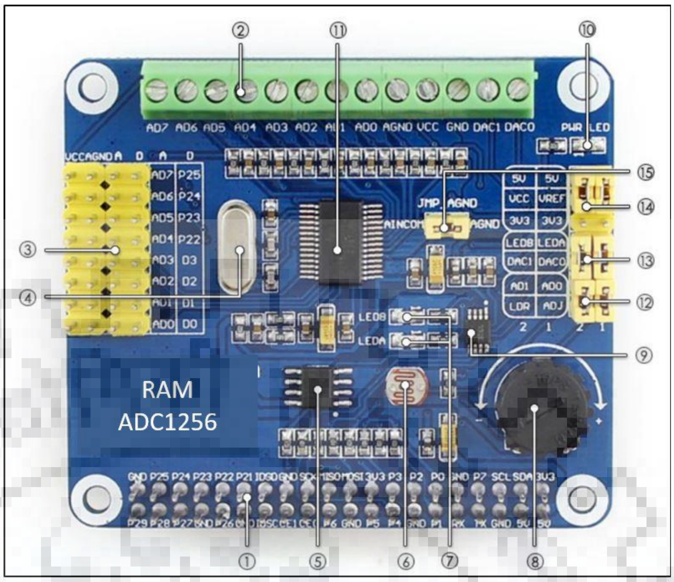


Figure 4.14 ADC1256 embedded ADC 24-bit module

Analog I/O ports are not available in Raspberry pi microprocessor. So, an ADC module is designed such that it can be mounted over Raspberry pi processor module which look like a multilayer PCB. ADC module GPIO pin header and mounting both Raspberry pi and ADC module are shown in Figure 4.15.

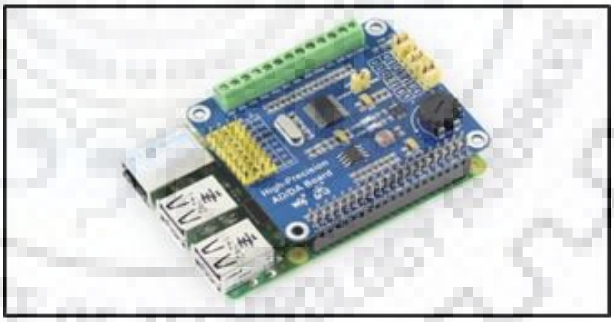


Figure 4.15 ADC module mounted on Raspberry pi processor

4.1.2 Sensor data transmission and server side algorithm

Once the data reach MCU i.e., micro controller unit data is controlled using C++ programming in micro-processor [23] [24]. We use MySQL and mysqlclientlib to connect C++ with remote database. We downscale micro-processor system clock and use it for SPI protocol to assign the data flow rate into database from MCU using the function shown below.

$$int\ bsp\_DelayUS(int);$$

Where: this function accepts only integer value and returns integer dimension is micro second.

List of header files we used for pre-processing this C++ code file MCU_to_Database_Ram.C is shown in Figure 4.16.

1. Bcm2835.h
2. Stdio.h
3. Unistd.h
4. String.h
5. Math.h
6. Errno.h
7. Stdlib.h
8. Mysql/mysql.h

Figure 4.16 List of pre-processing headers and initialisations in C++ programming

From figure we shown that we use bcm2835.h header file to interconnect ADC module and MCU, math.h for all mathematical operations and mysql.h to have connection between MCU and MySQL Database.

Once pre-processing is done we connect MCU to MySQL database with following flow chart in algorithm.

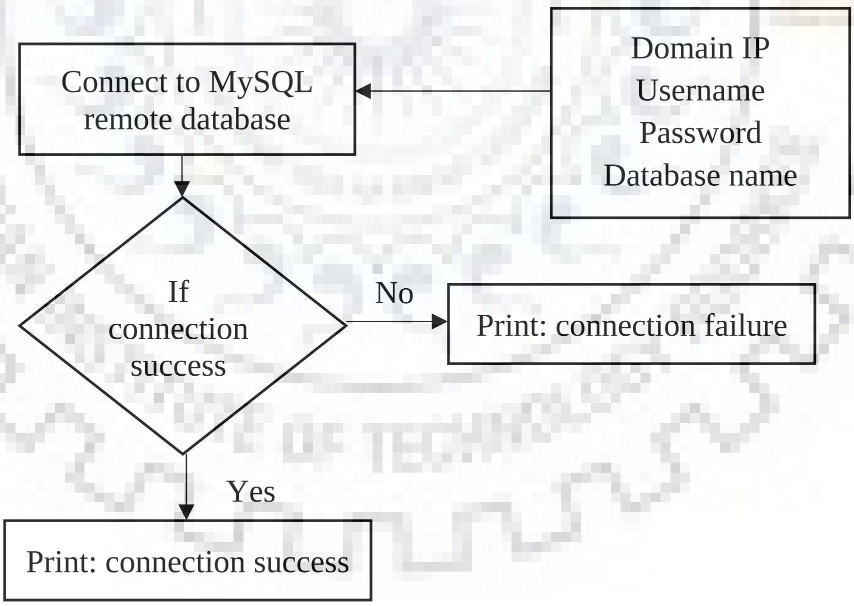


Figure 4.17 Connecting MCU and MySQL database

As shown in Figure 4.17 to connect MCU and MySQL database we first initiate MySQL libraries by calling below explicit function. All the variables are type casted as character data type as MySQL functions can only read characters.

$$mysql\_init();$$

Once initialization of MySQL is done properly we try to connect MCU with MySQL remote database using below function.

$$bool\ mysql\_real\_connect(char, char, char, char, char, int, bool, int);$$

From above function it is clear that it returns "NULL" if connection is unsuccessful or failure.

Finally reading data from ADC module to MCU through SPI protocol, we defined 0.1sec for SPI so, for ever 0.1sec we get fresh data from ADC unit. For achieving this we need to initiate ADC unit in C++ algorithm as shown in Appendix C. flow chart representation of the algorithm function is shown in Figure 4.18.

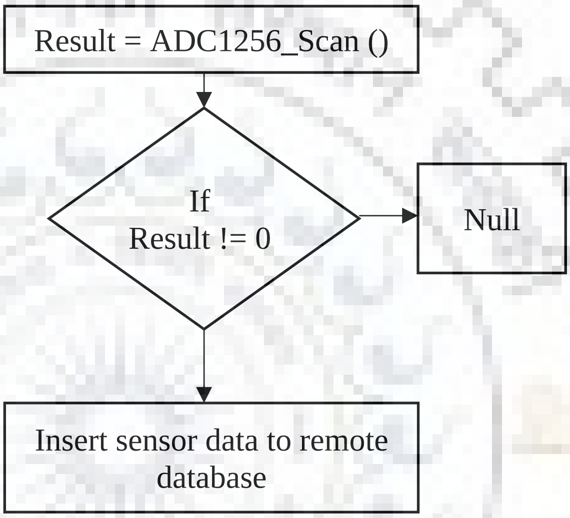


Figure 4.18 Connecting ADC unit with MCU

We use "ADS1256_CfgADC ()" function to initiate ADS1256 IC in ADC module and run a while loop forever using "while (1)" function. Next for receiving data to algorithm we use "ADS1256_Scan()" and "ADS1256_GetAdc()" functions according to algorithm developed for every 0.1 sec we are storing data into MySQL database one after one channel, and inserting into remote database using MySQL queries shown in Figure 4.18.

Once data inserting into database is completed we will run SPI clock, ADC module SPI end function and disconnect ADC module using following functions as shown in Figure 4.19.

$$bsp\_DelayUS(int);$$

$$bcm2835\_spi\_end();$$

$$bcm2835\_close();$$

Figure 4.19 Running SPI clock function

Now, data transmitter was successfully configured ADC module ADS1256 IC, MCU and MySQL database in C++ code and developed algorithm to insert values into remote database.

Complete algorithm of this method is shown in Appendix C. once the data transmission is done next is to decrypt the data from database and display them in monitoring panel. Displaying values in monitoring panel, plotting data, and forecasting data are discussed in chapter 5.

4.1.3 Bill of materials for data transmitter using Raspberry pi as Microprocessor.

List of modules used and their cost with description are listed in Table 4.3. From this cost analysis total cost of device to build was ₹8071.0 and server maintenance cost was ₹2600.0 per year.

Table 4.3: IoT data transmitter device using Raspberry pi as Processor bill of materials.

S.no	Quantity	Description	Component name	Cost
1.	1	Processing unit	Raspberry pi	₹3500.0
2.	1	Communication unit	On-board Ethernet	-
3.	1	Cooling system	CPU Cooling Fan	₹299.00
4.	1	3D printing material	White PLA 332 grams	₹532.00
5.	1	On board indicators and I/O connectors	5mm Ultra Bright LED 6 pin male connector	₹840.00
6.	1	Power supply module	220V to 12V /5V Isolated Switching Power Supply Module	₹600.00
7	1	Data convertor	ADC module	₹2300.0
	IoT data transmitter using Raspberry pi as Processor device cost			₹8071.0
7	1	AWS RDS Server service cost* (per year)	Amazon Web Services	₹2600.0*
	Yearly payment			**₹2600.0***

4.2 Method 2: Data transmitter using Arduino as processing unit

Arduino is a platform on which mounting sensors, fetch the sensors data from it are done. But if to use Arduino as an IoT device, network connectivity must be assembled with Arduino in order to get connected to cloud and storing sensor data in remote database using POST or GET methods via HTTP protocols. Data transmitter device using Arduino as a MCU flow chart is shown in Figure 4.20

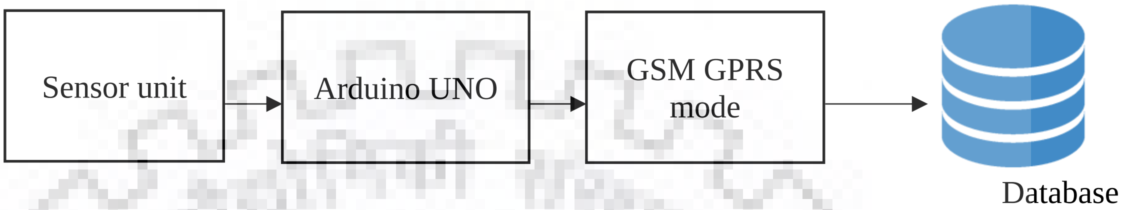


Figure 4.20 flow chart for Arduino MCU based data transmitter

4.2.1 Communication module

GSM/GPRS Modem-RS232 is built with Dual-Band GSM/GPRS engine- SIM900A works on frequencies 900/1800 MHz the Modem is coming with RS232 interface, which allows you connect PC as well as a microcontroller with RS232 Chip (MAX232). The baud rate is configurable from 9600-115200 through AT command. The GSM/GPRS Modem is having internal TCP/IP stack to enable you to connect with internet via GPRS. It is suitable for SMS, Voice as well as DATA transfer application in M2M interface. The on-board Regulated Power supply allows you to connect wide range unregulated power supply.

Communication unit initial settings and testing of its all functionality, algorithm is as follows. Arduino integrated development environment platform is used here for testing sending instruction sets into GSM module and checking its functions. Here we list all commands used to check the communication module shown in Table 4.4.

Table 4.4: Initial check AT commands for SIM900a module.

S.no	Command	Functionality
1	AT	To wake up GSM
2	AT+CMGF=1	Change to SMS Sending Mode
3	AT+CMGS="MOBILE NO." <MESSAGE {CTRL+Z}	To check SMS sending
4	ATD9876543210;	Check a voice call feature
5	AT+IPR=?	To see baud rate pre-set value

6	AT+IPR=0	To set module to auto-bauding mode
7	AT+COPS? +COPS: 0,0,"AirTel"	Operator selection

Once all the basic functionalities are confirmed working properly, GPRS settings are made to this module to get internet access with which this communication module will be transferring data to database. These GPRS AT commands are listed in Table 4.5.

Table 4.5: AT commands to check GPRS connectivity.

S.no	Command	Functionality
1	AT+CGATT	ATTACH/DETACH FROM GPRS SERVICE
2	AT+CGDCONT	DEFINE PDP CONTEXT
3	AT+CGDATA	ENTER DATA STATE
4	AT+CGPADDR	SHOW PDP ADDRESS
5	AT+CGCLASS	GPRS MOBILE STATION CLASS
6	AT+CGREG	NETWORK REGISTRATION STATUS
7	AT+CGCOUNT	GPRS PACKET COUNTERS

Once, internet connectivity settings are made in dual band GSM/GPRS engine, data transmitter main schematics are designed using MULTISIM software to interconnect this GSM/GPRS engine with Arduino MCU and facilitate I/O ports to connect sensor data out pins to this IoT data transmitter device are shown in Appendix E. Additionally, few on-board indicators are planned to indicate the working status of data transmitter as shown in Figure 4.21. Functionality of these indicators are as mentioned in Table 4.6.

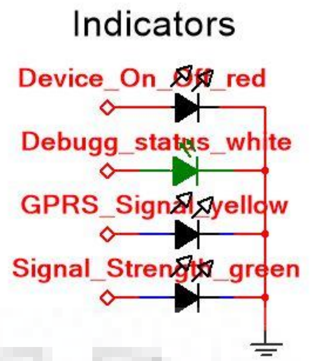


Figure 4.21 On-board indicators in Arduino based IoT data transmitter device

Table 4.6: On board indications and their functions

LED Colour	Functionality	Action
Red	Device ON or OFF state	On state continuously
White	Debugging mode indicator	While flash light for 0.3sec
Yellow	GPRS signal strength	Blink faster (0.5 sec): Poor Blink slower (2 sec): Good
Green	GSM network strength	Brighter: Good Lighter: poor

4.2.2 Power supply unit design

Power supply is like main fuel for the whole system, this unit is responsible to convert the available 220v 50Hz AC power supply to desired rated DC output.

Most of the electronics system demands either 12V 1A DC power supply or 5V 500mA DC power supply to switch their functionalities, here we design and build 5V/12V DC isolated power supply module using industrial graded step down transformer, diodes, RLC loads, capacitors, i/o connections and led indicators.

Full schematic and simulation of this module is shown in Figure 4.22 and Figure 4.23.

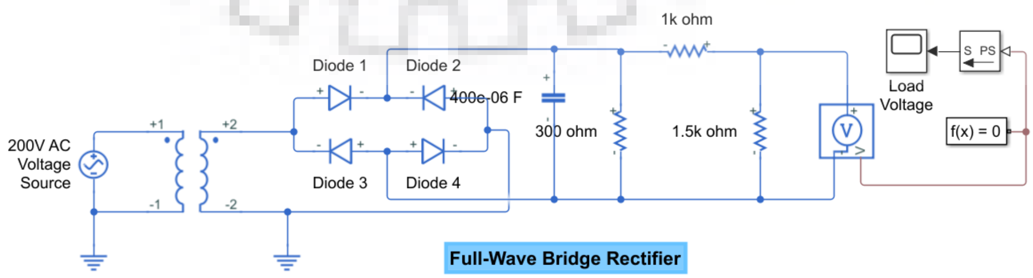


Figure 4.22 Full wave bridge rectifier schematic using MATLAB

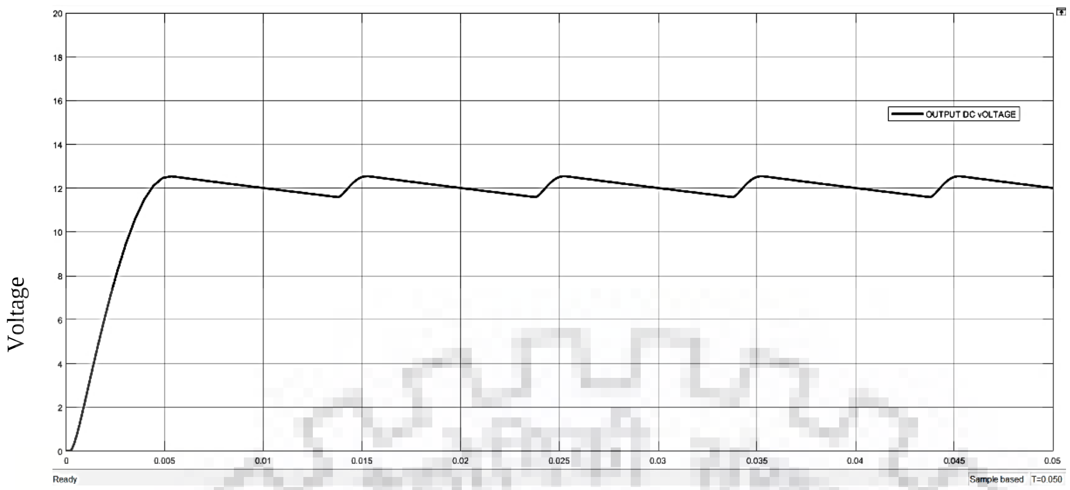


Figure 4.23 12V DC from Full wave bridge rectifier using MATLAB

Ripple voltage calculation for this power supply unit has been shown in Figure 4.24

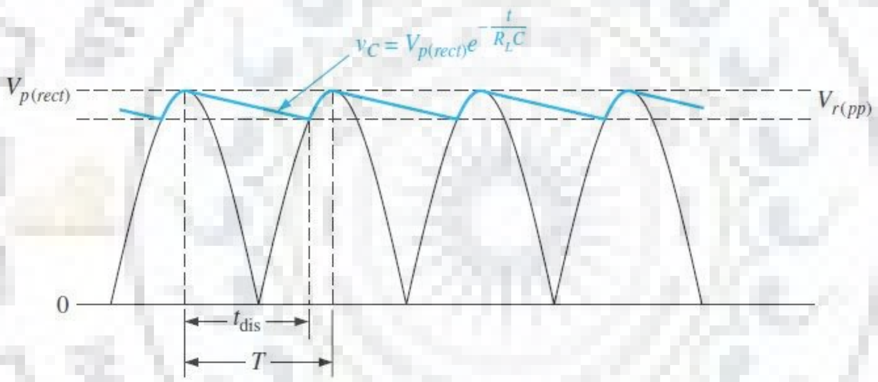


Figure 4.24 Ripple voltage calculation from FWR.

Where,

$V_{r\,(pp)}$ is the peak-to-peak ripple voltage.

V_{DC} is the dc (average) value of the filter's output voltage.

$V_{p\,(rect)}$ is the unfiltered peak rectified voltage.

Voltage after smoothing capacitor 400μF is shown in equation 4.1

$$V_c = V_{p\,(rect)} e^{\frac{-t}{R_L C}} \qquad (4.1)$$

$T_{dis} \approx T$ when V_C reaches minimum value.

$$V_{c\,(min)} = V_{p\,(rect)} e^{\frac{-T}{R_L C}} \qquad (4.2)$$

Since

RC>>T, T/R_LC becomes much less than 1 and e^{-T/R_LC} approaches 1 and can be expressed as

$$e^{\frac{-T}{R_L C}} \approx 1 - \left(\frac{T}{R_L C}\right) \tag{4.3}$$

Therefore,

$$V_{c\,(min)} = V_{p\,(rect)}[1 - \left(\frac{T}{R_L C}\right)] \tag{4.4}$$

$$V_{r(pp)} = V_{p(rect)} - V_{C(min)} \tag{4.5}$$

$$V_{r\,(pp)} = \left(\frac{1}{f R_L C}\right) V_{p\,(rect)} \tag{4.6}$$

In this power supply module we used step down transformer with transformer ratio 14.

General AC power supply in India is 220V 50Hz AC. As we know the relation between transformer ratio and voltages as shown in equation. Where V_1 and V_2 are primary and secondary voltages.

$$a = \frac{n_1}{n_2} = \frac{v_1}{v_2} \tag{4.7}$$

$$v_2 = \frac{220}{14} = 15.7\,Volt \tag{4.8}$$

So, finally ripple voltage Vr (pp) from this full wave rectifier is

$$V_{r\,(pp)} = \left(\frac{1}{50 * 300 * 400 * 10^{-6}}\right) * 15.7 \tag{4.9}$$

$$V_{r\,(pp)} = 1.61\,volt \tag{4.10}$$

We used simple voltage divider circuit to split this 15.7 V DC to 12V 1A and 5V 500mA.

4.2.3 Programming for IoT Transmitter using Embedded C

The main functionality of this algorithm is to collect the data from transducer and process it to digital levels and store it in database using communication unit. Programming language we used here was “Embedded C”. The whole algorithm is divided into 3 subdivisions as shown below Figure 4.25.

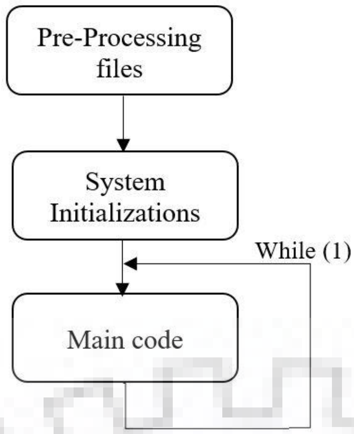


Figure 4.25 Working-flow block diagram of whole algorithm.

Pre-Processing files / System Initializations: SoftwareSerial.h header file is used in this algorithm because if we need more than one serial communication link but our Arduino model has only one USART, then you have to use the library SoftwareSerial that "simulates" the job of an USART only by software. Hence it is much less optimized than the hardware USART.

One advantage of SoftwareSerial is that we can map it to any pair of pins we like. Sample code for mapping to multiple communication systems is as shown in Figure 4.26.

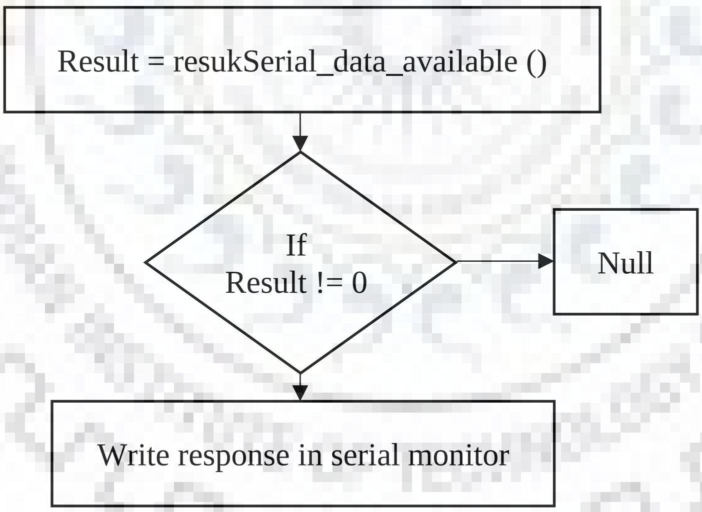


Figure 4.26 multiple communication system mapping algorithm.

Once the pre-processing file is added next an object is created in the name “SIM900”of the class of SoftwareSerial from pre-processing file as shown in Figure 4.27. The speed of communication over the channel i.e., baud rate is defined as 9600. Baud rate of serial port is also initialised to 9600 to enable serial monitor using this we can debug the process and fix error while system is working without switching off the whole system. Void setup is a function which will be running by compiler only once when system is switched on.

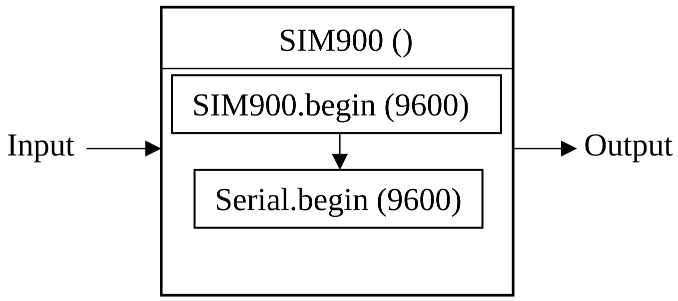


Figure 4.27 Creating object SIM900 () of the class SoftwareSerial.

The main part of code is written inside void loop function which will be running endless times as shown in Figure 4.28, full code is shown in Appendix B.

