

**IoT Based Water Level Sanitary
System for aquaculture application
Using Nodemcu**

A Project Report Phase-II

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Electronics and Communication Engineering

By

**VIJAY BALAJI S (39130513)
TONY MAURIZIO (39130542)**



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
SCHOOL OF ELECTRONICS**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

**Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHISALAI,
CHENNAI - 600119**

APRIL - 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with —All grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Final Year Project Report is the bonafide work of VIJAY BALAJI.S (39130513), TONY MAURIZIO (39130542) who have done the project work as an individual who carried out the project entitled “**IoT Based Water Level Sanitary System for aquaculture application Using Nodemcu**” under my supervision from Jan 2023 to April 2023.

Internal Guide

Dr. G. RAJALAKSHMI, M.E, Ph.D.

Head of the Department

Dr. T. RAVI, M.E., Ph.D.

Submitted for Viva voce Examination held on 26-04-2023

Internal Examiner

External Examiner

DECLARATION

We, **VIJAY BALAJI.S (39130513)** and **TONY MAURIZIO (39130542)** hereby declares that the Project Report on “**IoT Based Water Level Sanitary System for aquaculture application Using Nodemcu**” done by us under the guidance of **Dr. G. RAJALAKSHMI, M.E, Ph.D.** at Sathyabama Institute of Science and technology is submitted in partial fulfilment of the requirements of the award of Bachelor of Engineering degree in **Electronics and Communication Engineering**.

DATE: 26-04-2023

PLACE: Chennai

SIGNATURE OF THECANDIDATE

1.



2



ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N.M. NANDHITHA M.E., Ph. D, Dean**, School of Electronics, **Dr. T. RAVI M.E., Ph.D.**, Head of the Department of Electronics and Communication Engineering for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Dr. G. RAJALAKSHMI, M.E, Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of our phase-2 project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the **Department of Electronics and Communication Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Aquaculture is a multibillion-dollar industry that meets the demand for marine food while enhancing sustainability. Farmers in this area encounter a finite number of issues. Farmer's traditional methods for measuring water quality are less efficient and time consuming. To address the concerns related to pH, dissolved oxygen, and turbidity, a system is developed that would allow farmers to monitor water quality in aquafarms in real time. This project work presents a thorough illustration of how the system has been designed and developed. To test water quality in real time, a pH meter is introduced, TDS, and dissolved oxygen sensors connected to Nodemcu in various liquids and the observation and results are recorded. The sensors were calibrated to achieve the precise values in the system thus designed. The results were then displayed on a webpage once the system connected to the internet.

TABLE OF CONTENTS

Chapter No	TITLES	PAGE NO
	Abstract	v
	List of Figures	viii
1	INTRODUCTION	1
	1.1 History of Aquaculture	1
2	LITERATURE REVIEW	4
3	EXISTING SYSTEM	12
4	PROJECT DESCRIPTION	14
	4.1 PROBLEM STATEMENT	14
	4.2 PROPOSED MODE	14
	4.3 IOT (INTERNET OF THINGS)	15
	4.3.1 About Internet of Things (IoT)	15
	4.3.2 Features of IOT	17
	4.4 SYSTEM ARCHITECTURE	18
	4.4.1 Sensors	18
	4.4.2 Nodemcu	18
	4.4.3 Temperature Sensor	21

4.4.4	Water level Sensor	25
4.4.5	pH sensor	27
4.4.6	LCD Display	30
4.4.7	Buzzer	34
5	SOFTWARE REQUIREMENTS	36
5.1	Arduino uno and its programming	36
5.2	Arduino Software (IDE)	37
5.3	Language Support	44
6	CONCLUSION	46
6.1	Future scope	46
6.2	Research issue	47
6.3	Implement issues	48
6.4	Results	49
	REFERENCES	51
	Appendix	54