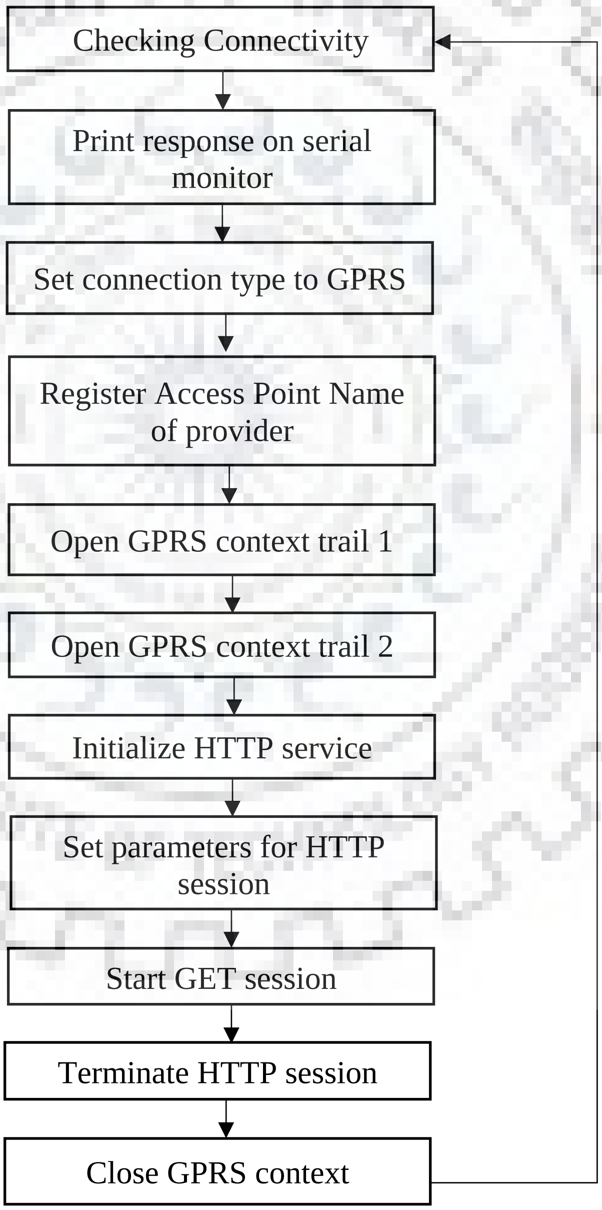


Figure 4.28 Main algorithm flow chart for IoT Transmitter device part 1.

In main code flow chart few set of commands need to be sent to communication module from processor they are, checking communication, registering communication type to GPRS, Access point network shortly named as APN, opening GPRS context to send data through http protocol, setting parameters for http protocol here we pass sensor data to public server, and finally closing the GPRS context. User can see the status of all these process through any serial monitor by setting baud rate to 9600. All these mentioned properties are shown in algorithm in Figure 4.28. Once, the transducer data is transmitted to public server, here in this sample code a variable named "pone" is defined to carry the sensor data. Php and MySQL programming.

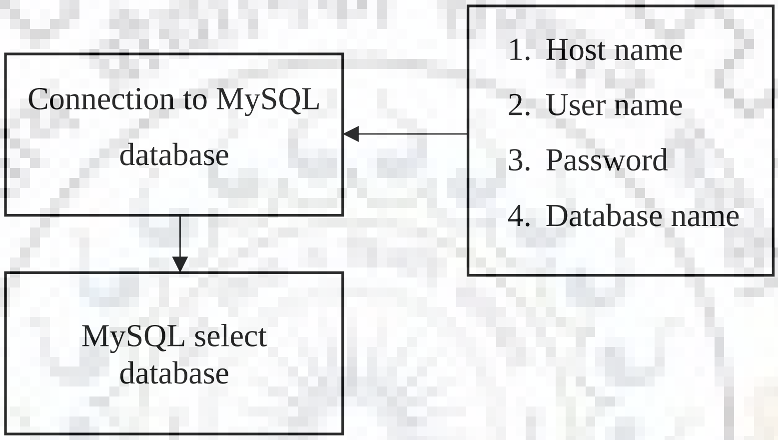


Figure 4.29 MySQL connection in server side script flow chart.

Languages are used to build server side link between database and IoT device communication module using GET method. Database connection and how sensor data is sent into database, codes are shown in Figure 4.29 and Figure 4.30. Full detailed code is shown in Appendix B.

Amazon Relational Database Services shortly named as RDS is been used to store sensor data in this project. There are some most important benefits using AWS RDS over other available database services are listed below.

1. Availability: AWS RDS is a highly available relational database that offers a feature called Multi-AZ, which provides a SLA up-time of 99.95%. With the Multi-AZ feature enabled in a production database.
2. Scalability: Database scalability can prove to be a real challenge if you try to scale your own, self-hosted database. Handling mandatory downtimes due to upgrades.
3. Automated backup: This functionality automatically performs a full daily snapshot of a database's data (during a preferred window of time set up by the user). It also captures your transaction logs as well as any updates to your RDS database.

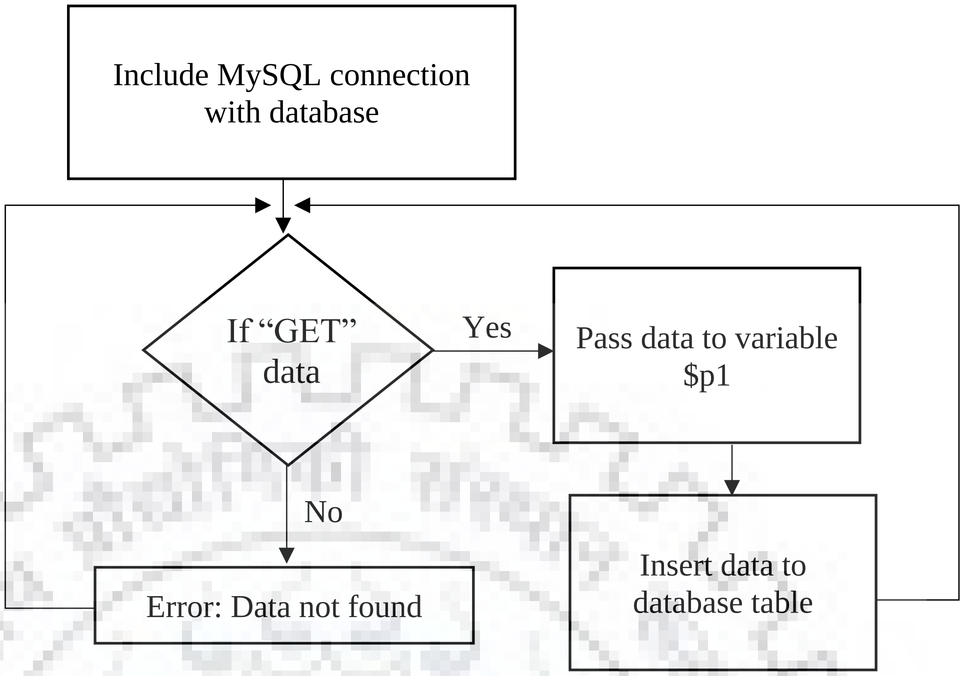


Figure 4.30 Sensor data receiving using GET method and sending to database flow chart.

4.2.4 System Debugging

Debugging is the process of finding and resolving defects or problems within a computer program that prevent correct operation of computer software or an electronic system.

Software serial is used here to build debugging feature. As we initiate serial monitor in baud rate 9600. Debugging of hardware working is possible using any serial monitors. Using this defects in coding or system functionality errors like loss of signal, wrong GPRS settings can be found immediately as shown in Figure 4.31.

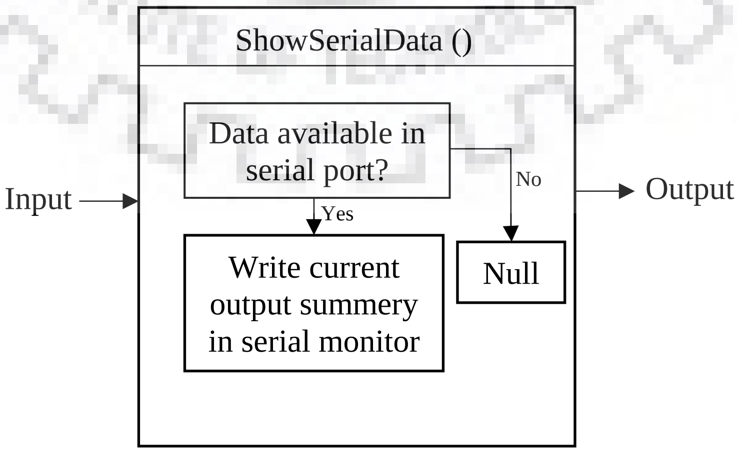


Figure 4.31 Serial monitor algorithm flow chart.

4.2.5 System 3D Modelling, building and testing

The process flow of building IoT transmitter device is as shown in Figure 4.32

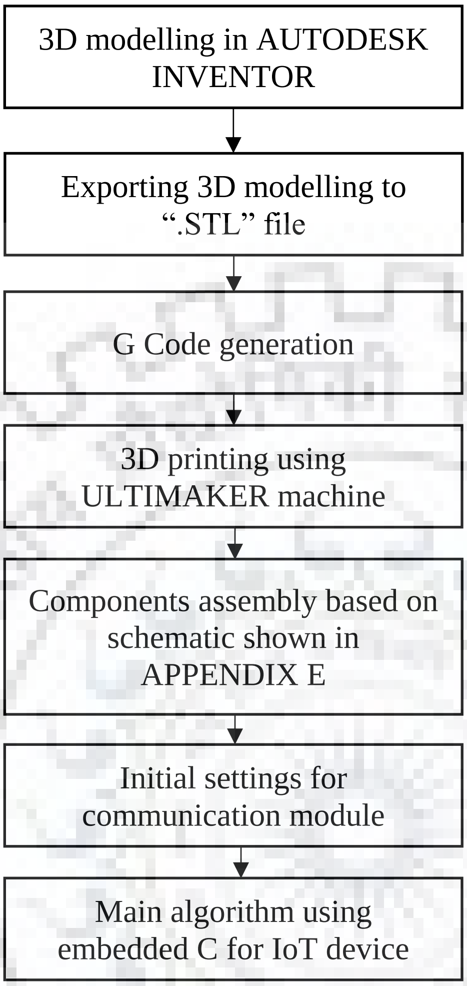


Figure 4.32 Data transmitter device manufacturing process flow chart.

In 3D computer graphics, 3D modelling is the process of developing a mathematical representation of any surface of an object in three dimensions via specialized software. Here Autodesk inventor design software has been used to realise the physical representation of this IoT transmitter device.

Physical view of final IoT data transmitter device is as shown in Figure 4.33. Device physical dimensions are 15.5x10.3x7 cm. one full 3D modelling done in Autodesk inventor software, we send this design for 3D printing by exporting modelling file to .STL file format.

Finally, using "Ultimaker CURA" software we generate Gcode for 3d printer machine to start printing each part. White PLA is used as printing material for 3D printing the design. 0.8mm nozzle profile is used here for printing, as nozzle size is high temperature around it is around 200^0C to 210^oC, so cooling fan is been switched on while printing this design.

Heat maintenance around the electronics system is very important criterion to maintain adequate performance of the whole system. There are multiple types of cooling systems available in existing electronics world as Convective air cooling, Forced air cooling, and Heat pipes

Here we used forced air cooling system as shown in Figure 4.33 to remove heat from the IoT device box.

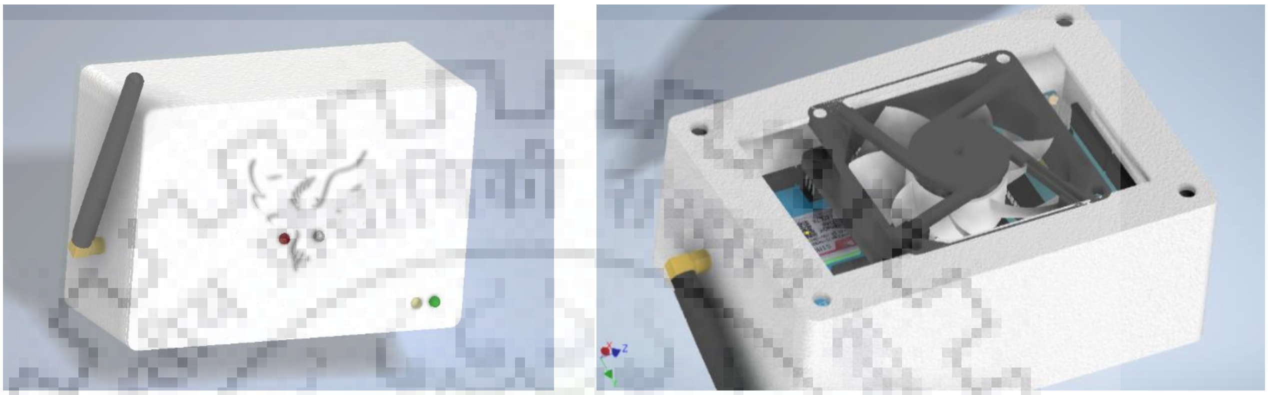


Figure 4.33 Dedicated Cooling system to remove heat accumulation around processor

Assembly of processing unit and communication unit inside casing is as shown in Figure 4.34, this alignment is made in such a way so that all the input and output ports are positioned on the same side of the case.

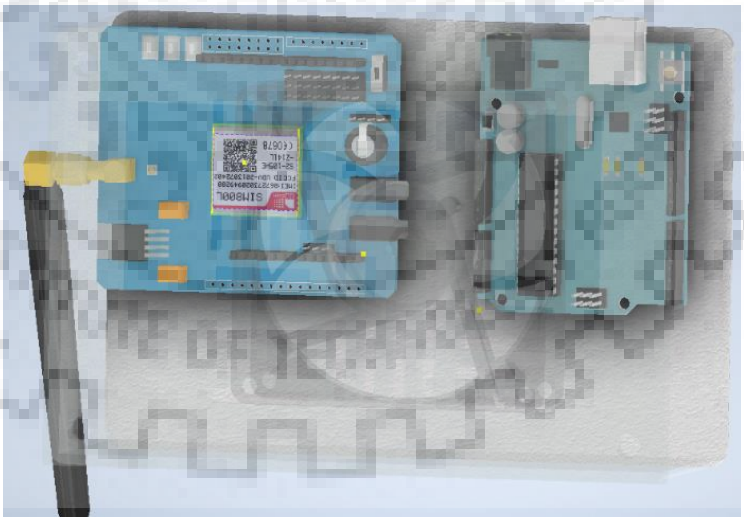


Figure 4.34 Alignment of Processor and Communication module

Once the building of whole IoT data transmitter using Arduino as a MCU is finished, wiring between 6 pin male connector and analogue pins in Arduino, connections between GSM/GPRS engine and Arduino, wiring between device cooling system and MCU, and on board indicators are made as shown in schematic in Appendix E. Finally, data transmitter using Arduino as MCU is developed, sensor data output ports will be connected to 6 pin male I/O port of this data transmitter device and transmit the data to remote database.

4.2.6 Bill of materials for data transmitter using Arduino UNO as Micro-controller.

Average cost of SCADA based monitoring and forecasting system was ₹50,000 per unit, the same tasks are performed using data transmitter using Arduino as MCU for Fixed cost of ₹4046 and yearly software service of ₹2600, detailed bill of materials are shown in Table 4.7. Device building cost of data transmitter using Raspberry pi as processor was ₹8071 and software service cost was same as using Arduino as shown in Table 4.3. So, it is clear that IoT data transmitter using Arduino as MCU is advantage with respect to cost as well as manufacturing and easy installation as most of the things during installation are automated using algorithms.

Table 4.7: IoT data transmitter device using Arduino as MCU bill of materials.

S.no	Quantity	Description	Component name	Cost
1.	1	Processing unit	ATmega328	₹450.00
2.	1	Communication unit	SIM 900A GSM modem	₹1325.0
3.	1	Cooling system	CPU Cooling Fan	₹299.00
4.	1	3D printing material	White PLA 332 grams	₹532.00
5.	1	On board indicators and I/O connectors	5mm Ultra Bright LED 6 pin male connector	₹840.00
6.	1	Power supply module	220V to 12V /5V Isolated Switching Power Supply Module	₹600.00
	IoT data transmitter using Arduino as MCU device cost			₹4046.0
7	1	AWS RDS Server service cost* (per year)	Amazon Web Services	₹2600.0*
	Yearly payment			**₹2600.0***

CHAPTER-5
DEVELOPMENT OF MONITORING PANEL AND FORECASTING METHODS

Monitoring panel and database system are built using following list of steps.

1. Hosting MySQL RDS instance in AWS service.
2. Writing server side script
3. Hosting file system and connecting to domain name service.

5.1 Hosting MySQL RDS instance in AWS service

AWS provides free tier 20GB Relational Database Service for education purpose, creating an instance here using RDS method and set our database is publically accessible. Main advantage of this server is its security levels are high using this we no need to maintain a separate data-security layer in our architecture [22].

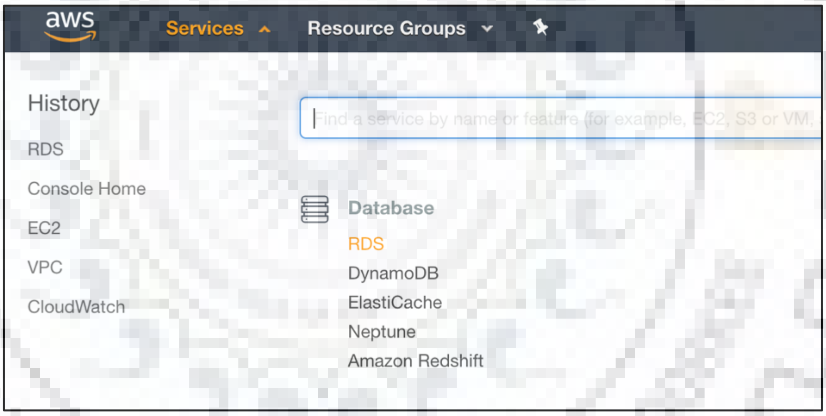


Figure 5.1 AWS RDS database services

Once creating RDS instance it is accessible in activated instances list as shown in Figure 5.1.

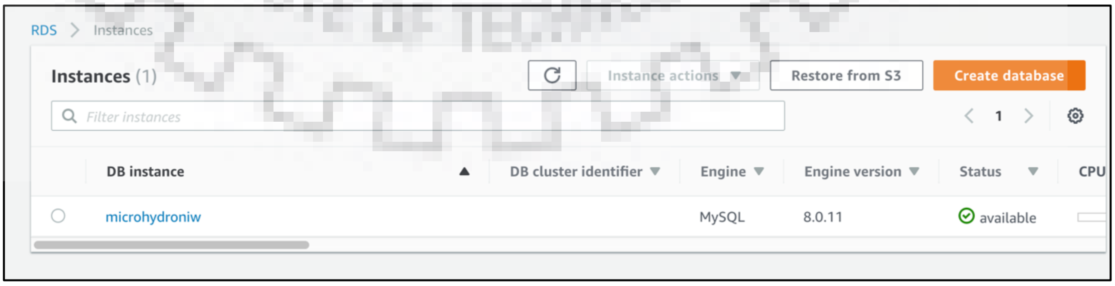


Figure 5.2 Creating MySQL engine RDS instance

Here, "*microhydroniw*" named RDS instance was created using MySQL engine with publically accessible settings in it as shown in Figure 5.2.

After entering into RDS instance finding necessary data to connect to this MySQL server using C++ or Php algorithms as explained in chapter 4. We need host name, database name, database password, user name to connect to RDS from remote servers, these data are found in details section as shown in Figure 5.3.

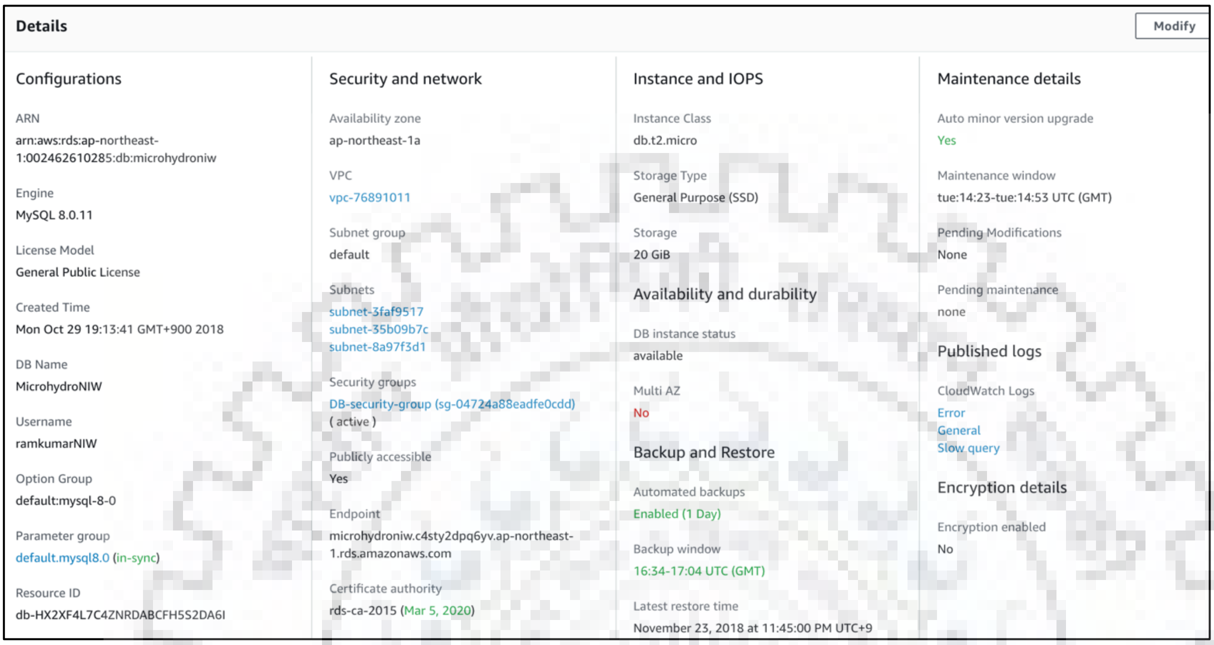
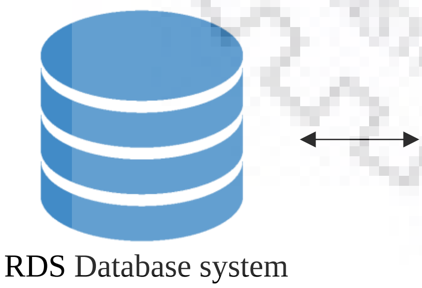


Figure 5.3 Details of RDS MySQL database created for this monitoring system

5.2 Writing server side script

All the server sire scripts are written in Php, MySQL and AJAX programming language. Automatic update of new data was included in programming using advanced java script and xml algorithm to avoid manual refresh to see current data. This monitoring panel working is shown in Figure 5.4.