List of Figures

Figure No.	Figure Name	Page No.
4.1	Block diagram of system	15
4.2	IOT (Internet of things)	16
4.3	IOT Architecture	17
4.4	NodeMCU	20
4.5	Thermocouple	22
4.6	Temperature sensor	24
4.7	Water level sensor	26
4.8	Water level sensor pinout	27
4.9	pH sensor	28
4.10	Water quality management	29
4.11	LCD display	32
4.12	Block Diagram of LCD	33
4.13	Buzzer	34
5.1	Arduino IDE	37
5.2	Language Support	44
6.1	Graph showing water level of the system	49
6.2	Graph showing temperature of the system	50
6.3	Graph showing pH level of the system	50

CHAPTER 1

INTRODUCTION

Aquaculture is a type of agriculture in which aquatic animals and plants are propagated, grown, and sold in a controlled environment. Seafood is one of the most widely consumed foods worldwide, and traditional marine fishing is unable to meet the rising demand, so many nations have turned to aquaculture, which involves rearing aquatic animals and plants in a controlled environment. World fish farming was first practiced in China around 2000 B.C. Aquaculture in Europe began sometime during the Middle Ages. The quality of the farm's water determines its success or failure to a large extent. Suspended solids, temperature, dissolved gases, pH, mineral content, and the potential danger of toxic metals must all be taken into account in these artificially created environments. Among the many water quality characteristics, temperature, dissolved oxygen, ammonia, pH, and alkalinity will be important. There are several methods for determining these parameters are time-consuming and inefficient. It is difficult to keep track of these parameters regularly. An IoT-based smart water quality monitoring system can solve all of these issues. There are several existing solutions to this problem but, our approach to the problem is straightforward, and the system designed will continuously monitor water quality and automatically alert the farmer when there is a significant change in water quality parameters. This has the potential to boost productivity. When the user is in a remote location, the web interface can keep him up to date on the status of the water quality.

1.1 History of Aquaculture

The earliest evidence of fish farming dates back to before 1000 BCE in China. The Zhou dynasty (1112-221 BCE), then the politician Fan Li around 500 BCE, were the first to describe carp, a symbol of good luck and fortune, as being farmed for food. During the Tang dynasty, around 618, the emperor Li, whose name means 'carp', forbade farming the fish that bore his name. Farmers then turned their attention to similar fish in the *Cyprinidae* family and developed the first form of polyculture. Liquid manure from livestock farming was also used to stimulate algae growth in ponds and make it more nutritious. The pond beds were then drained and used as fertiliser. The first integrated agriculture and aquaculture systems emerged in China, where still implemented today.

In Europe, aquaculture first began in ancient Rome. The Romans, who loved sea fish and oysters, created oyster farms and adopted the Assyrian vivarium, a kind of 'swimming pool' where fish and crustaceans caught in lagoons were kept alive until it was time to eat them. These vivaria were built inside wealthier homes, where guests eat the fish.

In the Middle Ages, throughout feudal Europe, the monastic orders and the aristocracy were the main users of freshwater fish vivaria, since then many monopoly over the land, forests, and water courses. Mussel farming was invented in the 13th century, and the technique remained largely unchanged until the 1960s. As with hunting, poaching was severely punished, and the less well-off would have to wait a few centuries before fresh fish was served on their plates.

Freshwater fish farming was further developed during the Renaissance. Several treatises were published, providing details on pond construction and management techniques, the choice of species to farm, their diseases, and their diet. Carp dominated the artificial ponds of Eastern Europe. Emperor Charles IV ordered many such ponds to be built in Bohemia, which is now the westernmost region of the Czech Republic.

Artificial breeding was discovered in Germany during the Enlightenment, but it was not until the 19th century, an era of rapid industrialization, that anyone paid much attention to it. In a hundred years, industry changed the European landscape. Pollution caused fish populations to diminish, and dams and irrigation canals obstructed the migratory paths of some species, such as salmon. To combat this dramatic decline, artificial breeding research focused on trout farming, and researchers managed to master all stages of the process, from fertilisation to egg storage and transportation, pond farming, and releasing fish into the wild. Hatcheries appeared all over the West, and, in the 1860s, trout and other salmonids colonised rivers around the world, including in the United States, India, New Zealand, and even Japan, one of the first producers of edible seaweed.

During the first five decades of the 20th century, colonists introduced and then farmed other species of fish in the Anglo-Belgian colonies in Africa, whether for leisure fishing, to prevent the spread of malaria (using insect-eating species), or as a food source (tilapia, for example). In the kibbutzim of Israel, farmers adapted traditional methods imported from Eastern Europe to the arid environment and developed new techniques, enabling them to achieve self-sufficiency in fish products.

In the late 1950s, the invention of artificially granulated food revolutionised fish farming, which until then relied on products from agriculture and livestock farming (raw meat, for example) to feed the fish.

During the 1970s, marine species aquaculture enjoyed a revival, thanks to new, lighter, more hard-wearing, and less expensive building materials (fibreglass, plastic tubes), and the use of floating cages rather than expensive glass and cast-iron saltwater ponds. However, these new facilities turned out to be commercially unviable, and the optimisation and stabilisation of marine fish production became a major concern in the following decade. At the start of the 21st century, aquaculture took on great importance worldwide. According to a report on fishing and aquaculture by the Food and Agriculture Organisation of the United Nations (FAO) in 2016, "In terms of global production volume, that of farmed fish and aquatic plants combined surpassed that of capture fisheries in 2013".

CHAPTER 2

LITERATURE REVIEW

Hong-Jun Zhu, (2010) has discussed on the Wireless sensors network (WSN) and Wireless personal area network (WPAN) rapid development and popularization, Driven sensing and transmission of information environment has become faster and more convenient. ZigBee short-range wireless transmission technologies, the use of standardized IEEE802.15.4 standard, features include low-power, low-rate, low-cost ... and so on. ZigBee is applied to a considerable number of occasions, Dae-Man Han and Jae-Hyun Lim, (2010) as the main signal transmission using ZigBee technology, Consumers with a variety of physical sensors and home equipment, set up a smart home energy management system to handle a variety of situations that may occur at home. This article applies ZigBee aquaculture systems, when combined sensor and fish farming equipment, and design their own ZigBee transmission path, enabling wireless transfer of information, the information is not easily blocked foreign body, resulting in signal loss. Production of fisheries, advancement and future food needs are firmly related.

Changhui Deng et.al has revealed how the aquatic life will impact because of progress in water quality parameters and how IoT is utilized to overcome the issue. A great deal of research work is done with IoT to take care of this kind of issues as recently IoT is achieving the ground level with its application to agriculturists [3], [4].

S.Kayalvizhi, Koushik Reddy et.al focuses on few kind sensors like pH, DO, Turbidity [5],[6], [7] and so forth and a solution for those issues. Be that as it may, the optimum fish production is absolutely subject to numerous chemical, physical and biological characteristics of water to the vast majority of the degree. Thus, effective pond management requires a realization of water quality. Water quality is determined by factors like Dissolved Oxygen (DO), temperature, turbidity, transparency, water colour, pH, carbon dioxide, alkalinity, hardness, conductivity, salinity, TDS, unionized ammonia, nitrate, nitrite, primary productivity, plankton population, BOD, etc. [8].

K. Raghu Sita Rama Raju and G. Harish Kumar Varma (2017) performed a work entitled as "Knowledge Based Real Time Monitoring System for Aquaculture Using IoT" which uses several sensors such as Dissolved Oxygen, Temperature, Ammonia,

Salt, pH, Nitrate and Carbonates [9]. But maintaining lots of sensors is costly and tedious. So, a system is needed which is not much costly and can determine the overall quality of the water effectively. This is the point which is the base of our research.

Delincé, Guy on Temperature pronouncedly affects biological and chemical procedures. Rates of biological and chemical responses double for each 10°C increment in temperature in general. Temperature significantly influences chemical treatments. Fish have poor resistance to sudden changes in temperature. Often, a quick change in temperature of as low as 5°C will stress or even slaughter fish [10]. pH, DO, conductivity, salinity etc. are directly dependent on temperature [8], [10], [11], [12]. So, temperature should be in the expected range first before checking other parameters. General threshold range for temperature is 21°C-33°C [9] which can be maintained easily. For these reasons, consider temperature as our first working parameter.