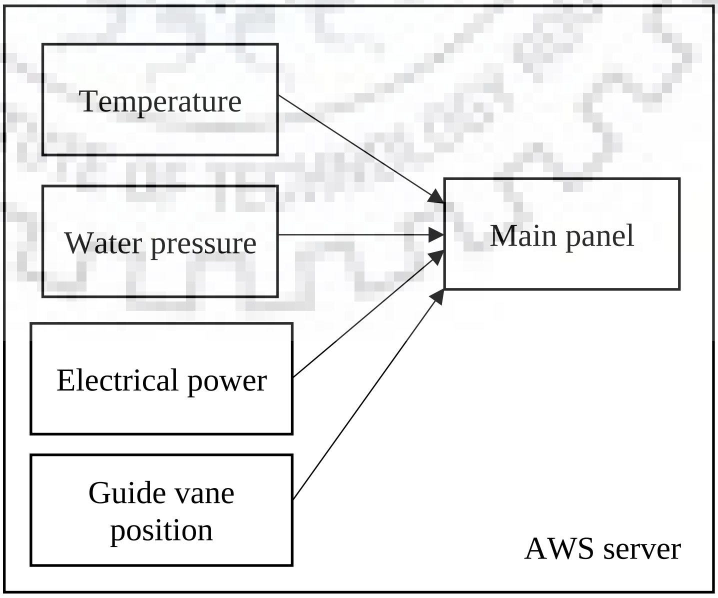


Figure 5.4 Monitoring panel working flow diagram

3 tables are created while Database design of this monitoring system. One was to store user details which is used to provide login system in monitoring panel and to avoid un authorised users to enter into monitoring panel. Second was table with 8 variables to store data received from IoT data transmitter using Raspberry pi as a processor. Last table is created for IoT data transmitter using Arduino as MCU to store sensor data in database. Full view of database design is shown in Figure 5.5

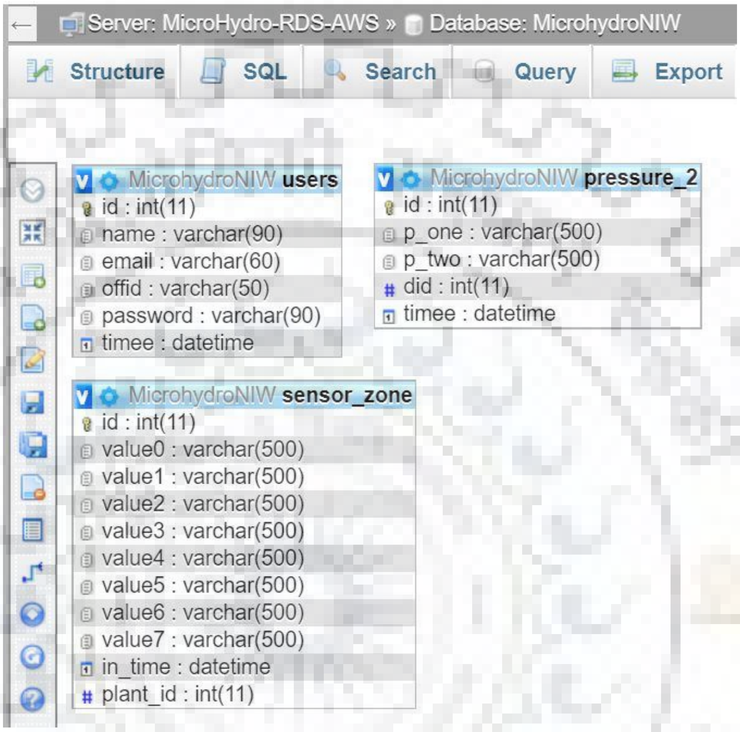


Figure 5.5 Database table design in RDS server

5.3 Hosting file system and connecting to domain name service

This monitoring system which is built was very user-friendly and lots of features available in it. PHP, CSS, JS and AJAX are used to build this system we use APACHE web server in AWS using EC2 service [24] [25] as shown in Figure 5.6.

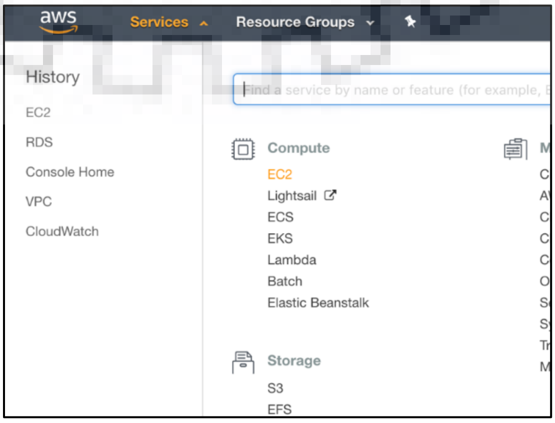


Figure 5.6 EC2 Services in AWS

After creating new instance in EC2 services by choosing windows 2012 server platform for monitoring system file management. It is possible to open this instance with key file as shown in Figure 5.7. EC2 create a unique global IP for our monitoring panel so that it will be globally accessible.

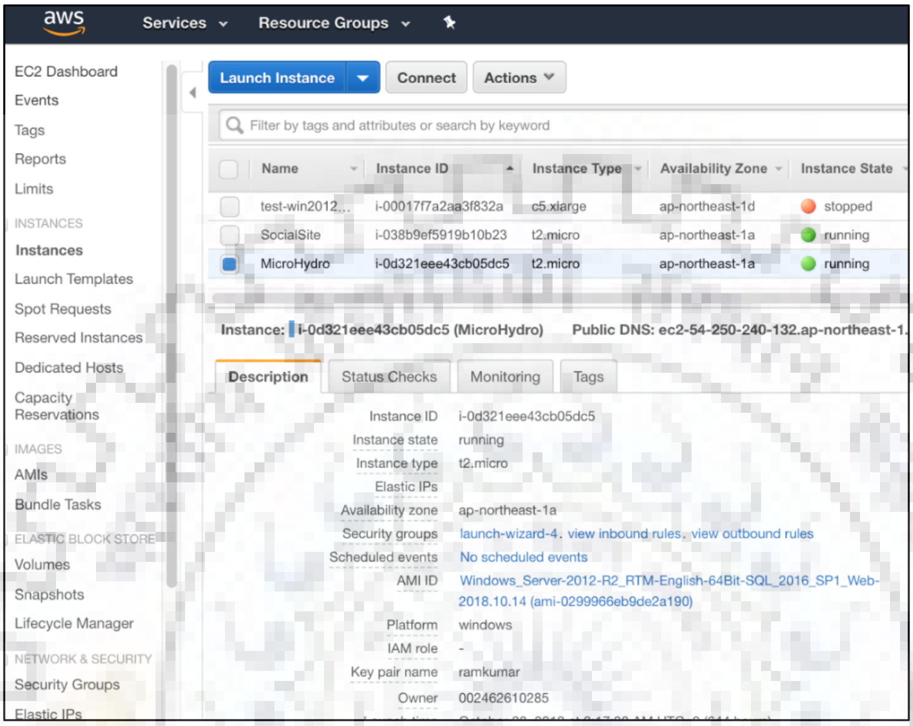


Figure 5.7 EC2 instance of AWS hosting our file system

File system includes all front-end and back-end Php algorithms in it, in which will together use to display build monitoring system. File system view is shown in Figure 5.8.

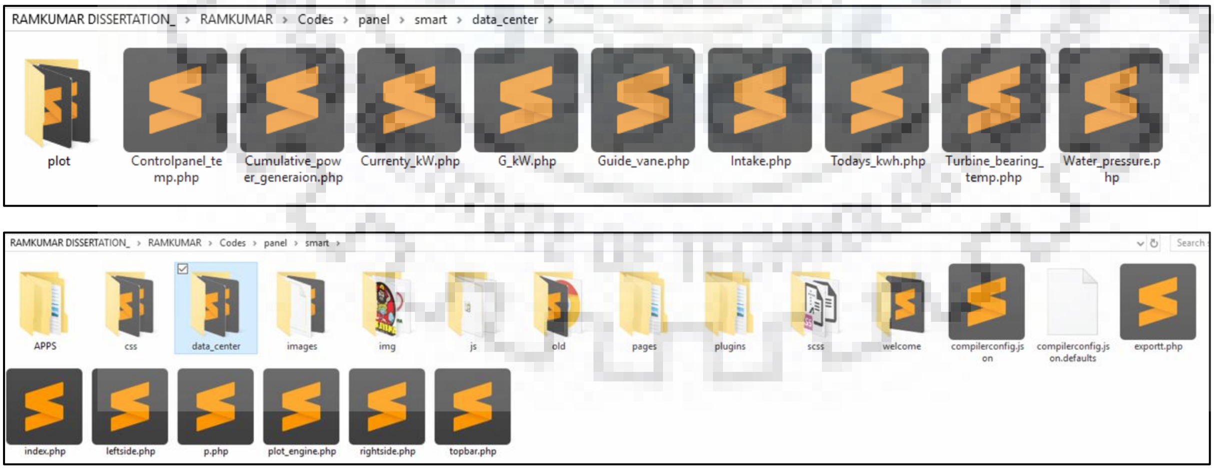


Figure 5.8 File system view of AWS EC2 services

All these algorithms do the following list of operations in Table 5.1.

Table 5.1 list of operations in this file system

1.	Login system
2.	Choosing MHPP
3.	Main Dashboard
4.	Exporting data to excel .csv format
5.	Data Plotting in time series plot
6.	Notification / Alert system

Usage flow of monitoring panel is shown in Figure 5.9.

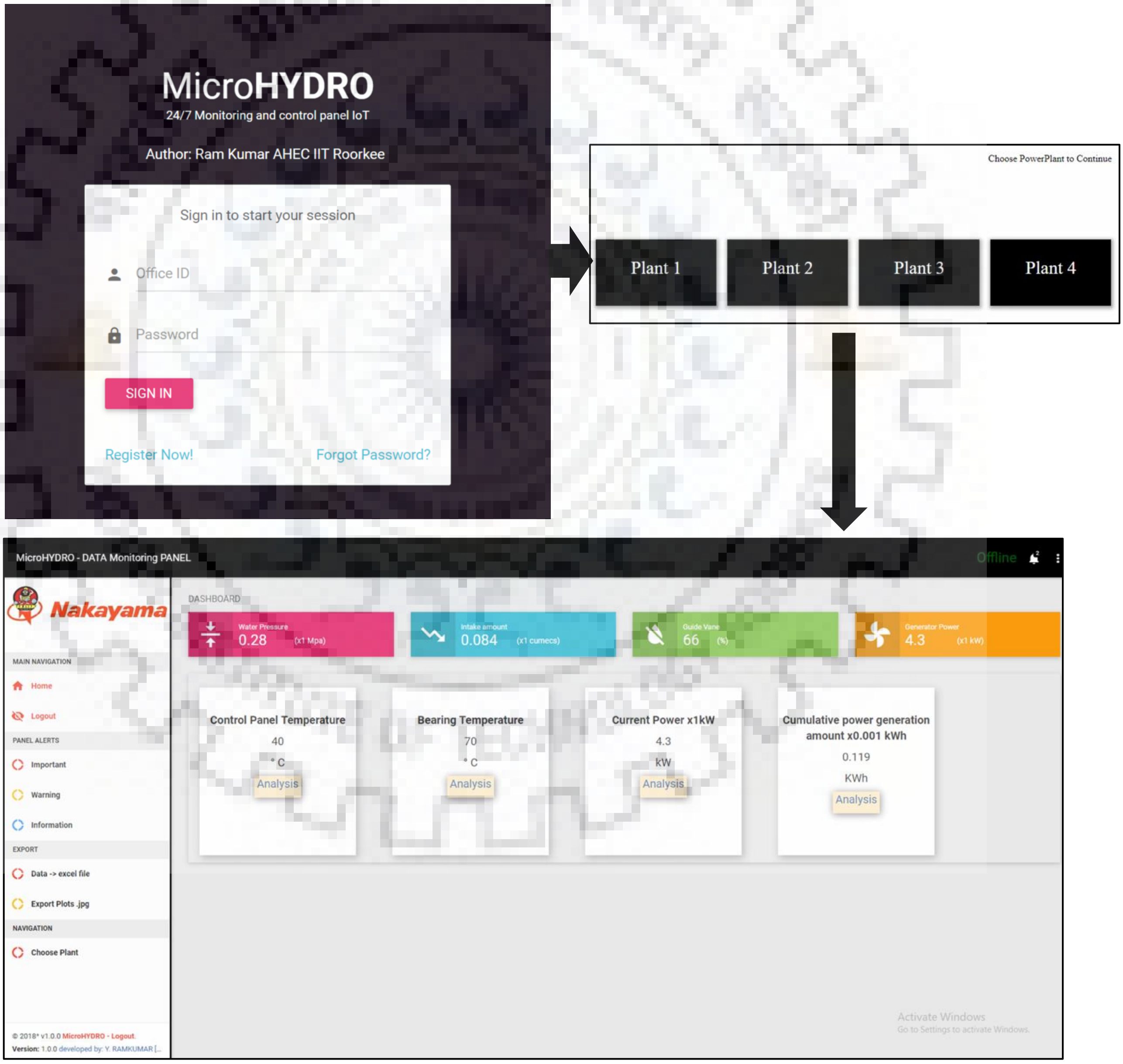


Figure 5.9 Monitoring panel of MHPP

RDS MySQL database [20] is as shown in figure 5.5. Sensor data are encrypted in digital levels format i.e., values range from 0 to 16777215. These digital levels are fetched and decrypted as shown in figure 5.9. Full algorithm of this system is given in Appendix C.

5.4 Time series Forecasting design

Forecasting is important in many situations: deciding whether to build another power plant in next five years requires to know about future demand, stocking an inventory also requires to be forecasted on stock requirements. Forecast can be required several years in prior, or only few minutes prior. In any circumstance forecasting is important tool to maintain effectiveness and efficiency in hydro power plants.

Before entering into forecasting model here all the list of terminologies are introduced listed below

a. Time series data
b. Univariate time series
c. Cross-Sectional data

a. **Time Series data:** A time series is a set of observations on the values that a variable takes different times. Such data may be collected at regular time intervals such as, monthly weekly or annually. This time series data in used for forecasting in data science.

b. **Univariate time series:** A univariate time series refers to a time series that consists of single observations recorded over regular time intervals.

c. **Cross-Sectional Data:** Such a type of data is collected by observing many subjects (such as control panel temperature, guide vane position, power generated) at the same point of time period.

5.4.1 Patterns emerging in time series data

Depending on the frequency of the data (hourly, daily, weekly, monthly, quarterly, annually) different patterns emerge in the data set which forms the component to be modelled. Sometimes the time series may be just increasing or decreasing over time with a constant slope or there may be patterns around the increasing slope.

Components of a Time series:

The pattern in a time series is sometimes classified into

a. Trend
b. Seasonal
c. Cyclical
d. Random components.

a. **Trend:** Along term relatively smooth pattern that usually persists for more than one year.
b. **Seasonal:** A pattern that appears in a regular interval where in the frequency of occurrence is within a year or even shorter.
c. **Cyclical:** the repeated pattern that appears in a time-series but beyond of one year. It is a wavelike pattern about a long term trend that is apparent over a number of years. Cycle are rarely regular in combination with other components.
d. **Random:** the component of a time series that is obtained after three patterns have been extracted out of the series is the random component. Therefore when we plot the residual series then the scatter plot should be devoid of any pattern and would be indicating only a random pattern around a mean value.

5.4.2 Different Time Series Process

1. **White Noise:** A series is called white noise if it is purely random in nature. Let ε_t denote such a series then it has zero mean [$E(\varepsilon_t)=0$], has a constant variance [$V(\varepsilon_t)=\sigma^2$] and is an uncorrelated [$E(\varepsilon_t\ \varepsilon_s)=0$] random variable. The scatter plot of such a series indicate no pattern and hence the future values of such a series is not possible.
2. **AutoRegressive Model:** An AR model is on ewhich Y_t depend only on its own past values Y_{t-1}, Y_{t-2}, Y_{t-3}, etc. thus $Y_t = f(Y_{t-1}, Y_{t-2}, Y_{t-3},\ldots., \varepsilon_t)$. A general representation of this model where is depends on p values called as AR(p) model and is represented as shown below:

$$Y_t = \beta_o + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \beta_3 Y_{t-3} + \beta_4 Y_{t-4} + \cdots + \beta_p Y_{t-p} + \varepsilon_t \qquad (5.1)$$

3. **Moving Average Model:** A moving average model is one when Yt depends only on the random error terms which follow a white noise process i.e.

$$Y_t = f(\varepsilon_t,\ \varepsilon_{t-1},\ \varepsilon_{t-2},\ \varepsilon_{t-3}, \ldots) \qquad (5.2)$$

A common representation of a moving average model where it depends on q of its pas values is called MA(q) model and is represented as show below:

$$Y_t = \beta_o + \varepsilon_t + \Phi_1 \varepsilon_{t-1} + \Phi_2 \varepsilon_{t-2} + \Phi_3 \varepsilon_{t-3} + \cdots + +\Phi_q \varepsilon_{t-q} \qquad (5.3)$$

The error term ε_t are assumed to be white noise process with mean zero and variance σ^2

4. **AutoRegressive Moving Average model:** there are situations where the time-series may be represented as a mix of both AR and MA models referred as ARMA(p,q). The general form of such a time-series model, which depends on p pf its own past values and q past values of white noise disturbances, takes the form:

$$Y_t = \beta_o + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \beta_3 Y_{t-3} + \beta_4 Y_{t-4} + \cdots + \beta_p Y_{t-p} + \varepsilon_t + \Phi_1 \varepsilon_{t-1} + \Phi_2 \varepsilon_{t-2} + \Phi_3 \varepsilon_{t-3} + \cdots + +\Phi_q \varepsilon_{t-q} \qquad (5.4)$$

5.4.3 Stationarity of a Time Series

A series is said to be "Strictly Stationary" if the marginal distribution of Y at time t [$p(Y_t)$] is the same as at any other point in time.

Therefore,

$P(Y_t) = P(Y_{t+k})$ and $P(Y_t, Y_{t+k})$ does not depend on t. (hence $t \geq 1$ and k is an integer)

This implies that the mean, variance and covariance of the series Y_t are time invariant.

A Series which is non-stationary can be made stationary after differencing. A series which is stationary after being differentiated once is said to be integrated of order 1 and is denoted by I(1). In general a series which is stationary after being differentiated d time is said to be integrated of order d, denoted by I(d).

Therefore, a series, which is stationary without differencing, is said to be I(0).

Necessity of the assumption of stationarity

1. The results of classical economic theory are derived under the assumption that variables of concern are stationary.
2. Standard techniques are largely invalid where data is non- stationary.
3. Sometimes, autocorrelation may result because the time series is non-stationary.
4. Non-stationary time series regressions may also result in spurious regressions, i.e. cases when the regression equation shows significant relationship between two variables when actually there should not be any such relation.

The predication and Forecasting of univariate time-series models is carried out using Box-Jenkins (B-J) methodology which has the following steps:

1. Identification
2. Estimation
3. Diagnostic Checking

The B-J methodology is applicable only to stationary variables.

1. **Identification:**

   a. **Autocorrelation function (ACF):** Autocorrelation refers to the way the observations in a time series are related to each other and is measured by a simple correlation between current observation (Y_t) and the observation p periods from the current one (Y_{t-p}).

$$\rho_k = Corr(Y_t, Y_{t-1}) = \frac{cov(Y_t, Y_{t-p})}{\sqrt{var(Y_t)}\sqrt{var(Y_{t-p})}} = \frac{\Upsilon_p}{\Upsilon_0} \qquad (5.5)$$

b. **Partial Autocorrelation function (PACF):** partial autocorrelation function are used to measure the degree of association between Y_t and Y_{t-p} when the effect of the other time lags 1, 2, 3… (p-1) are removed.

c. **Inference from ACF and PACF:** the theoretical ACFs and PACFs are available for various values of the lags of autoregressive and moving average components i.e. p and q. therefore, a comparison of the correlograms (Plot of samples ACFs versus lags) of the time series data with the theoretical ACFs and PACFs leads to the selection of the appropriate ARIMA (p, q) model.

d. The general characteristics of theoretical ACFs and PACFs are shown in table 5.2

Table 5.2: Characteristics of ACF and PACF

MODEL	ACF	PACF
AR(p)	Spike decay towards zero	Spike cut-off to zero
MA(q)	Spike cut-off to zero	Spike decay towards zero
ARMA(p,q)	Spike decay towards zero	Spike decay towards zero

2. Estimation:

Several methods available for estimating the parameters of an ARIMA models depending on the assumptions one makes on the error terms. They are (a) Yule Walker procedure (b) method of moments (c) Maximum Likelihood etc.

Since, no closed form estimates can be found analytically, one has to seek computer aided methods and most statistical econometric facilities the process of estimation.

3. Diagnostic checking:

Different methods can be obtained for various combination of AR and MA individually and collectively. The best model is obtained by following the diagnostic testing procedure. The various tests are given by:

a. **Lowest value of AIC/ BIC/ SBIC** – the model with the lowest value of the above criterion is chosen as the best model.

b. **Plot of the residual ACF:** on fitting the appropriate ARIMA model, the goodness of fit can be estimated by plotting the ACF of residuals of the fitted model. If most of the sample autocorrelation of the residuals lie within the limits $(-1.96/\sqrt{N}, +1.96/\sqrt{N})$, where N is the number of observations,

Then the residuals are white noise indicating that the model fit is appropriate.

5.4.4 R-Programming forecasting process flow chart

Detailed programming is shown in Appendix F. in the next chapter implementation of the monitoring and forecasting system in existing power plant is discussed. Flow chart of forecasting system implemented is shown in Figure 5.10.

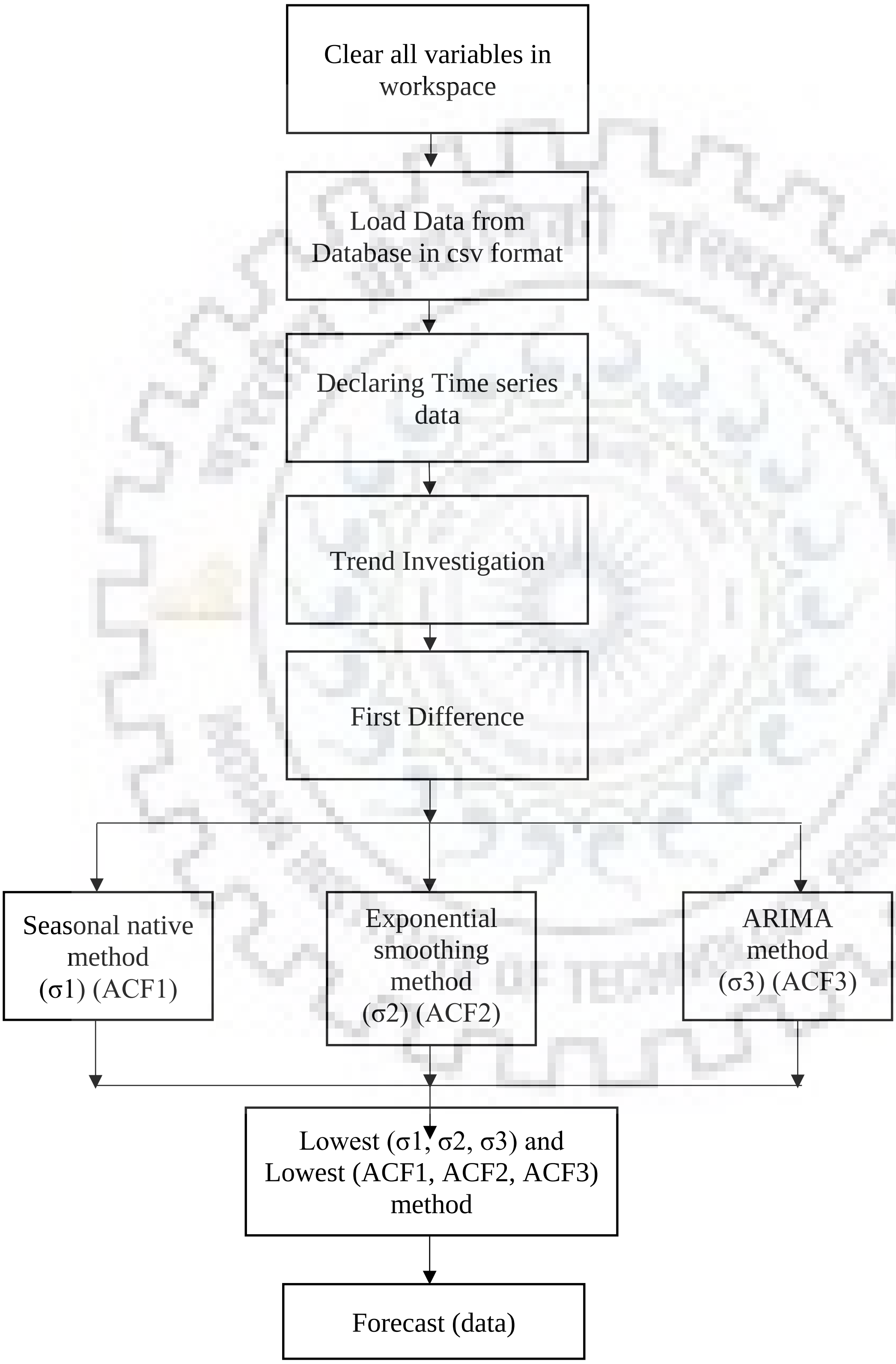
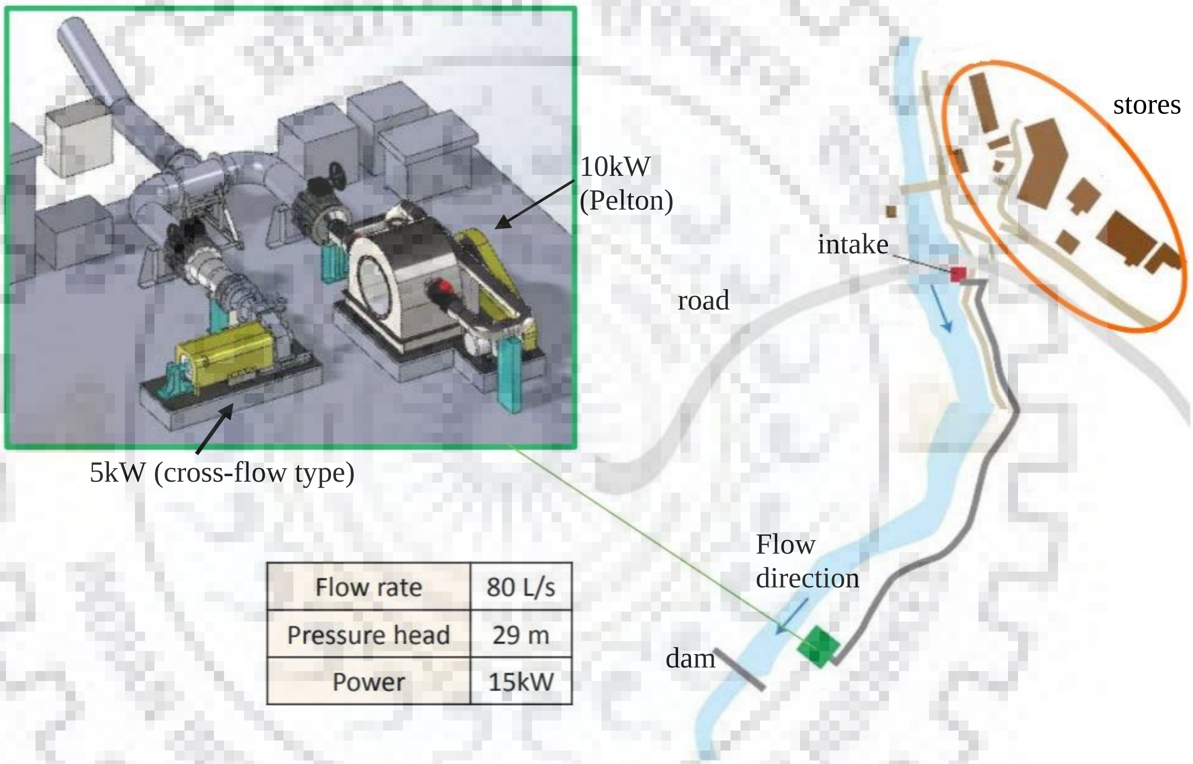


Figure 5.10 Data forecasting process flow chart

CHAPTER-6
INSTALLING IoT DEVICE AT MHPP IN SAGA, JAPAN

The IoT based monitoring and forecasting system was installed in existing MHPP, which was named as “SHIRAITO FALLS, FUKUOKA MHPP, SAGA, JAPAN. The basic plan of this power plant was to supply electricity for sightseeing facilities and appeal for MHPP generation as a symbol of the introduction of the renewable energy to the city. With aim to increase the environmental consciousness of citizens and new tourists attractions. Layout and location in map are shown in Figure 6.1 and Figure 6.2.



Flow rate	80 L/s
Pressure head	29 m
Power	15kW

Figure 6.1: Layout of Shiraito MHPP [29]

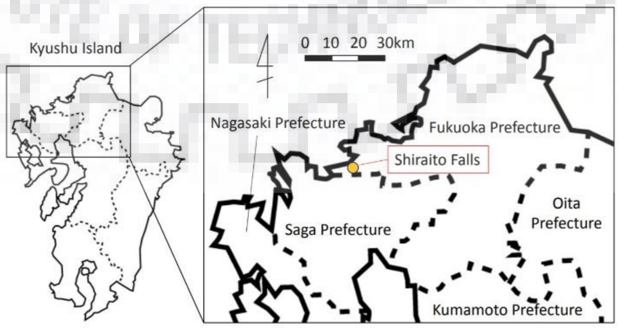


Figure 6.2: Layout of Shiraito MHPP [29]

This was a 15kW in which 10kW was generated from dual jet Pelton turbine and remaining 5kW was generated from cross flow turbine as shown in Figure 6.1. Power Plant specifications are given in Table 6.1. From past few years this MHPP was using PLC method for control and monitoring this plant from remote place. The main problem with this method was PLC is programmed such that it will send all sensor data with per min periodicity through SMTP protocol which is through e-mail after end of every day.

Table 6.1 MHPP Specification

SHIRAITO FALLS MHPP	
Flow Rate	0.09 m^3/sec
Head	29 m
Power Generation	15 kW
Turbine type	Cross Flow 5 kW and Pelton 10 kW
Nominal water pressure	0.317 Mpa
Transducers used in this MHPP	
Water Pressure	GP-M series dual mode sensor (hysteresis and digital mode)
Control Panel temperature	DHT22 series sensor
Bearing temperature	FT Series Digital Infrared Temperature Sensors
Guide Vane position	LP-200T series sensor
Power Measurement	KV-5000 series KL-WH1 sensor

6.1 Sensors used in MHPP are tested with this IoT data transmitter device

Before entering into sensor testing and simulations with IoT transmitter device, the process of selecting sensors according to the need in power plant is below.

6.1.1 Choosing Sensors and Signal Conditioning

Choosing transducers for industrial purpose is sometimes tricky task. As wide variety of sensing technologies and wide range of transducers available in market.

The actual difficulty is not about selecting sensor for the process, it is all about how to eliminate all the non-suitable choices. Choosing transduces is like a process of rising couple of questions to eliminate available wide range of technology or product that are not fit for our application [26].

These 6 steps are followed during the process of selecting sensors or transducers as shown in Figure 6.3.

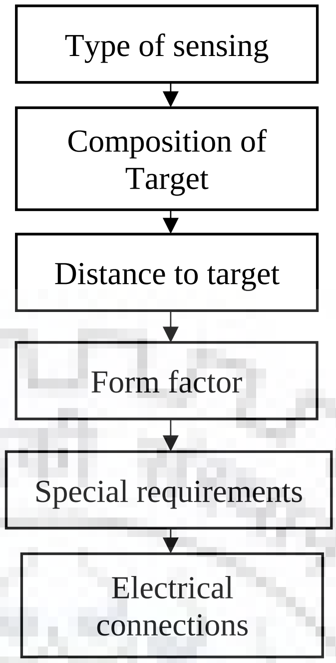


Figure 6.3 Steps followed while choosing sensor

Step 1 Type of Sensing: for example if it is required to sense the distance of obstacle, means we are looking for proximity transducer. But there are several kinds of proximity sensors available in market which can detect presence of object. Inductive, ultrasonic, capacitive, magnetic and so on. Are all possible types of transducers at this stage of the selecting a transducer.

Step 2 Composition of Target: the obstacle that was going to detect can be any kind of material, may be metallic, liquid, granular or sometimes non-metallic let us sat metallic obstacle was required to be detected. Then Photoelectric, Ultrasonic, Capacitive and Inductive are all capable of detecting metallic obstacle.

Step 3 Distance to Target: in this stage a question arises about distance between object and the transducer. If for example using this proximity sensor as encoder to detect the rpm of shaft between turbine and generator the in these kind of compact machinery all the sensors and machine parts will be close to each other, in such a situation inductive transducer will be the best choice. And this inductive type transducers have ability to detect obstacles in short distance (1mm to 50mm) when compared to other transducers technologies.

Step 4 Form Factor: here in our example sometimes it’s fairly tight space and we might not get space to choose most common inductive transducer. The threaded tubular housing one. So in this case we are going to choose block style rectangular type.

Step 5: Control Interface: now a days, there are multiple type I/O pin features available in market such as 3 wire dc and 2 wire ac/dc, but majority of control system requires 3 wire dc transducers. In our example also we need 3 wire dc transducer PNP normally open type.

Step 6 Special Requirements: here in this sage it is required to take some important protection requirements for inductive transducers such as this transducer may get damage due to high nearby welding temperature (more than 80°C) or sometimes water splash may happen and damage this transducer. So Special requirements will be taken according to different possible damages may happen to this transducer.

Step 7 Electrical Connection: typically transducers are available in three possible electrical connections a) pre-wired cable with flying leads, b) built in fast disconnect connectors, c) pig tail connector. Last connection type is terminal chamber type this connection was common in the days when proximity transducers were used to replace mechanical limit switches, but this kind of connection is becoming less coming in today's industrial electronics usage.

By following above steps we choose the following transducers to install in MHPP for sensing parameters and monitor the parameters of power plant. Simulations and testing of sensors which are used in existing MHPP with IoT data transmitter which was built are explained in below paragraphs.

The following list of sensors are chosen from MHPP to check handshaking with IoT data transmitter device as shown in Table 6.2.

Table 6.2 list of sensors chosen for testing purpose.

S no	Sensor name	Measuring parameter
1	DHT22	Humidity and temperature
2	GP-M	Water pressure
3	LP-200T	Value position, guide vane position
4	KV-7000	Electrical power

Before installing IoT data transmitter device to MHPP for collecting data all sensors are tested and necessary adjustments in schematics or settings changes are made to make all sensors output data signal voltage levels compatible to data transmitter device. This process prevents over current or voltage passing through data transmitter which may damage processing unit or sometimes communication unit.

6.1.2 DHT22 series sensor testing (ºC and %RH)

DHT stands for digital humidity and temperature module is one that contains the compound has been calibrated digital signal output of the temperature and humidity sensors. The sensor includes a capacitive sensor wet components and high-precision temperature measurement devices, and connected with a high-performance 8-bit microcontroller. The product has excellent quality, fast response, and strong anti-jamming capability.

Standard single-bus interface, system integration quick and easy. Small size, low power consumption, signal transmission distance up to 20 meters. It can measure the environment temperature and humidity to meet the high demand. The product has high reliability and good stability. If it's used and combined with special sensor Arduino expansion board, it will be easily implemented the interactive effect which related to the temperature and humidity perception. Detailed specifications of DHT22 sensor are shown in Table 6.3

Table 6.3 Specification of DHT22 Sensor

parameters	value
Supply Voltage	5V
Temperature Range	-40 to 120ºC / resolution 0.1ºC / error < ±0.5ºC
Humidity Range	0 to 100% RH / resolution 0.1 RH / error ± 2% RH
Wiring map	Vcc, Gnd, Signal (SDA type)
Size	38 x 20mm (1.50x0.79")

Schematics and physical layout:

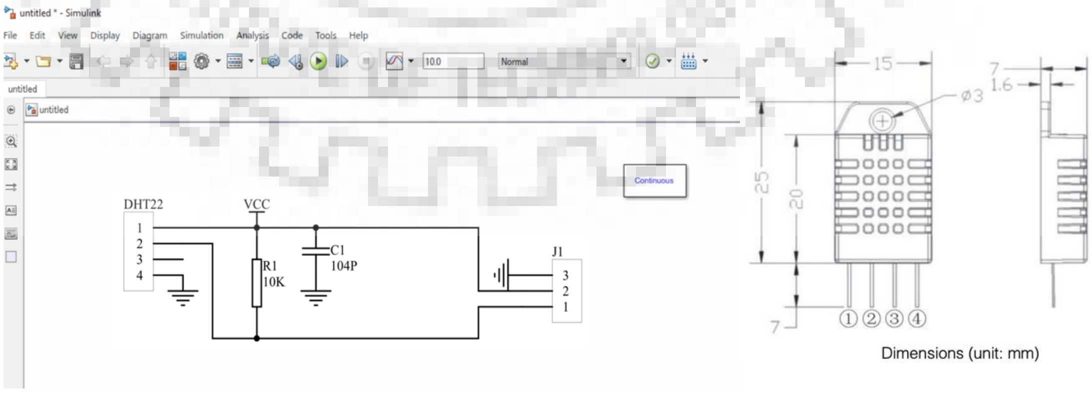


Figure 6.4 Schematics and physical layout of DHT22

As shown in Figure 6.4 MATLAB schematic explains the stage 1 i.e., front edge connection denoted by DHT22 of DHT transducer to Signal conditioning / ADC module which is in stage 2 i.e., processing edge input pin denoted by J1. A pull up resistor is added to schematic between VCC and SDA pin serial data bidirectional port, an example of serial data from sensor is explained here in equation 6.1 [17].

SDA for communication and synchronization between the microprocessor and the AM2302, single-bus data format, a transmission of 40-bit data, the high first-out. Specific communication timing shown in Figure 6.5, the communication format is depicted in Table 6.4.

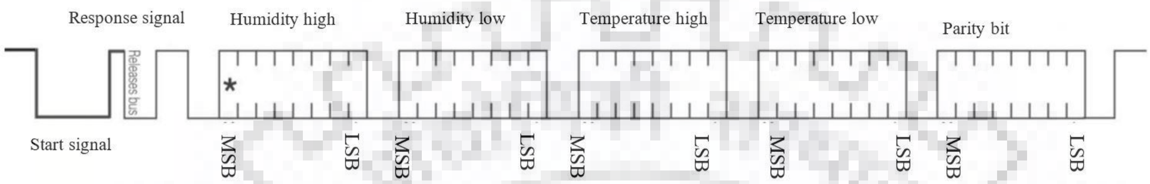


Figure 6.5 DHT22 Single-bus communication protocol

Table 6.4 Single bus data communication format is as follows

Name	**Single-bus format definition**
Start signal	Microprocessor data bus (SDA) to bring down a period of time (at least 800μ s) [1] notify the sensor to prepare the data.
Response signal	Sensor data bus (SDA) is pulled down to 80μ s, followed by high-80μ s response to host the start signal.
Data format	Host the start signal is received, the sensor one-time string from the data bus (SDA) 40 data, the high first-out.
Humidity	Humidity resolution of 16Bit, the previous high; humidity sensor string value is 10 times the actual humidity values.
Temperature	Temperature resolution of 16Bit, the previous high; temperature sensor string value is 10 times the actual temperature value; The temperature is the highest bit (Bit15) is equal to 1 indicates a negative temperature, the temperature is the highest bit (Bit15) is equal to 0 indicates a positive temperature; Temperature in addition to the most significant bit (Bit14 ~ bit 0) temperature values.
Parity bit	Parity bit = humidity high + humidity low + temperature high + temperature low

Testing calculations of temperature and humidity from sensor data through SDA pin 40-bit data while testing based on 40-bit data received is as shown in Appendix G. The total working architecture of DHT22 in MHPP is shown in flow-chart diagram in Figure 6.6.

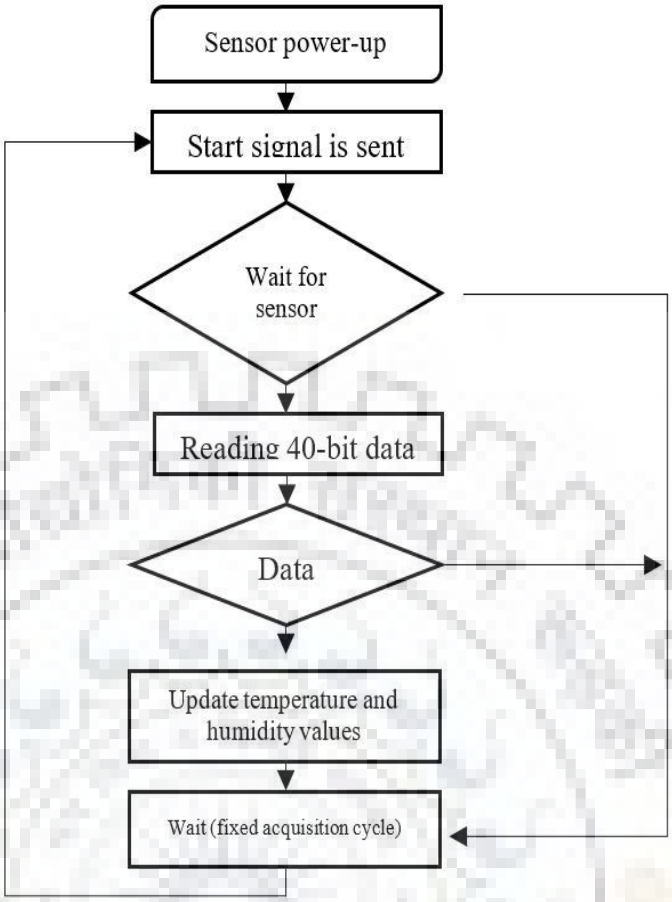


Figure 6.6 Single-bus to read the flow chart

6.1.3 KEYENCE GP-M series sensor settings (Pa)

This is capable of measuring pressure in water this single device can be used in wide variety of applications. Advantage of using digital pressure sensor are shown in Table 6.7.

In conventional analogue type pressure sensor it is very difficult to understand the values, more memory space required due to large size of data, and outputs are not provided. In GP-M series sensor it is easy to understand as numerical display is also included, output can be easily seen at a glance from any direction, and this includes range adjustable and analogue type. So, GP-M series sensor is advantages than traditional pressure sensor.

Physical layout and operations:

Operation features includes:

1. NPN / PNP selectable
2. Selectable between [2 outputs] or [1 output + analogue output]
3. Various cables and adapter accessories are applicable.

The GP-M Series can monitor fluids between -20 and +100°C-4 and +212°F and can therefore be used in a wide variety of applications ranging from cooling, heat retention, and sterilization. As shown in Figure 5.7.

Figure 6.7 GP-M sensor hardware

Ranging output analogue: The data is output within the range of 4 to 20 mA in correspondence with the upper and lower limit values of the required pressure range. The pressure range can be adjusted as desired to maximize resolution. Operate this is NPN mode so its schematic is as shown in Figure 6.8.

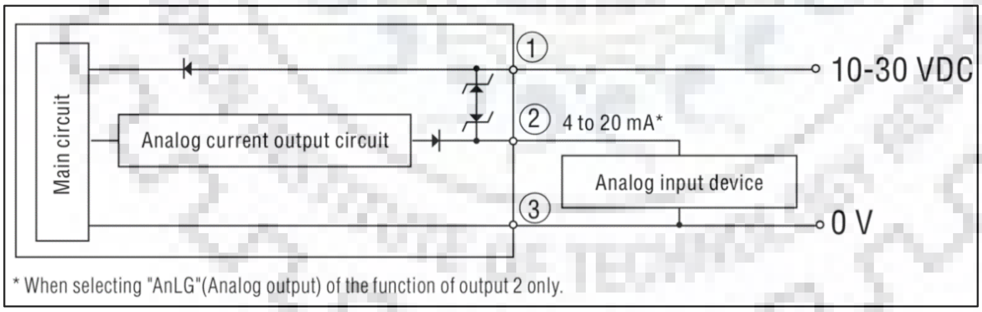


Figure 6.8 Analog output diagram

For installing this pressure sensor, setting its operation to Analog NPN output 2 modes. Then Figure 6.8 used to connect this sensor with ADC unit. This sensor has built in pull up resistor so no additional sensor signal conditioning needed for this. Allowable pressure is up to 30 Mpa and fluid temperature is up to 70°C.

6.1.4 LP-200T series sensor (Ω ohm)

This sensor is used to measure percentage of valve or guide vane position open in MHPP. Green Pot is the general term that LP-200T has used for conductive plastic contact potentiometer. Green Pot is identified by Green-coloured name plate attached to the part. This works with principle of potential divider as shown in Figure 6.9.

Schematic: If finite load resistance is loaded on sensor output, linearity will become distorted as expressed in the following equation 6.2.

$$E_{max} = \frac{14.8}{\left(\frac{RL}{RP}\right)+0.22}\ \% \qquad (6.2)$$

E_{max} : Maximum linearity distortion caused by load resistance

RP : Total Resistance of Potentiometer

RL : Load Resistance

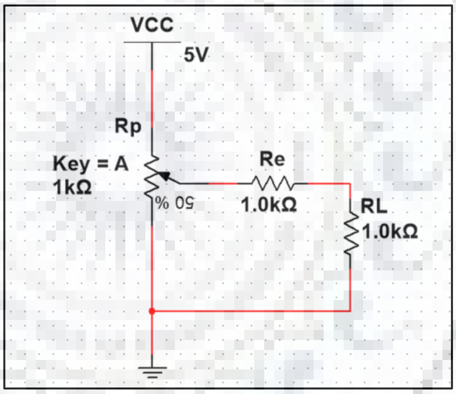


Figure 6.9 Working schematic of LP-200T series sensor

It is recommending that the added load resistance should be more than 100 times of the total resistance. The higher the load resistance, the less effect of electrical noise on the output of the potentiometer. It is also effective in increasing the durability of the Green Pot.

Equivalence Noise Resistance and Load Resistance:

Contact potentiometer generates the equivalence noise resistance because of its structure. It is possible to reduce the effect of noise resistance on the output by adding large load resistance to the potentiometer.

The equivalence noise resistance is expressed "Re" as shown the Figure 6.10. Voltage at point A is divided as shownin equation 6.3.

$$Va = \frac{RL}{(Re + RL)} \times Vout \tag{6.3}$$

Where:

V_a	: Voltage at point A
V_{out}	: Output voltage of Potentiometer
RL	: Load Resistance
Re	: Equivalence Noise Resistance

Physical Layout (dimensions are mm):

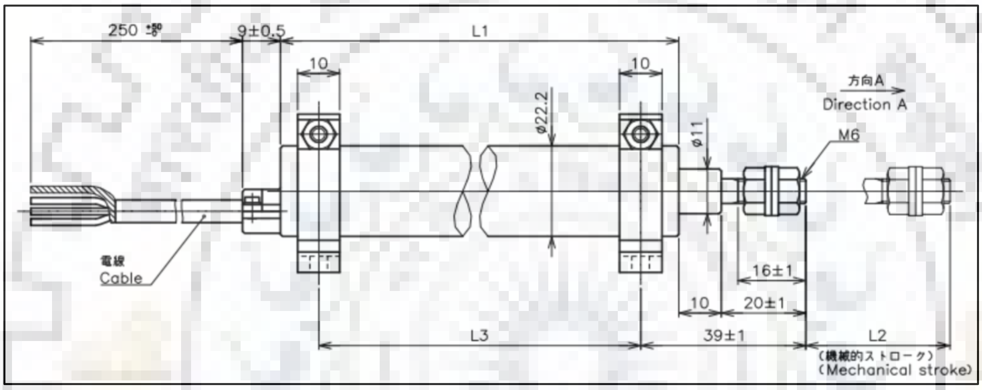


Figure 6.10 Physical layout of LP-200T

Output and Mounting characteristic:

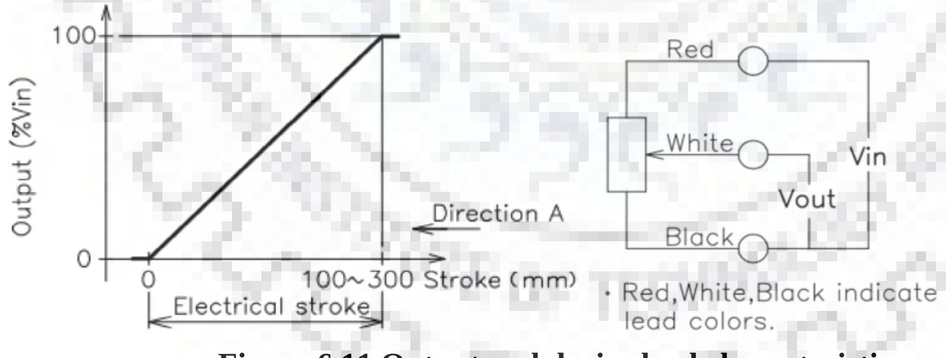


Figure 6.11 Output and device lead characteristics

6.1.5 KV-7000 series sensor (kW)

This module can measure Demand value, instantaneous active power, instantaneous reactive power, active power, reactive power, phase-to-phase voltage, phase current, power factor, and frequency. The specification of this module is in below Table 6.5.

Table 6.5 Specification of KV-700 power measurement module

Parameter	Values and range
Number of channels	1CH
Input voltage	100 to 250 VAC (sine wave), 50/60 Hz
Power consumption	10VA or less
Phase wire type	3-phase 3-wire, single-phase 2-wire, single phase 3-wire
Measurement items	Demand value, instantaneous active power, instantaneous reactive power, active power, reactive power, phase-to-phase voltage, phase current, power factor, frequency.
Power voltage	100 to 220 VAC ± 10%, 50/60 Hz
Current measurement	Optional external current sensors (three types for the CT: 50 A, 100 A, and 250 A) are available. Use the setting switch to select the current setting.
Power off memory	Non-volatile memory (integrated active power and reactive power)
Integrated value clear	Use the front panel key switch, control input terminal, or KL to provide the clear instruction.