Raju, K. Raghu Sita Rama et.al on pH is a ration of the hydrogen ion concentration and designates whether the water is acidic or basic in reaction. Phytoplankton and other marine plant life eliminate carbon dioxide from the water during photosynthesis, so the pH water body increases during the day and drops during the night. Waters with low aggregate alkalinity regularly have pH estimations of 6 to 7.5 preceding sunrise, however when phytoplankton development is substantial, at evening pH esteems may ascend to 10 or significantly higher [10]. The pH of natural waters is significantly impacted by the convergence of carbon dioxide which is an acidic gas [8]. pH changes in pond water are for the most part affected via carbon dioxide and ions in harmony with it. Control of pH is necessary for diminishing ammonia and H₂S poisonousness. The pH is directly or indirectly related to many other parameters and controlling it is comparatively easier. That's why pH is our second consideration.

Salinity is characterized as the aggregate concentration of electrically charged ions (anions – HCO₃⁻, CO₃⁻, SO₄⁻, Cl⁻; cations – Na⁺, K⁺, Ca⁺⁺, Mg⁺⁺ and other constituents such as NH₄⁺, NO₃⁻ and PO₄⁻). Salinity is a noteworthy driving element that influences the density and development of aquatic being's populace [8]. By and by, one is rarely able to measure concentration of all ions in water. A conductivity sensor can be used to quantify conductivity and estimate the approximate salinity and there is relationship between conductivity and TDS [10]. Conductivity of water relies

upon its ionic fixation and varieties of dissolved solids. So, it is enough to measure only conductivity instead of measuring TDS and other ions individually. Salinity is our third consideration for these.

The color of the water gives a sign of what kind of turbidity it is. If it is greenish, it is due to plankton and if it is brown, it is often due to clay [12]. If water is clear, it indicates low biological production - not sufficiently fertile and fish won't develop well in it. Muddy water is not good fish culture because fish can have their gills obstructed by the clay particles and this can bring about death. Dark green water demonstrates overgeneration of planktons that are served as food for fish yet happen because of utilization of more than sufficient composts, excrement or supplement rich sustains to a pond. Bluish green/ brown greenish, green color of water indicates good plankton population thus, useful for fish wellbeing [8].

Bhatnagar A, Jana SN et.al DO is one of the most important factors in aquaculture, but during measuring DO decreases as the temperature and salinity (conductivity) increase and increases as the temperature and salinity (conductivity) decrease. Again, it also fluctuates in a similar fashion to pH level [11], [12]. So that if temperature, pH and conductivity are balanced, DO will also be balanced. Assume, the condition of DO in water from its colour such as greenish water color implies that there are sufficient DO as adequate amount of DO develops phytoplankton which makes the water greenish.

Petersen, A., J. S. Andersen et.al start Conventional methods were adopted for monitoring the water quality, the samples of the water would be taken from the water and sent to the chemical laboratory for analyzing hazardous material. The drawback in the system is all the processing were being executed manually like measurements, maintenance and controlling of the system etc. On the other hand, manual system was time consuming. Some of the previous studies discussed the water monitoring models like culture models and forecasting models and integrated models [13], [14] but these systems had not ability of online monitoring, real time communication data collection. So that these models are not good for fish monitoring but, IoT is one of the fast-growing technologies of recent decades. The purpose of IoT is identifying, monitoring, tracking, locating the things [15] telecommunication and interconnectivity between devices. The invention of new sensors technologies, wireless

telecommunication technology, data transmission technology, many devices are made for real time monitoring in remote areas [16].