This module is integrated with PLC unit and data is stored in excel sheet is the traditional way of measuring. But now this new monitoring system will transfer all the measured power data to remote database RDS and make available to monitor from remote places.

6.2 Installing IoT data transmitter device in existing MHPP for testing monitoring system.

Traditional SCADA monitoring system with PLC are very costly and very sensitive maintenance is required. So in this thesis, building an affordable and efficient monitoring system for hydroelectric power plants for long use and less maintenance was introduced [28].

The plan was to install this IoT device and collect data for 6 hours, while testing in existing MHPP, Raspberry pi MCU and ADC module was used in IoT data transmitter device without changing any system architecture and algorithms as explained in chapter 4. After testing later it was figured out that same method can be done by using Arduino MCU this is the reason of explaining both MCU in chapter 3. This new monitoring system built using following electronic and software components as listed in Table 6.6.

Table 6.6 List of components used in this monitoring system

Electronics Component	Name & Specification
Micro-Processor	Broadcom BCM2837B0, Cortex-A53 64-bit SoC @ 1.4GHz, Raspberry Pi B. Arduino UNO.
Memory	Internal: 2GB; External: 16GB SD Type
GPIO Pins	Extended 40-pin GPIO header
Input power:	5V/2.5A DC via micro USB connector. 5V DC via GPIO header. Power over Ethernet (PoE)–enabled (requires separate PoE HAT).
Environment	Operating temperature, 0–80°C
Software Component	Name & Specification
Programming used	C++ is used to configure MCU to AD/DA module. Php, CSS, JS, AJAX, MySQL are used to build web monitoring panel with all user friendly features.
Software used	MATLAB Sublime text editor MobaXterm Raspberry Operating system
Servers used	AMAZON web services shortly AWS RDS MySQL server APACHE XAMPP server

The working of this whole IoT monitoring and forecasting system is categorised into 4 stages they are

Stage 1: Front edge

This stage deals only with sensor calibration according to power plant parameter requirements, interfacing them with signal conditioning unit.

Stage 2: Processing edge

Necessary Amplification or DE amplification are done here to adjust the sensor data to meet MCU (micro controller unit) GPIO requirement. The main challenge here we faced was BCM2837B0, Cortex-A53 64-bit Processors doesn’t support analogue input directly, so those analogue data must be converted into digital levels using ADC ladder network. But this MCU can be replaced with Arduino MCU which will reduce cost of the data transmitter device.

Stage 3: Data Management

Once the conversion of all these sensor data into same digital levels range they are sent into MCU through SPI (Serial Peripheral Interface) one by one with high data transfer rate.

Finally, using mysqlclientlib package in C++ programming that can upload those data into local and remote server database at same time

Stage 4: monitoring and forecasting

Now, raw data which is sensed from power plant is ready in database but it is very difficult to read it as they are in digital levels format. So, decrypting algorithm which translates digital levels into their corresponding dimensions was developed.

To reduce the service frequency and increase this power plant efficiency we installed this new IoT based monitoring and forecasting system.

6.3 Positioning of all sensors in MHPP.

Next step is to position the sensors in SHIRAITO MHPP, before installing to plant a 3D drawing of MHPP was done in Solidworks 3D design and simulation software.

Guide Vane Position: LP-200T linear potentiometer sensor used to detect guide vane position. In Figure 6.12 it is shown how guide vane is adjusted in 3D without installing LP-200T sensor and in Figure 6.13 it was shown how sensor is installed and operated in terms of percentage i.e., 0.1% opened to 99.9% open.

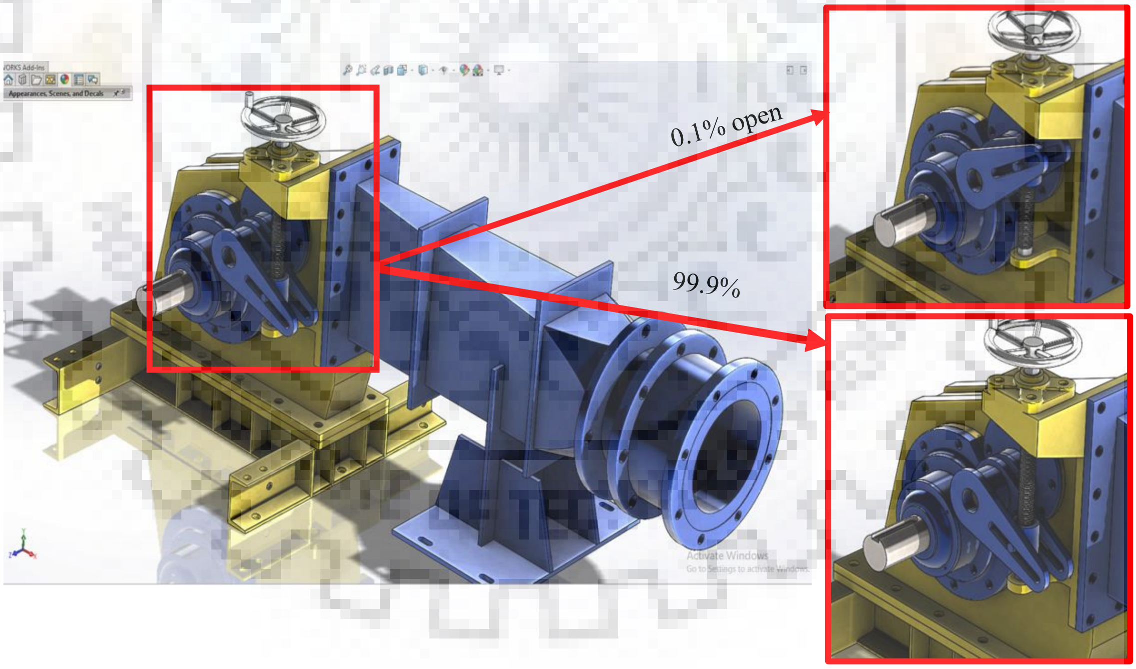


Figure 6.12 3D Modelling of MHPP Guide vane operation using Solidworks

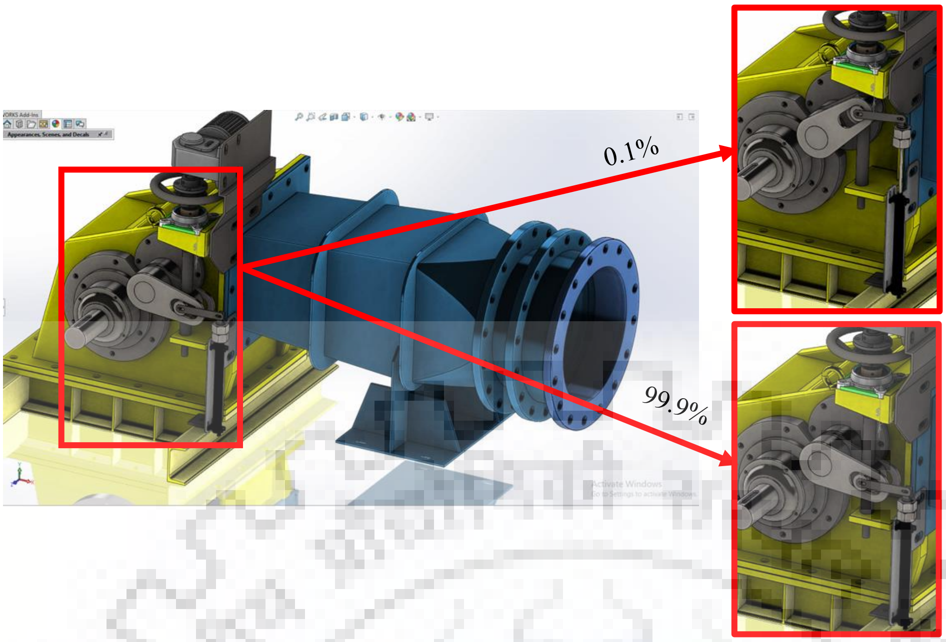


Figure 6.13 LP-200T installed in MHPP Guide

By adding one more degree of freedom to guide vane lever as shown in Figure 6.13, and adding plate to the right of it to fit LP-200T sensor that was fixed guide vane sensor to the existing system. Guide vane operation after installing position sensor is shown in Figure 6.13, at 0.1% opening and 99.9% opening position.

Pressure sensor: GP-M series KEYENCE pressure sensor for sensing water pressure in inlet before cross-flow turbine, GP-M pressure sensor by default its dimension is in Mpa installation is as shown in Figure 6.14. Details like testing and simulation about this sensor was discussed in chapter 4.

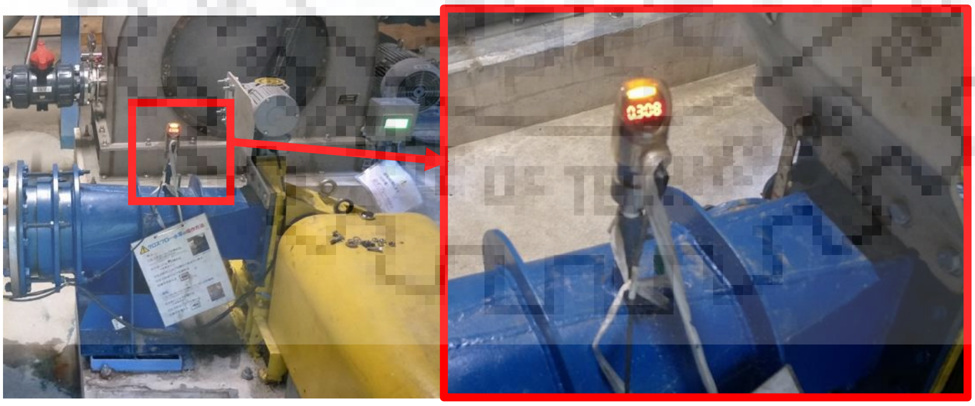


Figure 6.14 GP-M pressure sensor fixed on penstock

Temperature sensor: DHT temperature and humidity sensor is used to measure and control the temperature inside control panel as it was known that ADC module and micro-processor operating temperature are less than 70°C. Positioning of this sensor is placed inside control panel. Details like testing and simulation about this sensor was discussed in chapter 4.

Power measurement device: Sometimes due to high discharge power generated will exceed ratted value, at that time it is required to control guide vane and bring down the power generation below or equal to ratted value. Here this power sensing module is fixed with PLC in control panel. Details like testing and simulation about this sensor was discussed in chapter 4.

6.4 Displaying collected data in monitoring panel

RDS MySQL database [20] is as shown in chapter 4, sensor data are encrypted in digital levels format i.e., values range from 0 to 16777215 as we use analogue to digital convertors. These digital levels are fetched and decrypted using as shown in Figure 6.15. Full algorithm of this system is given in Appendix C.

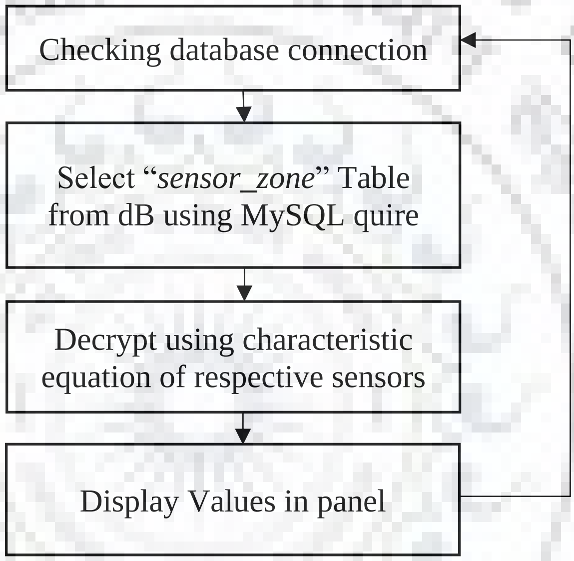


Figure 6.15 Fetching and decrypting data using Php and MySQL

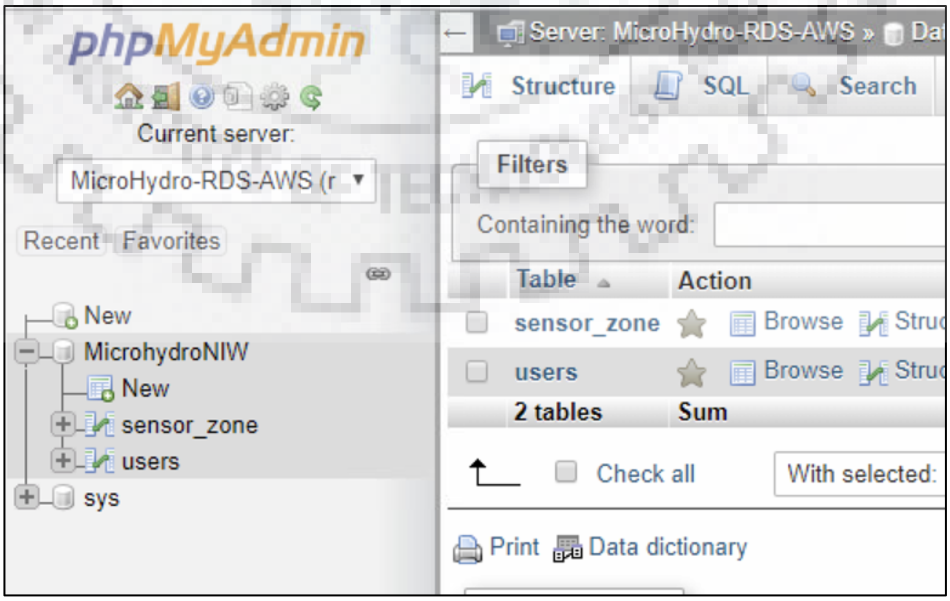


Figure 6.16 RDS Micro Hydro database using "phpmyadmin"

←T→	id ▲ 1	value0	value1	value2	value3	value4	value5	value6	value7	in_time	plant_id
Edit Copy Delete	1	0	1990129668	1995449268	1995650408	1995641392	2130413280	1990093404	2109	2018-11-16 01:16:38	1
Edit Copy Delete	2	0	0	1995449268	1995650408	1995641392	2130413280	1990093404	2109	2018-11-16 01:16:38	1
Edit Copy Delete	3	0	0	0	1995650408	1995641392	2130413280	1990093404	2109	2018-11-16 01:16:38	1
Edit Copy Delete	4	0	0	0	0	1995641392	2130413280	1990093404	2109	2018-11-16 01:16:38	1
Edit Copy Delete	5	0	0	0	0	0	2130413280	1990093404	2109	2018-11-16 01:16:38	1
Edit Copy Delete	6	0	0	0	0	0	0	1990093404	2109	2018-11-16 01:16:38	1
Edit Copy Delete	7	0	0	0	0	0	0	0	2109	2018-11-16 01:16:38	1
Edit Copy Delete	8	0	0	0	0	0	0	0	1991175	2018-11-16 01:16:38	1
Edit Copy Delete	9	1991362	0	0	0	0	0	0	1991175	2018-11-16 01:16:38	1
Edit Copy Delete	10	1991362	0	0	0	0	0	0	1991175	2018-11-16 01:16:38	1
Edit Copy Delete	11	1991362	0	0	0	0	0	0	1991175	2018-11-16 01:16:38	1
Edit Copy Delete	12	1991362	0	0	0	0	0	0	1991175	2018-11-16 01:16:38	1
Edit Copy Delete	13	1991362	0	0	0	0	0	0	1991175	2018-11-16 01:16:38	1
Edit Copy Delete	14	1991362	0	0	0	0	0	0	1991175	2018-11-16 01:16:38	1
Edit Copy Delete	15	1991362	0	0	0	0	0	0	1991175	2018-11-16 01:16:38	1
Edit Copy Delete	16	1991362	0	0	0	0	0	0	1991175	2018-11-16 01:16:38	1
Edit Copy Delete	17	1991362	0	0	0	0	0	0	1991175	2018-11-16 01:16:39	1
Edit Copy Delete	18	1991362	4729900	0	0	0	0	0	1991175	2018-11-16 01:16:39	1

Figure 6.17 Sensor data initialisation in database

From Figure 6.17 it is shown that in first cycle in C++ algorithm was initialize in ADS1256 module details are discussed in chapter 4, this can be noticed that here in this database table from id = 1 to id = 8, in id = 8 all sensors are initialized and ready to measure accurate value. Advantage of doing this is to avoid initial state junk readings in ADC module when device is ideal.

6.4.1 Time series plots of Power Plant parameters

Once all data fetching from database we use this data for analysis purpose by plotting them in time series graph as shown in Figure 6.18.

Time series data:

A time series is a series of data points indexed (or listed or graphed) in time order. Most commonly, a time series is a sequence taken at successive equally spaced points in time. Thus it is a sequence of discrete-time data. 6 hours data with 1 minute frequency was recorded and saved in database further this data will be used for forecasting purpose.

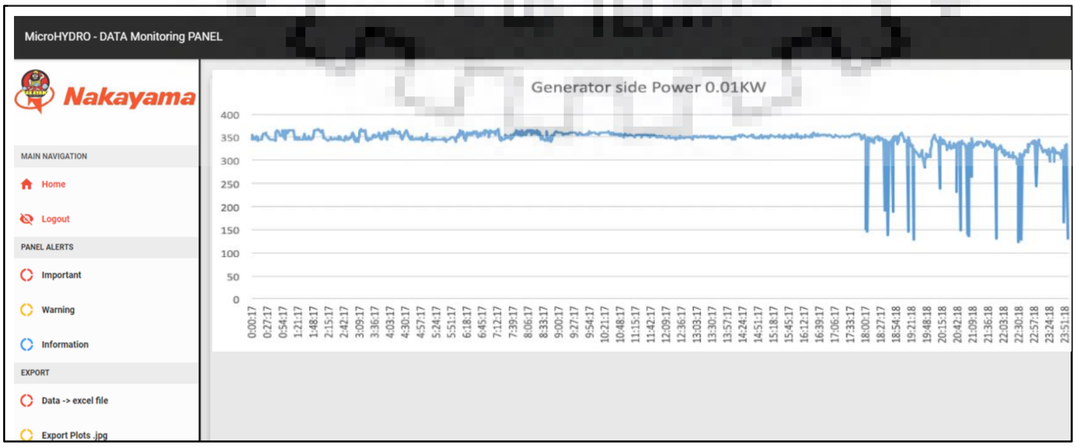


Figure 6.18 Time series plot in monitoring panel for generator side power

Finally, generator side power is being plotted using time series method from data sensed using sensors and fetching them from database. Detailed data is given in Appendix A.

Notification / Alert system: When any parameters move abnormal notification will be shown in monitoring panel. In this thesis was planned to notify user at 90% of abnormal value to save power plant from heavy damage [30]. An example notification result is shown in Figure 6.19.

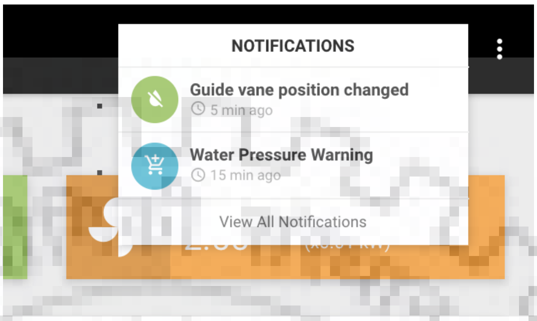


Figure 6.19 Notification / Alert on physical parameters in power plant

So, in this chapter installing of IoT data transmitter to existing MHPP and collecting data for 6h with 1 minute time frequency are shown in Appendix A. In the next discussions below these recorded was used to do forecasting methods and implementation of forecasting to this IoT monitoring system are discussed.

6.5 Forecasting flow results and discussions

The data was collect for duration of 6 hours on 19th November 2018, with every minute frequency. Real time power generation, intake, water pressure, water turbine bearing temperature control panel temperature and guide vane position are the variables taken here, full data table shown in Appendix A. detailed process of forecasting was shown in Figure 6.20.

Whole forecasting part of this report was developed using R-Programming language, as shown in process flow chart initially we clear all the junk data present by default in the workspace, so the following code is executed.

$$> \quad rm(list = ls()) \tag{6.6}$$

Second, it is time to include the forecasting library to the algorithm to use some predefined functions and methods to ease and increase the accuracy of the process of forecasting

$$> \quad library(fpp2) \tag{6.7}$$

Third step is to load the data in .csv that was collect in real time from MHPP, here we load the data shown in Appendix A.

$$> \quad data < - read.csv("/Users/Ram/Desktop/ARIMA/mhpp.csv") \tag{6.8}$$

Original data trend was observed by declaring time series plot on current power generated which was present in 5th column in data table Appendix A.

$$> \quad Y <- ts(data[,5], start = c(1), frequency = 1) \tag{6.9}$$

$$> \quad autoplot(Y) + ggtitle(\text{"}Time\ plot: Power\ generated\text{"}) + ylab(\text{"}Generated\ power\ in\ 0.001Kw\text{"}) \tag{6.10}$$

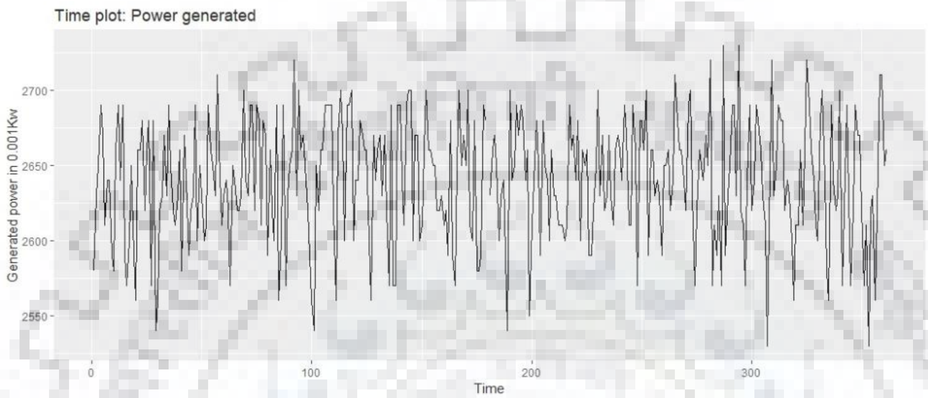


Figure 6.20 Time series plot for currently power generated

Fourth stage is to make the original data to stationary data about zero by differencing it my one using “diff” function and plotting the differenced original data as shown in Figure 6.21.

$$> \quad DY <- diff(Y) \tag{6.11}$$

$$> \quad autoplot(DY) + ggtitle(\text{"}Time\ plot\ Dfifference\ data: Power\ generated\text{"}) + ylab(\text{"}Generated\ power\ in\ 0.001Kw\text{"}) \tag{6.12}$$

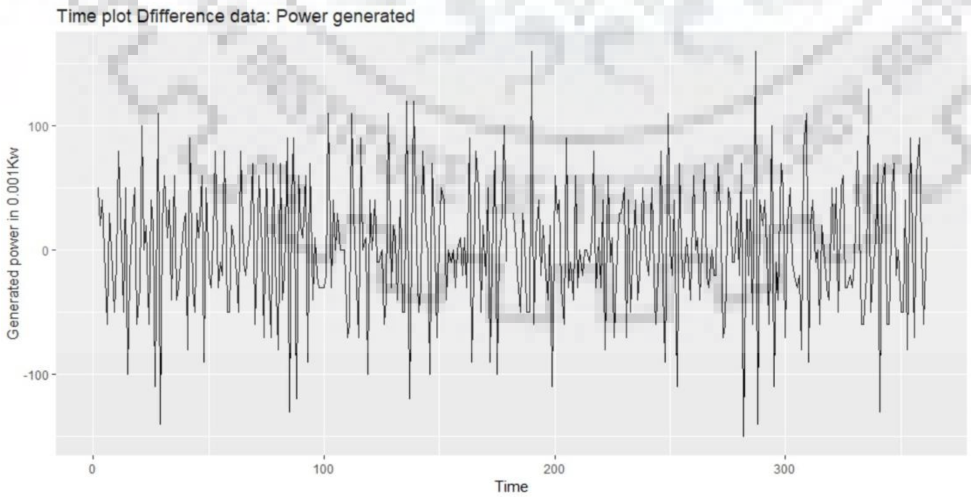


Figure 6.21 Differenced original power generated data.

Forecasting using basic seasonal native method using following set of functions. Residual plots, lag analysis and standard deviations σ1 are recorded here as shown in 6.22.

$$> \quad fit\ < - snaive(DY) \quad (6.8)$$
$$print(summary(fit))$$
$$checkresiduals(fit)$$

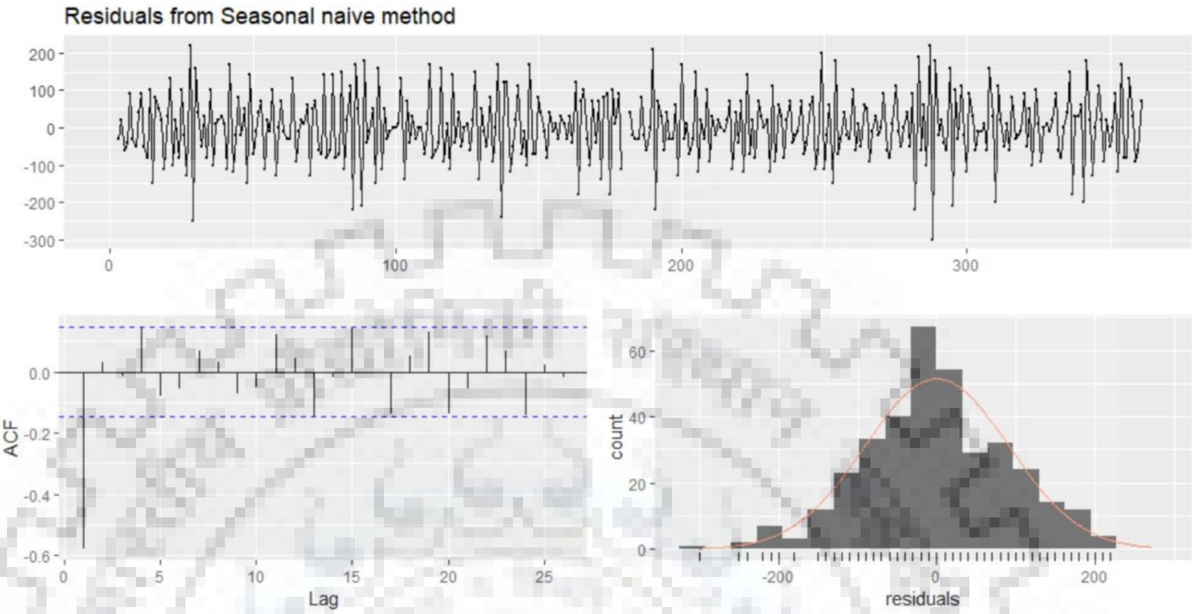


Figure 6.22: Correlograms and residual plots of seasonal native method

```
Forecast method: Seasonal naive method

Model Information:
Call: snaive(y = DY)

Residual sd: 90.8109 ### residual  Standard deviation

Error measures:
                     ME     RMSE     MAE MPE MAPE      MASE
Training set -0.1120448 90.68368 71.2605 NaN  Inf 0.9987693
                   ACF1
Training set -0.5800032

Forecasts:
    Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
362             10 -106.2158 126.2158 -167.7367 187.7367
363             10 -154.3540 174.3540 -241.3577 261.3577
    Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
362             10 -106.2158 126.2158 -167.7367 187.7367
363             10 -154.3540 174.3540 -241.3577 261.3577

data:  Residuals from Seasonal naive method
Q* = 136.31, df = 10, p-value < 2.2e-16

Model df: 0.   Total lags used: 10
```

From the above diagnosis report of seasonal native method’s 80% and 95% confidence limits ar e shown and Standard deviation is recorded as σ1

$$\boldsymbol{\sigma 1 = 90.8109}$$

Second forecasting which was done is Exponential smoothing method using the following function. Residual plots, lag analysis and standard deviations σ2 are recorded here.

$$> \quad fit\_ets < - ets(Y) \tag{6.13}$$

$$print(summary(fit\_ets))$$

$$checkresiduals(fit\_ets)$$

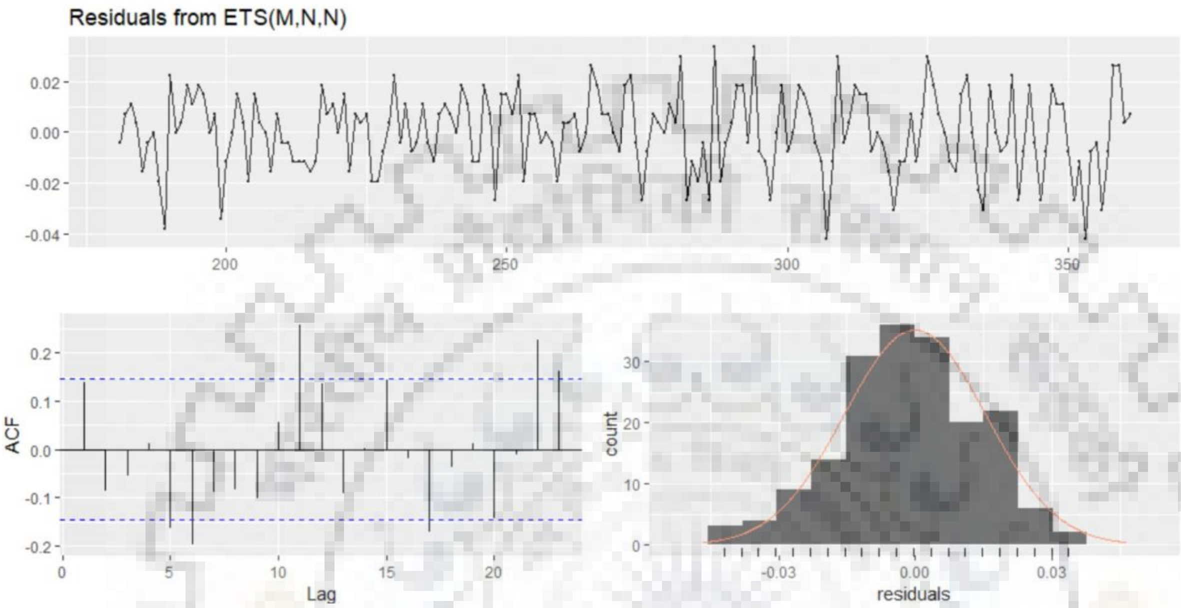


Figure 6.23 Correlograms and residual plots of Exponential smoothing method

```
ETS(M,N,N) ####general form of Exponential smoothing method

Call:
 ets(y = Y)

 Smoothing parameters:
    alpha = 1e-04

 Initial states:  ACF:
    l = 2640.8408  0.1234474

 sigma:  0.0156    ##### residual Standard Deviation

     AIC     AICc      BIC
2290.493 2290.628 2300.088

Training set error measures:
                     ME     RMSE      MAE         MPE     MAPE
Training set -0.0203964 40.91536 32.82594 -0.02488827 1.245206
                  MASE      ACF1
Training set 0.7743995 0.1392294
                     ME     RMSE      MAE         MPE     MAPE
Training set -0.0203964 40.91536 32.82594 -0.02488827 1.245206
                  MASE      ACF1
Training set 0.7743995 0.1392294

data:  Residuals from ETS(M,N,N)
Q* = 23.389, df = 8, p-value = 0.002899

Model df: 2.   Total lags used: 10
```

From the above diagnosis report of Exponential smoothing method's 80% and 95% co nfidence limits are shown and Standard deviation is recorded as σ2 as shown in Figure 6.23.

$$\sigma 2 = 0.0156$$

Third method of forecasting was done using Autoregressive integrative Moving Average method shortly named ARIMA, using the following function. Residual plots, lag analysis and standard deviations σ3 are recorded here as given in Figure 6.24.

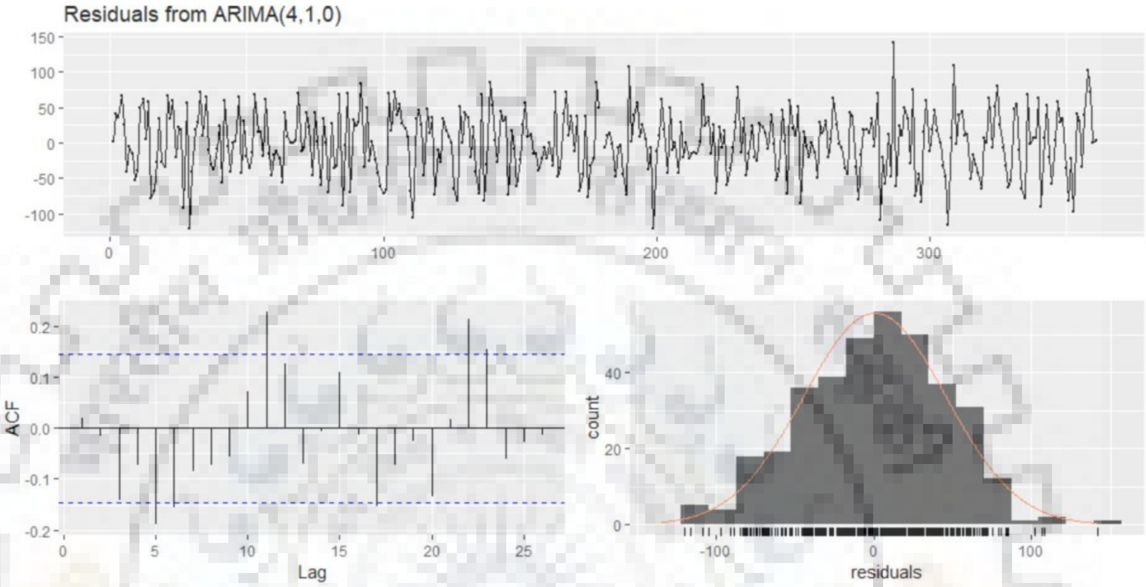


Figure 6.24: Correlograms and residual plots using ARIMA method

```
ARIMA(4,1,0)

Coefficients:
          ar1      ar2      ar3      ar4
      -0.6394  -0.5481  -0.2963  -0.0969
s.e.   0.0528   0.0612   0.0610   0.0530

sigma^2 estimated as 2052:  log likelihood=-1877.27 ACF=0.00080171
AIC=3764.53   AICc=3764.7   BIC=3783.96

Training set error measures:
                   ME     RMSE      MAE          MPE     MAPE
Training set 0.548853 45.04131 36.30923 -0.002712258 1.377469
                  MASE        ACF1
Training set 0.8468212 0.001051715
                   ME     RMSE      MAE          MPE     MAPE
Training set 0.548853 45.04131 36.30923 -0.002712258 1.377469
                  MASE        ACF1
Training set 0.8468212 0.001051715

data:  Residuals from ARIMA(4,1,0)
Q* = 39.332, df = 6, p-value = 6.162e-07

Model df: 4.   Total lags used: 10
```

From the above diagnosis report of ARIMA's 80% and 95% confidence limits are shown and S tandard deviation is recorded as σ3

$$\sigma 3 = 45.299$$

Finally, from above 3 methods the flowing standard deviations are recorded and shown in Table 6.7

Table 6.7: List of forecasting methods simulated and their Standard deviations obtained.

S.no	Forecasting method and Standard deviation
1	Seasonal native method, residual Standard deviation σ1 = 90.8109
2	Exponential Smoothing method, residual Standard deviation σ2 = 0.0156
3	ARIMA method, residual Standard deviation σ3 = 45.299

It is clear from above residual standard deviations Exponential smoothing method has less sigma value but ARIMA method has less ACF value, in this situation ACF is given more priority in decision making. So, ARIMA forecasting using this method for next 15 minutes will be more accurate predictions. Forecasting plots and including the confidence limits are shown in Figure 6.25

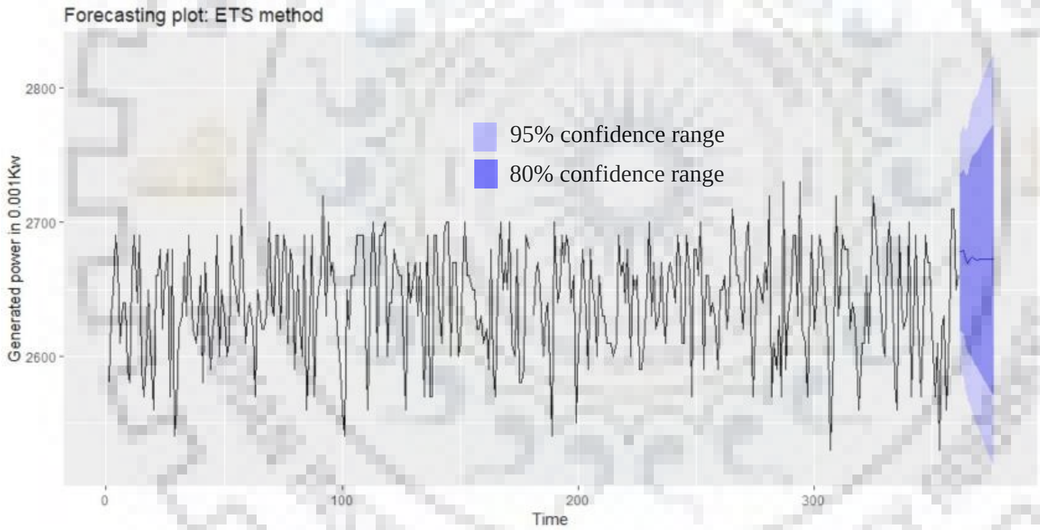


Figure 6.25: Forecast plot using ARIMA method

Comparing forecasting results with original last 15 minutes power generated data and verifying whether all the 15 data are within the 80% confidence limit shown in Figure 6.26. Point wise forecasting data are shown in Appendix A.

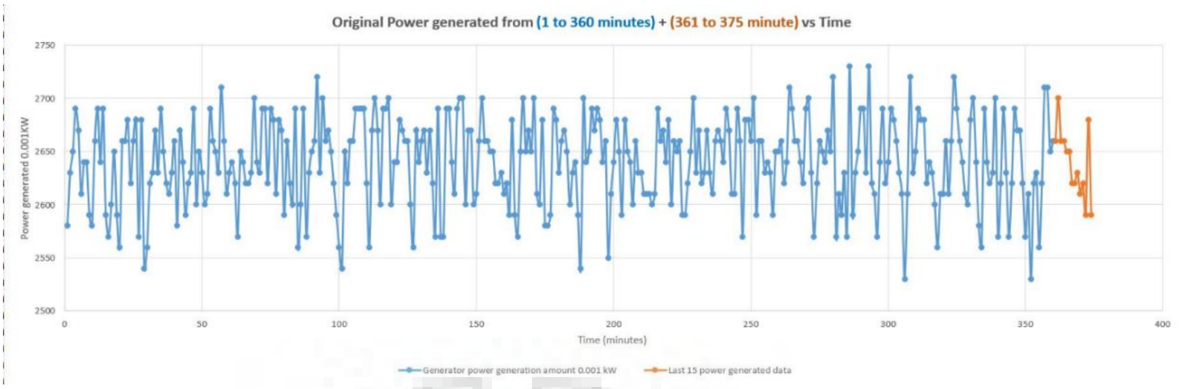


Figure 6.26: Original Power generated plot include last 15 data points.

From the original plot in Figure 6.26 all the data points in last 15 minutes time span are bounded within 80% confidence range of predications in above Figure detailed compassion of forecasted data with original data was shown in Table 6.5 and Appendix A. By designing this forecasting feature into the designed monitoring system, servicing machine parts at right time and inventory management can be done effectively by preventing major damages and economical losses. Finally, monitoring and forecasting system was implemented for existing MHPP.

Table 6.8: Last 15 minutes data to check accuracy in forecasting

S.no (minutes)	DATE	TIME	Generator power generation amount 0.001 kW	Generator power generation amount 0.001 kW [Forecasted Data]
360	19-11-2018	00:01:00	2660	2670.0
361	19-11-2018	00:02:00	2700	2672.0
362	19-11-2018	00:03:00	2660	2670.0
363	19-11-2018	00:04:00	2660	2670.0
367	19-11-2018	00:05:00	2650	2665.0
368	19-11-2018	00:06:00	2650	2665.0
369	19-11-2018	00:07:00	2620	2660.0
370	19-11-2018	00:08:00	2620	2660.0
371	19-11-2018	06:01:00	2630	2665.0
372	19-11-2018	06:03:00	2610	2660.0
373	19-11-2018	06:04:00	2620	2660.0
374	19-11-2018	06:05:00	2690	2671.0
375	19-11-2018	06:06:00	2680	2669.0
376	19-11-2018	06:07:00	2590	2667.0
377	19-11-2018	06:08:00	2660	2670.0

CHAPTER-7
CONCLUSIONS AND RECOMMENDATIONS

7.1 Conclusions

1. A cost effective approach has been introduced and developed in monitoring and forecasting system for hydro power plants using IoT methods and web technologies.
2. In this method multiple data sets from sensors are well organized and auto real-time series plots are also developed in monitoring panel, which will ease the data analysis.
3. An innovative approach, has been presented by solving easy scalable data storage space feature, which is lacking in existing monitoring and forecasting systems and this method also supports wide variety of sensors available in market.
4. The monitoring panel built here was hosted in public domain space using AWS, so this monitoring panel is supported in wide range of devices and operating systems. Additionally, this is globally accessible as the whole system was hosted using HTTP protocol and in Amazon virtual servers.
5. High accurate forecasting models and notification system are implemented, for prevention of machine parts from big damage. Inventory management are done properly by using this forecasting, which results to increase in efficiency of power plant.
6. The data which was store here in data base are encrypted in digital count format so that it is difficult to decrypt for hackers to steal power plant data without knowing the encryption method.

7.2 Recommendations for Future Study

1. Hybrid network connectivity feature can be included in IoT Transmitter module to minimise data loss in time series analysis. Hybrid means using WLAN and GPRS together.
2. ANN and machine learning module can be included to facilitate forecasting and caution predictions in advance and to increase accuracy.
3. This IoT Transmitter module can be generalized so that the same data transmitter can be used even in SPV, Nuclear, Thermal power plants for safe data recordings.

PUBLICATIONS AND AWARDS

Publications:

[1] Y Ram Kumar & M K Singhal., "Monitoring system for Micro Hydro Power Plants using IoT", International Conference on Recent Trends in Electronics, Information & Communication Technology 2019 organized by Sri Venkateshwara college of engineering, Bangalore during May 17th to 18th, 2019, Paper ID-RTEICT 580.

[2] Y Ram Kumar & M K Singhal., "Cost effective modelling of IoT data transmitter and forecasting", International Conference on Energy, Systems and Information Processing 2019 organized by Indian Institute of Information Technology, Chennai. IEEE ICESIP 2019-PES-2346-0731.

Awards:

[1] Y Ram Kumar., "Cost effective IoT data transmitter device design and manufacturing", Won 1st runner up with cash prize of ₹50,000 and representing IIT Roorkee among 100 leading India's engineering colleges participated in, National level CREATECH product development hackathon 2019 organized by Larsen & Toubro. Grand finale was hosted in Madh island Mumbai on February 15th 2019.

REFERENCE

[1] Alves, E.F., Yano, M.L. and Hofmann, M., 2010, November. "Integration of control, protection and supervisory systems in hydro power plants-state of art and trends". In *2010 IEEE/PES Transmission and Distribution Conference and Exposition: Latin America (T&D-LA)* (pp. 285-289). IEEE.

[2] Anees, A.S., 2012, December. "Grid integration of renewable energy sources: Challenges, issues and possible solutions". In *2012 IEEE 5th India International Conference on Power Electronics (IICPE)* (pp. 1-6). IEEE.

[3] Chang, V., Tim, U., Wong, J. and Gadia, S., 2016. "Cloud-based emerging services systems". *International Journal of Information Management*, pp.1-12..

[4] Deb, S., Ghoshal, S., Malepati, V.N., Kleinman, D.L. and Cavanaugh, K.F., "Qualtech Systems Inc, 2004". *Remote diagnosis server*. U.S. Patent 6,795,799.

[5] Fang, X., Misra, S., Xue, G. and Yang, D., 2011. "Smart grid—The new and improved power grid: A survey". *IEEE communications surveys & tutorials*, *14*(4), pp.944-980.

[6] Fleury, A., Vacher, M. and Noury, N., 2009. "SVM-based multimodal classification of activities of daily living in health smart homes: sensors, algorithms, and first experimental results". *IEEE transactions on information technology in biomedicine*, *14*(2), pp.274-283.

[7] Hameed, Z., Ahn, S.H. and Cho, Y.M., 2010. "Practical aspects of a condition monitoring system for a wind turbine with emphasis on its design, system architecture, testing and installation". *Renewable Energy*, *35*(5), pp.879-894.

[8] Handziski, V., Köpke, A., Willig, A. and Wolisz, A., 2006, May. "Twist: a scalable and reconfigurable testbed for wireless indoor experiments with sensor networks". In *Proceedings of the 2nd international workshop on Multi-hop ad hoc networks: from theory to reality* (pp. 63-70).

[9] Jain, M.B., Srinivas, M.B. and Jain, A., 2008, October. "A novel web based expert system architecture for on-line and off-line fault diagnosis and control (FDC) of power system equipment". In *2008 Joint International Conference on Power System Technology and IEEE Power India Conference* (pp. 1-5). IEEE.

[10] Jain, S., Kumar, V., Paventhan, A., Chinnaiyan, V.K., Arnachalam, V. and Pradish, M., 2014, "March. Survey on smart grid technologies-smart metering, IoT and EMS". In *2014 IEEE Students' Conference on Electrical, Electronics and Computer science* (pp. 1-6). IEEE.

[11] Joshi, S. and Boyd, S., 2008. "Sensor selection via convex optimization". *IEEE transactions on Signal Processing*, *57*(2), pp.451-462.

[12] Kavis, M.J., 2014. "*Architecting the cloud: design decisions for cloud computing service models (SaaS, PaaS, and IaaS)"*. John Wiley & Sons.

[13] Kawadia, V. and Kumar, P.R., 2005. "Principles and protocols for power control in wireless ad hoc networks". *IEEE journal on selected areas in communications*, *23*(1), pp.76-88.

[14] Lee, I. and Lee, K., 2015. "The Internet of Things (IoT): Applications, investments, and challenges for enterprises". *Business Horizons*, *58*(4), pp.431-440.

[15] Lidula, N.W.A. and Rajapakse, A.D., 2011. "Microgrids research: A review of experimental microgrids and test systems". *Renewable and Sustainable Energy Reviews*, *15*(1), pp.186-202.

[16] Miller, F.P., Vandome, A.F. and McBrewster, J., 2010. "*Amazon web services"*. Alpha Press.

[17] Mustafa Music., Hasanspahic, N., Bosovic, A., Aganovic, D. and Avdakovic, S., 2016, June. "Upgrading smart meters as key components of integated power quality monitoring system". In *2016 IEEE 16th international conference on environment and electrical engineering (EEEIC)* (pp. 1-6). IEEE.

[18] Nema, P., Nema, R.K. and Rangnekar, S., 2009. "A current and future state of art development of hybrid energy system using wind and PV-solar: A review". *Renewable and Sustainable Energy Reviews*, *13*(8), pp.2096-2103.

[19] Pi–Teach, R., 2017. Learn, and Make with Raspberry Pi; 2018. *Recuperado de: www. raspberrypi. org*.

[20] Poor, H.V., 2006, September. "Signal processing issues in wireless sensor networks". In *2006 14th European Signal Processing Conference* (pp. 1-1). IEEE.

[21] Ramakrishnan, A. and Pecht, M.G., 2003. “A life consumption monitoring methodology for electronic systems”. *IEEE Transactions on Components and Packaging technologies, 26*(3), pp.625-634.

[22] Salvucci, D.D. and Goldberg, J.H., 2000, November. “Identifying fixations and saccades in eye-tracking protocols”. In *Proceedings of the 2000 symposium on Eye tracking research & applications* (pp. 71-78). ACM.

[23] Sauter, T. and Lobashov, M., 2010. “End-to-end communication architecture for smart grids”. *IEEE Transactions on industrial electronics, 58*(4), pp.1218-1228.

[24] Sung, G.N., Wu, C.M., Chen, W.C. and Huang, C.M., 2013. “Design of DC Power Line Communication System Using Time-Division Multiplexing”. In *Applied Mechanics and Materials* (Vol. 431, pp. 325-330). Trans Tech Publications.

[25] Tao, F., Zuo, Y., Da Xu, L. and Zhang, L., 2014. “IoT-based intelligent perception and access of manufacturing resource toward cloud manufacturing”. *IEEE Transactions on Industrial Informatics, 10*(2), pp.1547-1557.