Zhu, X., D. Li, D. He, J. Wang, D. Ma, and F. Li on the growth of aquatic life is affected by the water quality variation which have been discussed in many studies. Mostly studies used a lot of sensors but management of these sensors are costly but most papers concentrate on few forms of sensors such as PH, DO, Temperature, Turbidity, water level, [and suggested the solutions of the problems because all parameters are not nursery to monitor these parameters (PH, DO, Temperature, Turbidity, water level) are dependent on each other if one parameter unstable cause of other variable unstable. That's the reason for choosing these parameters.

Proposed architecture uses different sensors (Temperature, PH water level, Turbidity, Motion detection). These sensors are configured with Arduino Uno for sensing and observing measurements in aquatic environment. The previous proposed approaches by authors used cloud database [17] for the storage of output data which makes the architecture costly by maximum consumption of internet, to overcome this consumption Computer system acts like a server host to compute and manage the output values generated through sensors are easily managed and the necessary data could be retrieved by user by consuming minimum cost of internet. Local database helps the former for analytics and take pro-active measures when its needed. Most of models using focus on sending the sensor data but in our model, that provide proper solution if parameter crosses the limits and also discuss the factor that effect the fish growth. Our main focus is reducing internet consumption. In the field areas internet is main problem is to reducing internet consumption also using GSM modem for sending the message if internet is not available, and also using Android application, desktop application for the formers.

Binti Hasim, Ramalingam, Erna wan & Puviarasi [17] undertook research entitled "Developing Fish Feeder System using Raspberry Pi". No longer will the fish owner have to be concerned about feeding the fish at home while they are away. This eliminates the risk of the fish dying, under/overfeeding the fish and also maintain the quality of the water in the fish tank. As a result, this system was important. This research proposed to design a food feeding system using a micro-controller - a Raspberry Pi - as well as web application. The owner could set the time for dispensing the fish food. This system was also designed to prevent the death of fish due to

malfunction.

Meethongjan & Kongsong [18] undertook research into the smart fish farming system by using the IoT and mobile application technology. It presented a mobile application system which worked with the IoT and Android application to control the level of water in a fish pond. This system used Android studio, Java, C, Arduino IDE, SQL and Firebase software. The controller system was from Node ESP8266 MCU V.2 board, Wemos-D1 boards and module ultrasonic hardware. This farming system was experimented on using a small fish pond in a house. Data was collected in order to determine the efficiency of the system. Black Box Testing was used to evaluate the system. Average and standard deviation was used for data analysis purposes. The results of the evaluation on the part of both feeders and experts were shown as 4.21, 4.15 and 0.74, 0.72 respectively. As a result, this system could be used for feeding purposes and it could be the model for the other smart farmers.

Akila, Karthikeyan, Hari & Hari [19] developed a domestic fish food feeding system by using the IoT (IoT-Based Domestic Fish Feeder). Raspberry pi was the main module and a step motor was used to dispense the food. A Pi-camera was used to see the fish in real time. There was a program to manage the system involving control of such items as an automatic feeder, manual feed, and the live streaming of fish. It was possible to set the time of feeding using an automatic feeder controlled by the program. Fish food could also be dispensed from a distance by using a web-interface. It was possible to see, capture, record using video and downloading such a video showing the fish. This research was beneficial to the feeders in that it operated perfectly. However, there were 2 mistakes: the readiness of the internet and the power supply which needed to work 24/7. Otherwise, the overall system would fail. This system was easy to repair.

In 2012, teacher of K.L University outlined monitoring framework with the assistance of Zigbee Module and ARM7 controller to control constant aquaculture ecological factors and achieve the issues raised. Daudi S. Simbeye and Shi Feng Yang gave the outline of water quality monitoring and control framework for aquaculture in view of remote sensor systems and single chip PC innovation as a base in the genuine activity. It understands monitoring of the water natural parameters for escalated aquaculture and caution, Zhu Wang Qi Wang, Xiao Qiang Hao examined the issue of the manual explanatory technique received in water quality location with awful ongoing character

and presented a novel sort of remote water quality estimating and monitoring framework in view of WSN. Shruti Sridharan et al. tended to in their venture about building up a productive remote sensor organize (WSN) based water quality monitoring framework, which looks at water quality.

Geetha and Gouthami [5] undertook research entitled “Internet of Things enabled real time water quality monitoring system”. It proposed: 1. to survey the latest research with regard to the measurement of the quality of water using ICT and various kinds of sensor, 2. to present a system for measuring water system quality on a low budget and using Wi-Fi as the controller for measuring parameters such as pH, despondency and electric conductivity. This included notification for the change with regard to the parameter in the quality of the water. 5 parameters were experimented with in the form of electric conductivity, pH, despondency, temperature and water levels. This experiment was connected to the Ubidots platform. The sensors were put in 3 places around the researchers’ university. This system could send SMS, emails and notifications to the feeders if the parameters were out of balance. This research presented the details of the equipment and technique used to measure the quality of water, using a low budget quality of water evaluation system. This system would cause the sensor to provide online information to the feeders. This experiment indicated the suitability of the system for measuring for quality of water

Buated and Junmark [22] undertook research into “Automatic catfish farming in a cement pond”. The purposes of this research were 1. to decrease the problems of feeding and waste water, 2. to control the temperature in the fish pond in the winter and 3. to apply a micro-controller to control the automatic catfish farming in a cement pond. This fish farming system consisted of a cement pond, a fish food feeding machine, an oxygen generator, a heater, a micro-controller and a solar cell system.

The cement pond was circular with a diameter of 100 cm. and a height of 50 cm. The food feeding machine was made of 3 mm. acrylic. It also had a DC motor which was the main controller for dispensing food and for measuring the level of food in 2 levels such as being full or empty. The controller consisted of a micro-controller, a switch, a feeding fish food kit, a water level sensor, a buffer, a water pump system, an oxygen pump, an electric valve, a heater and a food level sensor. The microcontroller would calculate the input then put the output to buffer in order to increase the pressure from the micro-controller to 5V. After that, the circuit would work. The energy system

consisted of a solar panel, a battery and a charger. The result in terms of automatic catfish farming using a cement pond was that the fish food supplied was less than the demand by about 17-20%. The water temperature level experiment found that when the heater was turned on, the temperature increased from 18 to 20 centigrade degree. If it operated for more than 15 minutes, the temperature would be higher. As a result, the water level temperature kit could not work because the heat transferred for a spark plug was not sufficient. Water transfer worked automatically according to the timer. When the water was released up to the 2nd level, the valve would be closed. The pump would then add water up to the 3rd level and the pump stopped. The oxygen pump worked according to the program setting.

Mourvika S, Monika A, Jyoti B, Trupti M system is composed of Sensors, Communication modules, Power source and Microcontroller. The microcontroller to be used in this experimentation is AtMega328 embedded on Arduino Nano Development board. The microcontroller is responsible for controlling all the communication and data transfer processes as well as control of sensors [23]. A 170-point breadboard is used for extending power and ground connections. The sensors to be used include Water Temperature sensor, pH sensor, Ammonium and Nitrite ion sensors, Dissolved Oxygen sensor, Electrical Conductivity sensor and Redox Potential sensor. The sensors mentioned are used for measuring the most crucial parameters corresponding to sustenance of aquaculture fish ponds or shrimp farms. Any fluctuation in these factors affect the mortality rate of the organisms and hence it is necessary to constantly monitor these water quality parameters in aquaculture ponds and farms. A GPRS/GSM (sim800L) module, Magnetometer (HMC588L) and a GPS module (U-Blox Neo-6M) is connected to the Arduino board. The GPRS/GSM module alerts the farmer by sending a SMS to their mobile numbers. The SMS will include details such as the specifications of the parameter which has fluctuated beyond permissible range as well as the GPS location of the point on the pond where the fluctuation was measured. The GPS module is preprogrammed with certain pin points around the pond where the System is to perform monitoring of water quality parameters and collect the data. The chassis of the system represents the typical shape of a speed boat which is pentagonal in shape in order to accommodate swift movement through the water. The parameters analyzed during each interval along

with the time, place and date of analysis will be stored in the form of text file in the Micro SD card connected to the Arduino board using a MicroSD Card Module for reading and writing to the SD card. In the event of failure of GSMmodule to send the parameter data and warnings (if any) at that particular interval of analysis, the data will be sent to a separate text file in the microSD card. Once connection is back online, the data stored in that text file will be read by the Arduino board and sent to GSM module to resend the SMS with those parameters [24].