[26] Wang, Q., Lin, Y. and Zhan, H., 2012, May. “A hybrid wireless system for power line monitoring”. In *IEEE PES Innovative Smart Grid Technologies* (pp. 1-6). IEEE.

[27] Xiaohui, Xu, Zhu Peng, Shi Jiafeng, Yang Longyu, and Yong Shaohua. "Research on new monitoring system for important power users in smart distribution network." (2015): 4-4.

[28] Y Ram Kumar and K Selvajyothi “Transformer Monitoring System Using IoT” *ICONDM international conference* 978-1- 5090-2976- 1/16 ©2016 IEEE.

[29] Yoshinobu Watanabe, Yoichi Kiuchi, Mina Takagi, and Tokihiko Fujimoto “Re-build up small scale Hydropower technologies in Japan by local industrial power” *International Conference on Hydropower for Sustainable Development*. Dehradun, India feb 5th -7th 2015 (pp. 267-23).

[30] Zhu, Q., Wang, R., Chen, Q., Liu, Y. and Qin, W., 2010, December. “IoT gateway: Bridgingwireless sensor networks into internet of things”. In *2010 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing* (pp. 347-352). Ieee.

APPENDIX A

Data collected while Sensor network testing for first 6 hours of a day using MATLAB and RDS MySQL Server in Shiraito Fall MHPP, Japan.

Shiraito falls MHPP Sensor data Plant ID = 1									
S.no	DATE	TIME	Currently 0.001 kW	Generator power generation amount 0.001 kW	Intake amount 0.001 m 3 / s	Water pressure 0.001 Mpa	Water turbine bearing temperature 0.1 degree C	Control panel inside temperature 0.1 degree C	Guide vane%
0	19-11-2018	00:01:00	2509	2580	61	838	149	287	24
1	19-11-2018	00:02:00	2510	2630	61	838	150	280	24
2	19-11-2018	00:03:00	2526	2650	61	838	150	273	24
3	19-11-2018	00:04:00	2517	2690	61	838	149	265	24
4	19-11-2018	00:05:00	2514	2670	61	839	150	260	24
5	19-11-2018	00:06:00	2520	2610	61	839	150	254	24
6	19-11-2018	00:07:00	2520	2640	61	838	149	253	24
7	19-11-2018	00:08:00	2526	2640	61	838	149	248	24
8	19-11-2018	00:09:00	2518	2590	61	838	146	275	24
9	19-11-2018	00:10:00	2521	2580	61	838	145	288	24
10	19-11-2018	00:11:00	2511	2660	61	838	143	294	24
11	19-11-2018	00:12:00	2512	2690	61	838	143	298	24
12	19-11-2018	00:13:00	2522	2640	61	838	142	302	24
13	19-11-2018	00:14:00	2508	2690	61	838	142	306	24
14	19-11-2018	00:15:00	2513	2590	61	839	143	307	24
15	19-11-2018	00:16:00	2525	2570	61	839	141	307	24
16	19-11-2018	00:17:00	2528	2600	61	838	142	309	24
17	19-11-2018	00:18:00	2505	2650	61	838	141	311	24
18	19-11-2018	00:19:00	2515	2590	61	838	142	312	24
19	19-11-2018	00:20:00	2514	2560	61	838	142	315	24
20	19-11-2018	00:21:00	2516	2660	61	838	141	316	24
21	19-11-2018	00:22:00	2519	2660	61	838	141	316	24
22	19-11-2018	00:23:00	2519	2680	61	838	141	318	24
23	19-11-2018	00:24:00	2518	2620	61	838	141	319	24
24	19-11-2018	00:25:00	2524	2660	61	838	141	322	24
25	19-11-2018	00:26:00	2514	2680	61	839	141	322	24
26	19-11-2018	00:27:00	2521	2570	61	838	141	322	24
27	19-11-2018	00:28:00	2517	2680	61	839	141	322	24
28	19-11-2018	00:29:00	2513	2540	61	838	140	323	24
29	19-11-2018	00:30:00	2514	2560	61	838	141	324	24
30	19-11-2018	00:31:00	2514	2620	61	838	140	325	24
31	19-11-2018	00:32:00	2510	2630	61	839	139	326	24

32	19-11-2018	00:33:00	2514	2670	61	838	140	326	24
33	19-11-2018	00:34:00	2515	2630	61	838	140	326	24
34	19-11-2018	00:35:00	2519	2690	61	838	140	328	24
35	19-11-2018	00:36:00	2521	2650	61	838	140	327	24
36	19-11-2018	00:37:00	2514	2620	61	838	139	329	24
37	19-11-2018	00:38:00	2524	2610	61	838	140	329	24
38	19-11-2018	00:39:00	2514	2630	61	838	140	329	24
39	19-11-2018	00:40:00	2515	2660	61	839	139	330	24
40	19-11-2018	00:41:00	2514	2580	61	838	139	332	24
41	19-11-2018	00:42:00	2520	2670	61	838	140	331	24
42	19-11-2018	00:43:00	2513	2640	61	838	139	331	24
43	19-11-2018	00:44:00	2519	2590	61	838	139	332	24
44	19-11-2018	00:45:00	2516	2620	61	839	139	332	24
45	19-11-2018	00:46:00	2527	2630	61	839	139	334	24
46	19-11-2018	00:47:00	2514	2690	61	838	140	334	24
47	19-11-2018	00:48:00	2523	2600	61	838	140	335	24
48	19-11-2018	00:49:00	2516	2650	61	838	139	335	24
49	19-11-2018	00:50:00	2519	2630	61	838	140	334	24
50	19-11-2018	00:51:00	2501	2600	60	839	140	336	24
51	19-11-2018	00:52:00	2514	2610	61	838	139	336	24
52	19-11-2018	00:53:00	2504	2690	60	839	138	336	24
53	19-11-2018	00:54:00	2521	2660	61	838	139	337	24
54	19-11-2018	00:55:00	2515	2650	61	838	138	335	24
55	19-11-2018	00:56:00	2512	2630	61	838	139	336	24
56	19-11-2018	00:57:00	2520	2710	61	839	139	337	24
57	19-11-2018	00:58:00	2507	2660	61	838	140	338	24
58	19-11-2018	00:59:00	2521	2610	61	838	139	336	24
59	19-11-2018	01:00:00	2521	2630	61	838	139	337	24
60	19-11-2018	01:01:00	2517	2640	61	838	139	339	24
61	19-11-2018	01:02:00	2518	2620	61	838	139	340	24
62	19-11-2018	01:03:00	2511	2570	61	838	138	340	24
63	19-11-2018	01:04:00	2520	2650	61	838	138	340	24
64	19-11-2018	01:05:00	2512	2640	61	839	138	340	24
65	19-11-2018	01:06:00	2518	2620	61	838	139	340	24
66	19-11-2018	01:07:00	2518	2620	61	838	139	340	24
67	19-11-2018	01:08:00	2516	2630	61	838	139	339	24
68	19-11-2018	01:09:00	2516	2700	61	838	139	341	24
69	19-11-2018	01:10:00	2518	2640	61	839	139	341	24
70	19-11-2018	01:11:00	2514	2630	61	838	138	340	24
71	19-11-2018	01:12:00	2523	2690	61	839	138	341	24
72	19-11-2018	01:13:00	2510	2690	61	839	139	341	24
73	19-11-2018	01:14:00	2520	2620	61	838	138	342	24
74	19-11-2018	01:15:00	2523	2690	61	838	138	341	24
75	19-11-2018	01:16:00	2517	2680	61	839	138	342	24
76	19-11-2018	01:17:00	2510	2610	61	838	137	343	24

77	19-11-2018	01:18:00	2516	2680	61	838	138	343	24
78	19-11-2018	01:19:00	2517	2670	61	838	138	343	24
79	19-11-2018	01:20:00	2502	2590	60	839	138	343	24
80	19-11-2018	01:21:00	2529	2660	61	838	138	344	24
81	19-11-2018	01:22:00	2510	2620	61	838	138	345	24
82	19-11-2018	01:23:00	2516	2600	61	838	138	345	24
83	19-11-2018	01:24:00	2518	2690	61	838	139	345	24
84	19-11-2018	01:25:00	2516	2560	61	838	139	344	24
85	19-11-2018	01:26:00	2516	2600	61	838	138	344	24
86	19-11-2018	01:27:00	2520	2690	61	838	138	344	24
87	19-11-2018	01:28:00	2505	2570	61	838	138	345	24
88	19-11-2018	01:29:00	2508	2630	61	838	138	345	24
89	19-11-2018	01:30:00	2517	2650	61	838	138	345	24
90	19-11-2018	01:31:00	2518	2660	61	839	138	345	24
91	19-11-2018	01:32:00	2515	2720	61	839	138	346	24
92	19-11-2018	01:33:00	2514	2630	61	838	137	345	24
93	19-11-2018	01:34:00	2517	2700	61	838	137	346	24
94	19-11-2018	01:35:00	2499	2660	60	838	137	347	24
95	19-11-2018	01:36:00	2518	2670	61	838	137	345	24
96	19-11-2018	01:37:00	2511	2650	61	838	137	346	24
97	19-11-2018	01:38:00	2506	2620	61	838	138	346	24
98	19-11-2018	01:39:00	2520	2590	61	838	137	347	24
99	19-11-2018	01:40:00	2517	2560	61	839	137	348	24
100	19-11-2018	01:41:00	2508	2540	61	838	137	345	24
101	19-11-2018	01:42:00	2514	2650	61	838	137	347	24
102	19-11-2018	01:43:00	2512	2620	61	838	137	347	24
103	19-11-2018	01:44:00	2516	2660	61	838	137	347	24
104	19-11-2018	01:45:00	2514	2660	61	838	137	346	24
105	19-11-2018	01:46:00	2514	2690	61	838	137	347	24
106	19-11-2018	01:47:00	2516	2690	61	838	137	347	24
107	19-11-2018	01:48:00	2520	2690	61	838	137	346	24
108	19-11-2018	01:49:00	2520	2690	61	838	137	346	24
109	19-11-2018	01:50:00	2511	2620	61	838	138	346	24
110	19-11-2018	01:51:00	2524	2560	61	838	138	348	24
111	19-11-2018	01:52:00	2524	2670	61	838	137	348	24
112	19-11-2018	01:53:00	2518	2700	61	838	138	348	24
113	19-11-2018	01:54:00	2504	2670	60	838	137	347	24
114	19-11-2018	01:55:00	2516	2600	61	838	137	347	24
115	19-11-2018	01:56:00	2525	2690	61	838	137	347	24
116	19-11-2018	01:57:00	2517	2690	61	838	136	348	24
117	19-11-2018	01:58:00	2519	2700	61	838	136	348	24
118	19-11-2018	01:59:00	2528	2600	61	838	137	349	24
119	19-11-2018	02:00:00	2514	2640	61	838	137	349	24
120	19-11-2018	02:01:00	2508	2640	61	838	137	348	24
121	19-11-2018	02:02:00	2521	2680	61	838	136	348	24

122	19-11-2018	02:03:00	2508	2670	61	838	137	348	24
123	19-11-2018	02:04:00	2516	2660	61	838	136	348	24
124	19-11-2018	02:05:00	2518	2660	61	838	136	348	24
125	19-11-2018	02:06:00	2516	2600	61	838	136	348	24
126	19-11-2018	02:07:00	2513	2560	61	838	136	348	24
127	19-11-2018	02:08:00	2516	2670	61	839	136	350	24
128	19-11-2018	02:09:00	2514	2640	61	838	135	349	24
129	19-11-2018	02:10:00	2524	2660	61	838	135	349	24
130	19-11-2018	02:11:00	2505	2670	61	838	136	349	24
131	19-11-2018	02:12:00	2520	2630	61	838	136	349	24
132	19-11-2018	02:13:00	2512	2670	61	838	136	350	24
133	19-11-2018	02:14:00	2520	2620	61	838	136	349	24
134	19-11-2018	02:15:00	2519	2570	61	838	136	349	24
135	19-11-2018	02:16:00	2507	2690	61	838	137	350	24
136	19-11-2018	02:17:00	2521	2570	61	839	137	349	24
137	19-11-2018	02:18:00	2517	2570	61	838	137	350	24
138	19-11-2018	02:19:00	2523	2690	61	838	137	349	24
139	19-11-2018	02:20:00	2517	2690	61	839	137	348	24
140	19-11-2018	02:21:00	2518	2640	61	838	137	345	24
141	19-11-2018	02:22:00	2519	2610	61	838	137	349	24
142	19-11-2018	02:23:00	2521	2690	61	838	137	349	24
143	19-11-2018	02:24:00	2514	2700	61	838	137	348	24
144	19-11-2018	02:25:00	2517	2700	61	838	137	348	24
145	19-11-2018	02:26:00	2522	2600	61	838	137	349	24
146	19-11-2018	02:27:00	2511	2670	61	838	136	350	24
147	19-11-2018	02:28:00	2520	2670	61	838	137	350	24
148	19-11-2018	02:29:00	2519	2600	61	838	136	350	24
149	19-11-2018	02:30:00	2514	2610	61	838	136	348	24
150	19-11-2018	02:31:00	2502	2660	60	838	136	349	24
151	19-11-2018	02:32:00	2517	2700	61	839	136	350	24
152	19-11-2018	02:33:00	2523	2660	61	838	135	349	24
153	19-11-2018	02:34:00	2522	2660	61	838	135	349	24
154	19-11-2018	02:35:00	2521	2650	61	838	135	351	24
155	19-11-2018	02:36:00	2518	2650	61	838	136	348	24
156	19-11-2018	02:37:00	2522	2620	61	838	136	350	24
157	19-11-2018	02:38:00	2514	2620	61	838	136	350	24
158	19-11-2018	02:39:00	2509	2630	61	838	135	349	24
159	19-11-2018	02:40:00	2521	2610	61	838	136	350	24
160	19-11-2018	02:41:00	2517	2620	61	838	136	350	24
161	19-11-2018	02:42:00	2523	2590	61	839	136	350	24
162	19-11-2018	02:43:00	2510	2680	61	838	135	350	24
163	19-11-2018	02:44:00	2521	2590	61	838	135	350	24
164	19-11-2018	02:45:00	2524	2570	61	838	135	350	24
165	19-11-2018	02:46:00	2516	2650	61	838	135	349	24
166	19-11-2018	02:47:00	2513	2700	61	838	135	350	24

167	19-11-2018	02:48:00	2516	2650	61	838	135	350	24
168	19-11-2018	02:49:00	2507	2670	61	838	135	350	24
169	19-11-2018	02:50:00	2508	2650	61	838	135	349	24
170	19-11-2018	02:51:00	2514	2700	61	838	135	349	24
171	19-11-2018	02:52:00	2518	2610	61	839	135	348	24
172	19-11-2018	02:53:00	2508	2600	61	838	135	348	24
173	19-11-2018	02:54:00	2516	2680	61	838	135	349	24
174	19-11-2018	02:55:00	2509	2580	61	838	135	350	24
175	19-11-2018	02:56:00	2507	2580	61	838	135	349	24
176	19-11-2018	02:57:00	2522	2590	61	838	135	349	24
177	19-11-2018	02:58:00	2518	2690	61	838	135	349	24
178	19-11-2018	02:59:00	2514	2680	61	839	135	350	24
179	19-11-2018	03:00:00	2511	2630	61	838	135	350	24
180	19-11-2018	03:01:00	2511	2660	61	838	135	350	24
181	19-11-2018	03:02:00	2517	2670	61	838	135	349	24
182	19-11-2018	03:03:00	2507	2650	61	838	134	349	24
183	19-11-2018	03:04:00	2514	2600	61	838	135	349	24
184	19-11-2018	03:05:00	2515	2630	61	838	135	349	24
185	19-11-2018	03:06:00	2509	2640	61	838	135	349	24
186	19-11-2018	03:07:00	2517	2590	61	838	135	349	24
187	19-11-2018	03:08:00	2512	2540	61	838	134	350	24
188	19-11-2018	03:09:00	2517	2700	61	839	134	347	24
189	19-11-2018	03:10:00	2522	2640	61	838	134	348	24
190	19-11-2018	03:11:00	2514	2650	61	838	134	350	24
191	19-11-2018	03:12:00	2518	2690	61	838	134	348	24
192	19-11-2018	03:13:00	2526	2670	61	838	134	349	24
193	19-11-2018	03:14:00	2510	2690	61	838	134	349	24
194	19-11-2018	03:15:00	2517	2680	61	838	134	348	24
195	19-11-2018	03:16:00	2515	2640	61	838	134	348	24
196	19-11-2018	03:17:00	2521	2660	61	838	134	350	24
197	19-11-2018	03:18:00	2506	2550	61	838	134	348	24
198	19-11-2018	03:19:00	2520	2610	61	838	134	348	24
199	19-11-2018	03:20:00	2521	2640	61	838	134	349	24
200	19-11-2018	03:21:00	2518	2680	61	838	134	350	24
201	19-11-2018	03:22:00	2523	2650	61	838	134	349	24
202	19-11-2018	03:23:00	2520	2590	61	838	134	350	24
203	19-11-2018	03:24:00	2518	2680	61	838	134	350	24
204	19-11-2018	03:25:00	2508	2650	61	838	134	350	24
205	19-11-2018	03:26:00	2519	2640	61	838	133	349	24
206	19-11-2018	03:27:00	2512	2600	61	838	133	349	24
207	19-11-2018	03:28:00	2511	2660	61	838	134	348	24
208	19-11-2018	03:29:00	2512	2630	61	838	133	349	24
209	19-11-2018	03:30:00	2513	2630	61	838	133	347	24
210	19-11-2018	03:31:00	2514	2610	61	838	134	347	24
211	19-11-2018	03:32:00	2516	2610	61	838	134	347	24

212	19-11-2018	03:33:00	2511	2610	61	839	134	350	24
213	19-11-2018	03:34:00	2518	2600	61	838	134	349	24
214	19-11-2018	03:35:00	2512	2610	61	838	134	349	24
215	19-11-2018	03:36:00	2520	2690	61	838	133	350	24
216	19-11-2018	03:37:00	2514	2660	61	838	134	349	24
217	19-11-2018	03:38:00	2514	2670	61	838	134	349	24
218	19-11-2018	03:39:00	2514	2640	61	838	134	349	24
219	19-11-2018	03:40:00	2509	2680	61	838	133	349	24
220	19-11-2018	03:41:00	2512	2600	61	839	134	349	24
221	19-11-2018	03:42:00	2510	2660	61	839	134	348	24
222	19-11-2018	03:43:00	2512	2650	61	838	134	349	24
223	19-11-2018	03:44:00	2522	2660	61	839	134	347	24
224	19-11-2018	03:45:00	2519	2590	61	838	134	346	24
225	19-11-2018	03:46:00	2516	2590	61	838	134	348	24
226	19-11-2018	03:47:00	2522	2620	61	838	134	347	24
227	19-11-2018	03:48:00	2512	2650	61	838	134	348	24
228	19-11-2018	03:49:00	2514	2700	61	838	134	349	24
229	19-11-2018	03:50:00	2515	2630	61	838	134	349	24
230	19-11-2018	03:51:00	2518	2670	61	838	133	349	24
231	19-11-2018	03:52:00	2514	2620	61	838	133	349	24
232	19-11-2018	03:53:00	2520	2630	61	838	134	348	24
233	19-11-2018	03:54:00	2510	2670	61	838	134	348	24
234	19-11-2018	03:55:00	2519	2630	61	838	133	348	24
235	19-11-2018	03:56:00	2507	2610	61	838	133	348	24
236	19-11-2018	03:57:00	2517	2660	61	839	132	348	24
237	19-11-2018	03:58:00	2516	2670	61	838	132	348	24
238	19-11-2018	03:59:00	2505	2660	61	838	132	348	24
239	19-11-2018	04:00:00	2524	2640	61	839	133	348	24
240	19-11-2018	04:01:00	2513	2690	61	838	132	347	24
241	19-11-2018	04:02:00	2516	2670	61	838	133	347	24
242	19-11-2018	04:03:00	2527	2610	61	838	133	347	24
243	19-11-2018	04:04:00	2506	2610	61	838	133	347	24
244	19-11-2018	04:05:00	2519	2690	61	838	133	348	24
245	19-11-2018	04:06:00	2518	2660	61	838	133	348	24
246	19-11-2018	04:07:00	2521	2570	61	838	133	346	24
247	19-11-2018	04:08:00	2519	2680	61	838	133	348	24
248	19-11-2018	04:09:00	2522	2680	61	838	132	348	24
249	19-11-2018	04:10:00	2511	2660	61	838	132	347	24
250	19-11-2018	04:11:00	2510	2700	61	838	132	348	24
251	19-11-2018	04:12:00	2515	2590	61	838	133	348	24
252	19-11-2018	04:13:00	2509	2660	61	838	132	348	24
253	19-11-2018	04:14:00	2526	2660	61	838	132	347	24
254	19-11-2018	04:15:00	2500	2630	60	838	132	348	24
255	19-11-2018	04:16:00	2510	2640	61	838	133	347	24
256	19-11-2018	04:17:00	2514	2630	61	838	133	347	24

257	19-11-2018	04:18:00	2521	2590	61	838	132	347	24
258	19-11-2018	04:19:00	2528	2650	61	839	133	347	24
259	19-11-2018	04:20:00	2514	2650	61	838	133	346	24
260	19-11-2018	04:21:00	2509	2660	61	838	133	347	24
261	19-11-2018	04:22:00	2515	2620	61	839	133	347	24
262	19-11-2018	04:23:00	2525	2640	61	838	133	346	24
263	19-11-2018	04:24:00	2508	2710	61	838	132	347	24
264	19-11-2018	04:25:00	2506	2690	61	838	132	347	24
265	19-11-2018	04:26:00	2512	2660	61	838	132	347	24
266	19-11-2018	04:27:00	2509	2660	61	838	132	347	24
267	19-11-2018	04:28:00	2512	2640	61	838	132	347	24
268	19-11-2018	04:29:00	2514	2620	61	838	132	346	24
269	19-11-2018	04:30:00	2519	2690	61	838	132	346	24
270	19-11-2018	04:31:00	2522	2700	61	838	132	346	24
271	19-11-2018	04:32:00	2517	2630	61	839	132	346	24
272	19-11-2018	04:33:00	2515	2570	61	838	132	346	24
273	19-11-2018	04:34:00	2523	2620	61	838	132	346	24
274	19-11-2018	04:35:00	2522	2660	61	838	132	346	24
275	19-11-2018	04:36:00	2510	2650	61	838	132	347	24
276	19-11-2018	04:37:00	2526	2640	61	838	132	347	24
277	19-11-2018	04:38:00	2516	2670	61	838	132	347	24
278	19-11-2018	04:39:00	2514	2650	61	838	132	347	24
279	19-11-2018	04:40:00	2518	2720	61	838	132	347	24
280	19-11-2018	04:41:00	2516	2570	61	838	132	346	24
281	19-11-2018	04:42:00	2507	2610	61	838	132	346	24
282	19-11-2018	04:43:00	2516	2590	61	838	132	345	24
283	19-11-2018	04:44:00	2523	2630	61	839	132	345	24
284	19-11-2018	04:45:00	2512	2570	61	838	132	346	24
285	19-11-2018	04:46:00	2511	2730	61	838	132	346	24
286	19-11-2018	04:47:00	2520	2590	61	838	132	346	24
287	19-11-2018	04:48:00	2512	2630	61	838	132	346	24
288	19-11-2018	04:49:00	2513	2650	61	838	131	346	24
289	19-11-2018	04:50:00	2518	2690	61	838	132	346	24
290	19-11-2018	04:51:00	2518	2690	61	838	132	346	24
291	19-11-2018	04:52:00	2511	2630	61	838	131	346	24
292	19-11-2018	04:53:00	2513	2730	61	838	131	346	24
293	19-11-2018	04:54:00	2522	2620	61	838	132	345	24
294	19-11-2018	04:55:00	2518	2610	61	838	132	346	24
295	19-11-2018	04:56:00	2513	2570	61	838	132	346	24
296	19-11-2018	04:57:00	2518	2640	61	838	132	346	24
297	19-11-2018	04:58:00	2511	2690	61	838	132	346	24
298	19-11-2018	04:59:00	2522	2620	61	838	132	346	24
299	19-11-2018	05:00:00	2520	2640	61	838	131	344	24
300	19-11-2018	05:01:00	2520	2690	61	838	131	343	24
301	19-11-2018	05:02:00	2520	2680	61	838	131	345	24

302	19-11-2018	05:03:00	2511	2660	61	838	131	345	24
303	19-11-2018	05:04:00	2516	2630	61	838	131	343	24
304	19-11-2018	05:05:00	2519	2610	61	838	131	343	24
305	19-11-2018	05:06:00	2515	2530	61	838	131	346	24
306	19-11-2018	05:07:00	2520	2610	61	838	131	345	24
307	19-11-2018	05:08:00	2510	2720	61	838	132	344	24
308	19-11-2018	05:09:00	2520	2630	61	838	132	345	24
309	19-11-2018	05:10:00	2510	2650	61	838	132	345	24
310	19-11-2018	05:11:00	2516	2690	61	838	131	345	24
311	19-11-2018	05:12:00	2511	2680	61	838	131	345	24
312	19-11-2018	05:13:00	2515	2680	61	838	131	345	24
313	19-11-2018	05:14:00	2513	2620	61	838	131	345	24
314	19-11-2018	05:15:00	2514	2640	61	838	131	345	24
315	19-11-2018	05:16:00	2511	2630	61	838	131	344	24
316	19-11-2018	05:17:00	2514	2600	61	838	131	344	24
317	19-11-2018	05:18:00	2510	2560	61	838	131	343	24
318	19-11-2018	05:19:00	2509	2610	61	838	131	345	24
319	19-11-2018	05:20:00	2502	2610	60	838	131	345	24
320	19-11-2018	05:21:00	2517	2660	61	838	132	344	24
321	19-11-2018	05:22:00	2522	2610	61	838	132	343	24
322	19-11-2018	05:23:00	2520	2660	61	838	131	344	24
323	19-11-2018	05:24:00	2506	2720	61	838	131	343	24
324	19-11-2018	05:25:00	2517	2690	61	838	131	344	24
325	19-11-2018	05:26:00	2513	2660	61	838	131	345	24
326	19-11-2018	05:27:00	2516	2640	61	838	131	344	24
327	19-11-2018	05:28:00	2512	2610	61	838	131	343	24
328	19-11-2018	05:29:00	2512	2600	61	838	131	344	24
329	19-11-2018	05:30:00	2509	2680	61	838	131	344	24
330	19-11-2018	05:31:00	2509	2700	61	838	131	344	24
331	19-11-2018	05:32:00	2514	2640	61	838	131	344	24
332	19-11-2018	05:33:00	2516	2580	61	838	131	344	24
333	19-11-2018	05:34:00	2515	2560	61	838	131	344	24
334	19-11-2018	05:35:00	2512	2690	61	838	131	344	24
335	19-11-2018	05:36:00	2508	2640	61	838	131	343	24
336	19-11-2018	05:37:00	2506	2620	61	838	131	344	24
337	19-11-2018	05:38:00	2509	2630	61	838	131	342	24
338	19-11-2018	05:39:00	2517	2700	61	838	131	343	24
339	19-11-2018	05:40:00	2504	2570	60	838	130	343	24
340	19-11-2018	05:41:00	2518	2620	61	838	130	343	24
341	19-11-2018	05:42:00	2519	2690	61	838	130	342	24
342	19-11-2018	05:43:00	2504	2630	60	838	130	344	24
343	19-11-2018	05:44:00	2514	2570	61	838	130	344	24
344	19-11-2018	05:45:00	2515	2620	61	838	130	344	24
345	19-11-2018	05:46:00	2515	2690	61	838	130	344	24
346	19-11-2018	05:47:00	2507	2670	61	838	130	343	24

347	19-11-2018	05:48:00	2520	2670	61	838	130	343	24
348	19-11-2018	05:49:00	2516	2620	61	838	130	343	24
349	19-11-2018	05:50:00	2514	2570	61	838	130	343	24
350	19-11-2018	05:51:00	2511	2610	61	838	130	343	24
351	19-11-2018	05:52:00	2511	2530	61	838	130	344	24
352	19-11-2018	05:53:00	2509	2620	61	838	130	342	24
353	19-11-2018	05:54:00	2509	2630	61	838	130	342	24
354	19-11-2018	05:55:00	2512	2560	61	838	130	343	24
355	19-11-2018	05:56:00	2501	2620	60	838	130	342	24
356	19-11-2018	05:57:00	2510	2710	61	838	130	342	24
357	19-11-2018	05:58:00	2511	2710	61	838	129	342	24
358	19-11-2018	05:59:00	2518	2650	61	838	130	342	24
359	19-11-2018	06:00:00	2518	2660	61	838	129	342	24

Last 15 minutes data to check accuracy in forecasting

S.no	DATE	TIME	Generator power generation amount 0.001 kW	Generator power generation amount 0.001 kW [Forecasted Data]
360	19-11-2018	00:01:00	2660	2670.0
361	19-11-2018	00:02:00	2700	2672.0
362	19-11-2018	00:03:00	2660	2670.0
363	19-11-2018	00:04:00	2660	2670.0
367	19-11-2018	00:05:00	2650	2665.0
368	19-11-2018	00:06:00	2650	2665.0
369	19-11-2018	00:07:00	2620	2660.0
370	19-11-2018	00:08:00	2620	2660.0
371	19-11-2018	06:01:00	2630	2665.0
372	19-11-2018	06:03:00	2610	2660.0
373	19-11-2018	06:04:00	2620	2660.0
374	19-11-2018	06:05:00	2690	2671.0
375	19-11-2018	06:06:00	2680	2669.0
376	19-11-2018	06:07:00	2590	2667.0
377	19-11-2018	06:08:00	2660	2670.0

APPENDIX B

Microcontroller programming for IoT Data Transmitter device using Embedded C language.

```
#include <SoftwareSerial.h>

/* Create object named SIM900 of the class SoftwareSerial */
SoftwareSerial SIM900(9, 10);

const int analog1 = A0;
const int analog2 = A1;
/*********************************************************************
Name: Initial settings function, runs only once in beginning
Function: setup ()
*********************************************************************/
void setup() {
pinMode(A0, INPUT);          /* Analog input pin 1*/
pinMode(A1, INPUT);          /* Analog input pin 2*/

SIM900.begin(9600);  /* Define baud rate for software serial communication */
Serial.begin(9600);          /* Define baud rate for serial communication */
Serial.println("System Installation Program initiated:");
Serial.print("AT\\r\\n");
SIM900.println("AT");          /* Check Communication */
delay(5000);
ShowSerialData();              /* Print response on the serial monitor */
delay(1000);

Serial.print("AT+CMGF=1\\r\\n"); SIM900.println("AT+CMGF=1");        /*
Check Communication */

delay(5000); ShowSerialData();        /* Print response on the serial monitor */
delay(1000);

Serial.print("AT+CMGS="+917358225508"\\r\\n");
SIM900.println("AT+CMGS="+917358225508""); /* Check Communication */
delay(5000);
ShowSerialData();     /* Print response on the serial monitor */
delay(1000);

Serial.print("Monitoring system's assigned unique device ID to register in
monitoring panel is: 1\\r\\n");
SIM900.println("Monitoring system's assigned unique device ID to register in
monitoring panel is: 1");     /* Check Communication */
delay(5000);
ShowSerialData();       /* Print response on the serial monitor */
delay(1000);

Serial.print("AT+CMGF=1\\r\\n");
SIM900.println("AT+CMGF=1");      /* Check Communication */
delay(5000);
ShowSerialData();     /* Print response on the serial monitor */
```

```
delay(1000);

Serial.print("AT+CMGS="+917358225508"\\r\\n");
SIM900.println("AT+CMGS="+917358225508""); /* Check Communication */
delay(5000);
ShowSerialData();        /* Print response on the serial monitor */
delay(1000);

Serial.print("Congrats! System initiation is successfull\\r\\n");
SIM900.println("Congrats! System initiation is successfull");          /* Check Communication */
delay(5000);
ShowSerialData();        /* Print response on the serial monitor */
delay(1000);

Serial.print("AT+CMGF=1\\r\\n");
SIM900.println("AT+CMGF=1");      /* Check Communication */
delay(5000);
ShowSerialData();        /* Print response on the serial monitor */
delay(1000);

Serial.print("AT+CMGS="+917358225508"\\r\\n");
SIM900.println("AT+CMGS="+917358225508""); /* Check Communication */
delay(5000);
ShowSerialData();       /* Print response on the serial monitor */
delay(1000);

Serial.print("please use http://phoenix.ncss.asia/indexmain.php to access monitoring panel. note: Use PC for best experience\\r\\n");
SIM900.println("please use http://phoenix.ncss.asia/indexmain.php to access monitoring panel. note: Use PC for best experience");        /* Check Communication */
delay(5000);
ShowSerialData();        /* Print response on the serial monitor */
delay(1000);
}
/*****************************************************************
Name: Continues loop
Function: loop ()
*****************************************************************/
void loop() {
   int p_one= analogRead(analog1);
   int p_two= analogRead(analog2);

   Serial.println("HTTP get method :");
   Serial.print("AT\\r\\n");
   SIM900.println("AT");        /* Check Communication */
   delay(5000);
   ShowSerialData();   /* Print response on the serial monitor */
   delay(1000);

   Serial.print("AT+SAPBR=3,1,\"CONTYPE\",\"GPRS\"\\r\\n");
   SIM900.println("AT+SAPBR=3,1,\"CONTYPE\",\"GPRS\"");
```

```
delay(5000);
ShowSerialData();
delay(500);
Serial.print("AT+SAPBR=3,1,\"APN\",\"airtelgprs.com\"\\r\\n");
SIM900.println("AT+SAPBR=3,1,\"APN\",\"airtelgprs.com\"");  /* APN of the provider */
delay(5000);
ShowSerialData();
delay(500);
Serial.print("AT+SAPBR=1,1\\r\\n");
SIM900.println("AT+SAPBR=1,1"); /* Open GPRS context */
delay(5000);
ShowSerialData();
delay(500);
Serial.print("AT+SAPBR=2,1\\r\\n");
SIM900.println("AT+SAPBR=2,1"); /* Query the GPRS context */
delay(5000);
ShowSerialData();
delay(500);
Serial.print("AT+HTTPINIT\\r\\n");
SIM900.println("AT+HTTPINIT"); /* Initialize HTTP service */
delay(5000);
ShowSerialData();
delay(500);
Serial.print("AT+HTTPPARA=\"CID\",1\\r\\n");
SIM900.println("AT+HTTPPARA=\"CID\",1");  /* Set parameters for HTTP session */
delay(5000);
ShowSerialData();
delay(500);

String phoenix = "AT+HTTPPARA=\"URL\",\"phoenix.ncss.asia/index2.php?pone=";
String phoenix1 = "&ptwo=";
Serial.print("AT+HTTPPARA=\"URL\",\"phoenix.ncss.asia/index2.php?pone=");
Serial.print(p_one);
Serial.print("&ptwo=");
Serial.print(p_two);
Serial.print("\"");
Serial.print("\\r\\n");

SIM900.println(phoenix + phoenix1 + p_one + p_two"\"");  /* Set parameters for HTTP session */

delay(5000);
ShowSerialData();
delay(500);

Serial.print("AT+HTTPACTION=0\\r\\n");
SIM900.println("AT+HTTPACTION=0");  /* Start GET session */
delay(10000);
```

```
    ShowSerialData();
    delay(500);

    Serial.print("AT+HTTPTERM\\r\\n");
    SIM900.println("AT+HTTPTERM");  /* Terminate HTTP service */
    delay(5000);
    ShowSerialData();
    delay(500);
    Serial.print("AT+SAPBR=0,1\\r\\n");
    SIM900.println("AT+SAPBR=0,1"); /* Close GPRS context */
    delay(5000);
    ShowSerialData();
    delay(500);
}

void ShowSerialData()
{
    while(SIM900.available()!=0)  /* If data is available on serial port */
    Serial.write(char (SIM900.read())); /* Print character received on to the serial monitor */
}
```

APPENDIX C

Raspberry Pi Microprocessor programming for ADC Data Conversion and Transmitting data to public server RDS database using C++ language.

```
#include <bcm2835.h>
#include <stdio.h>
#include <unistd.h>
#include <string.h>
#include <math.h>
#include <errno.h>
#include <stdlib.h>
#include <mysql/mysql.h>
MYSQL mysql,*connection;
MYSQL_RES result;
MYSQL_ROW row;
char * ip = (char*)"microhydroniw.c4sty2dpq6yv.ap-northeast-1.rds.amazonaws.com";
char * usr = (char*)"ramkumarNIW";
char * pass = (char*)"Ramkumar1994";
char * db = (char*)"MicrohydroNIW";
int plantID = 1;

/*****************************************************************
Name: ADS1256_StartScan
Function: Configuration DRDY PIN for external interrupt is triggered
*****************************************************************/
void ADS1256_StartScan(uint8_t _ucScanMode)
{
        g_tADS1256.ScanMode = _ucScanMode;
        {
                uint8_t i;
                g_tADS1256.Channel = 0;
                for (i = 0; i < 8; i++)
                {
                        g_tADS1256.AdcNow[i] = 0;
                }
        }
}

/*****************************************************************
Name: ADS1256_Send8Bit
Function: SPI bus to send 8 bit data
*****************************************************************/
static void ADS1256_Send8Bit(uint8_t _data)
{
        bsp_DelayUS(2);
        bcm2835_spi_transfer(_data);
}
```

```
/****************************************************************
Name: ADS1256_DelayDATA
Function: delay
****************************************************************/
static void ADS1256_DelayDATA(void)
{
        bsp_DelayUS(10);      /* The minimum time delay 6.5us */
}

/****************************************************************
Name: main
Function:  main
****************************************************************/
int  main()
{
        uint8_t id;
        int32_t adc[8];
        int32_t volt[8];
        uint8_t i;
        uint8_t ch_num;
        int32_t iTemp;
        uint8_t buf[3];
        if (!bcm2835_init())
        return 1;
        bcm2835_spi_begin();

        id = ADS1256_ReadChipID();
        mysql_init(&mysql);
        connection = mysql_real_connect(&mysql, ip, usr, pass, db, 0, NULL,
0);
if (connection==NULL)
{
        printf(mysql_error(&mysql));
        printf("\n");
        printf("connection null");
}else{
        printf("connection successfull");
}
printf("\r\n");
printf("ID=\r\n");
if (id != 3)
{
         printf("Error, ASD1256 Chip ID = 0x%d\r\n", (int)id);
}
else
{
        printf("Ok, ASD1256 Chip ID = 0x%d\r\n", (int)id);
}
ADS1256_CfgADC(ADS1256_GAIN_1, ADS1256_15SPS);
ADS1256_StartScan(0);
ch_num = 8;
```

```
	while(1)
	{
		while((ADS1256_Scan() == 0));
		for (i = 0; i < ch_num; i++)
		{
			adc[i] = ADS1256_GetAdc(i);
			int sensor0 = adc[0];
	int sensor1 = adc[1];
	int sensor2 = adc[2];
	int sensor3 = adc[3];
	int sensor4 = adc[4];
	int sensor5 = adc[5];
	int sensor6 = adc[6];
	int sensor7 = adc[7];

	char queryy[500];

	sprintf(queryy, "INSERT INTO sensor_zone
	(value0,value1,value2,value3,value4,value5,value6,value7,plant_id) VALUES
	(%d,%d,%d,%d,%d,%d,%d,%d,%d)",sensor0,sensor1,sensor2,sensor3,sensor4,
	sensor5,sensor6,sensor7,plantID);
	if (mysql_query(&mysql, queryy) != 0)
	{
		printf("Query Failure\n");
		printf("%d",i);printf("endl");
	}
		volt[i] = (adc[i] * 100) / 167;
	}

	for (i = 0; i < ch_num; i++)
	{
		buf[0] = ((uint32_t)adc[i] >> 16) & 0xFF;
		buf[1] = ((uint32_t)adc[i] >> 8) & 0xFF;
		buf[2] = ((uint32_t)adc[i] >> 0) & 0xFF;
		printf("%d=%02X%02X%02X, %8ld", (int)i, (int)buf[0],
		(int)buf[1], (int)buf[2], (long)adc[i]);
		iTemp = volt[i];

	if (iTemp < 0)
	{
		iTemp = -iTemp;
		printf(" (-%ld.%03ld %03ld V) \r\n", iTemp /1000000,
	(iTemp%1000000)/1000, iTemp%1000);
	}
	else
	{
		printf(" ( %ld.%03ld %03ld V) \r\n", iTemp /1000000,
	(iTemp%1000000)/1000, iTemp%1000);
	}
	}
		printf("%d", (int)ch_num);
```

```
            bsp_DelayUS(100000);        /* 0.1 seconds delay */
    }
            bcm2835_spi_end();
            bcm2835_close();

    return 0;
    }
```

APPENDIX D

IoT Transmitter Simulation results for 1 while (1) loop cycle in debugging mode observed through UART Serial monitor for IoT Data Transmitter.

1. 02:36:46.287 -> System Installation Program initiated:
2. 02:36:46.326 -> AT\r\nAT
3. 02:36:51.297 ->
4. 02:36:51.297 -> OK
5. 02:36:52.289 -> AT+CMGF=1\r\nAT+CMGF=1
6. 02:36:57.323 ->
7. 02:36:57.323 -> OK
8. 02:36:58.311 -> AT+CMGS="+917358225508"\r\nAT+CMGS="+917358225508"
9. 02:37:03.385 ->
10. 02:37:03.385 -> > Monitoring system's assigned unique device ID to register in monitoring panel is: 1\r\nMonitoring system's assigned unique device ID to register in mo
11. 02:37:15.480 -> +CMGS: 145
12. 02:37:15.480 ->
13. 02:37:15.480 -> OK
14. 02:37:16.468 -> AT+CMGF=1\r\nAT+CMGF=1
15. 02:37:21.490 ->
16. 02:37:21.490 -> OK
17. 02:37:22.479 -> AT+CMGS="+917358225508"\r\nAT+CMGS="+917358225508"
18. 02:37:27.537 ->
19. 02:37:27.537 -> > Congrats! System initiation is successfull\r\nCongrats! System initiation is successfull
20. 02:37:33.627 ->
21. 02:37:33.627 -> >
22. 02:37:39.594 -> +CMGS: 146
23. 02:37:39.594 ->
24. 02:37:39.594 -> OK
25. 02:37:40.582 -> AT+CMGF=1\r\nAT+CMGF=1
26. 02:37:45.593 ->
27. 02:37:45.593 -> OK
28. 02:37:46.610 -> AT+CMGS="+917358225508"\r\nAT+CMGS="+917358225508"
29. 02:37:51.631 ->
30. 02:37:51.671 -> > please use http://phoenix.ncss.asia/indexmain.php to access mon
31. 02:38:03.758 -> +CMGS: 147
32. 02:38:03.758 ->
33. 02:38:03.758 -> OK
34. 02:38:04.751 -> HTTP get method :
35. 02:38:04.790 -> AT\r\nAT
36. 02:38:09.776 ->
37. 02:38:09.776 -> OK

38. 02:38:10.766 -> AT+SAPBR=3,1,"CONTYPE","GPRS"\r\nAT+SAPBR=3,1,"CONTYPE","GPRS"
39. 02:38:15.850 ->
40. 02:38:15.850 -> OK
41. 02:38:16.324 -> AT+SAPBR=3,1,"APN","airtelgprs.com"\r\nAT+SAPBR=3,1,"APN","airtelgprs.com"
42. 02:38:21.382 ->
43. 02:38:21.382 -> OK
44. 02:38:21.853 -> AT+SAPBR=1,1\r\nAT+SAPBR=1,1
45. 02:38:26.881 ->
46. 02:38:26.881 -> ERROR
47. 02:38:27.355 -> AT+SAPBR=2,1\r\nAT+SAPBR=2,1
48. 02:38:32.402 ->
49. 02:38:32.402 -> +SAPBR: 1,1,"100.72.19.242"
50. 02:38:32.442 ->
51. 02:38:32.442 -> OK
52. 02:38:32.879 -> AT+HTTPINIT\r\nAT+HTTPINIT
53. 02:38:37.905 ->
54. 02:38:37.905 -> ERROR
55. 02:38:38.414 -> AT+HTTPPARA="CID",1\r\nAT+HTTPPARA="CID",1
56. 02:38:43.471 ->
57. 02:38:43.471 -> OK
58. 02:38:43.945 -> AT+HTTPPARA="URL","phoenix.ncss.asia/index2.php?pone=235"\r\n235667AT+HTTPPARA="URL","phoenix.ncss.asia/index2.php?pone=235&ptwo=6AT+HTTPACTION=0\r\nAT+HTTPACTION=0
59. 02:38:59.578 ->
60. 02:38:59.578 -> OK
61. 02:38:59.578 ->
62. 02:38:59.578 -> +HTTPACTION:0,200,43
63. 02:39:00.053 -> AT+HTTPTERM\r\nAT+HTTPTERM
64. 02:39:05.090 ->
65. 02:39:05.090 -> OK
66. 02:39:05.564 -> AT+SAPBR=0,1\r\nAT+SAPBR=0,1
67. 02:39:10.610 ->
68. 02:39:10.610 -> OK

APPENDIX E

Schematic of IoT Data Transmitter device using Multisim and Ultiboard

APPENDIX F

Auto Regressive Integrative Moving Average (ARIMA) forecasting algorithm written in R-Programming language by taking 6 hours real time data collected from MHPP.

```
###########
#forecasting generated power for 6h time series data
###########

###########
#clear all variables in workspace
###########
rm(list=ls())

# Load forcasting packages
library(fpp2)

#load the data
data <- read.csv("/Users/Anuhya Ram/Desktop/ARIMA/mhpp.csv")

#Declare time series data
Y <- ts(data[,5],start = c(1), frequency = 1)

#preliminary analysis

#time plot
autoplot(Y) + ggtitle("Time plot: Power generated") +
ylab("Generated power in 0.001Kw")

#trend investigations by taking first difference
DY <- diff(Y)

#time plot of diff data
autoplot(DY) + ggtitle("Time plot Dfifference data: Power generated") +
ylab("Generated power in 0.001Kw")
# Series is converted to stationarynow,  forcasting using various models

###########
#1 Seasonal native as out Benchmark method Residual SD= 90.8109
# y_t =y_(t-s) + e_t
###########
fit <- snaive(DY)
print(summary(fit))
checkresiduals(fit)

###########
#2 ETS Exponential smooting models Residual SD= 0.0156
###########
fit_ets <- ets(Y)
print(summary(fit_ets))
checkresiduals(fit_ets)
```

```
#3 Fit ARIMA model Residual SD= 45.299
fit_arima <- auto.arima(Y,d=1,D=1, approximation = FALSE, trace = FALSE)
print(summary(fit_arima))
checkresiduals(fit_arima)

###########
#Finally Forcasting Here
###########
fcst <- forecast(fit_arima,h=15)
autoplot(fcst)+ ggtitle("Forecasting plot: ETS method") +
ylab("Generated power in 0.001Kw")
```

APPENDIX G

Calculations for DHT22 series sensor with burst signal from output port in 40 bit single bus data format as shown in equation A.

0000 0010	1001 0010	0000 0001	0000 1101	1010 0010	(A)
High humidity	Low humidity	High temp	Low temp	Parity bit	

Calculations:

Humidity $[0000\ 0010\ 1001\ 0010]_2 = [0292H]_{16}$ (Hexadecimal) = 2×256 + 9×16 + 2 = $[658]_{10}$

Humidity = $[65.8]_{10}$% RH

Temperature $[0000\ 0001\ 0000\ 1101]_2 = [10DH]_{16}$ (Hexadecimal) = 1×256 + 0×16 + 13 = $[269]_{10}$

Temperature = $[26.9]_{10}$℃

When the temperature is below 0oC calculation is as following statements we know $[-10.1]_{10}$ ℃ Expressed as $[1\ 000\ 0000\ 0110\ 0101]_2$

Temp $[0000\ 0000\ 0110\ 0101]_2 = [0065H]_{16}$ (Hexadecimal) ＝ 6×16 +5 = $[101]_{10}$

Temp. ＝ $[-10.1]_{10}$℃

To check the data received is correct we use the following calculation and compare with parity bit.

$[0000\ 0010]_2 + [1001\ 0010]_2 + [0000\ 0001]_2 + [0000\ 1101]_2 = [1010\ 0010]_2$ （Parity bit）
So, received data is correct.