CHAPTER 3

EXISTING SYSTEM

The system concerns about enhancing development in aquaculture by improved information and communication processes. Need to extend the ability of organizations to meet the needs of its farmers. An IOT based on architecture in is to check SCM of aquaculture product. Uses GSM & GPRS as remote sensor for checking framework, which could be a mandatory piece. The main goal of this research is for the farmers to have an E-Aquaculture. As the quality of water is a censorious factor for culturing marine & fresh water organisms, it is observed repeatedly for better growth and survival. The sensors used are arranged in the water to sense the information which is later transferred. The Raspberry Pi is used in which has inbuilt Wi-Fi module in it. For calculating a systematized solar harvesting system many factors are required for the potential ease of energy harvesting. The Solar energy transformation and battery energy storage can produce enough electricity for the experimental systems to power sensor nodes by pumping continuously to increase the oxygen of fishponds efficiently. The authors have chosen minimum essential product approach to establish the product as quickly as possible. The objective is to give access at low cost for the measurements of the water quality, to test the prototype with their robustness in African weather conditions, to gather information about the fish pond and with poor internet connection the installment of low-cost gateway to be tested. The Aquaponics is a combination of Aquaculture and hydroponics. The main aim is to produce organic food from the waste which is been produced by the aquatic creature. The system embraces continuous tracking of the water quality and the surrounding environment by utilizing different sensors. The data from sensors can be retrieved remotely by using the IOT platform. The purpose of this work is to give an automatic fish farming monitoring system as its saving time, lowering cost and electricity for the aqua farmer. In the fish farming process, several sensors like pH, temperature and water level sensors are used. By utilizing these sensors, all the work is automated and it will be easy to observe the fish farming remotely from any location. The WSN plays an important role in system with great potential for improving current applications in intensive aquaculture. In ZigBee is being used due to its low cost and low power consumption property. Based

on WSN the system involves the design of monitoring the water quality and a control system. In future, the work would gain remote access by using internet. This paper presents a prototype and idea of a distributed system with some necessary variables in aquaculture. The proposed system works by observing the quality of the water based on WSN and IOT. The main goal is to design and implement a system for aquaculture to track remotely the values of dissolved oxygen, pH and temperature. This prototype is still under development for future use.

DISADVANTAGES

Fish farms can impact wild fish populations by transferring disease and parasites to migrating fish. Aquaculture can also pollute water systems with excess nutrients and fecal matter due to the large numbers and concentrations of farmed fish.

CHAPTER 4

PROJECT DESCRIPTION

4.1 Problem statement

The farmers visit many times in a day to monitor the overall activities in the farm. Workers roam around the aqua farm routinely as per their scheduled time to check the water quality of the pond. Some uses handheld meters to manually check water parameters and others test in a laboratory by giving their water sample. Aquaculture is in a lagging scope of technology compared to other areas such as agriculture. So, it is supremely important to solve the problems with the support of technology.

4.2 Proposed mode

The system proposed is being implemented with sensors for temperature, water level, salinity, dissolved oxygen, and ammonia. These sensors are kept in the pond for observing the farming system. Each and every sensor gets detected instantly, stores the information in cloud and checks for upper and lower limit range, if any deviation is found at once an alert message is sent. The system designed runs in of nodemcu connected with sensors to it. The farming system can be monitored remotely with internet. The hurdle towards the establishing the aqua farm is based upon IOT for monitoring the real environment, the sensors will be accordingly interfaced and works automatically by giving the values

- NodeMCU Controller
- Temperature
- Water level sensor
- Ph sensor
- LCD Display
- Buzzer
- Power Supply

Block Diagram

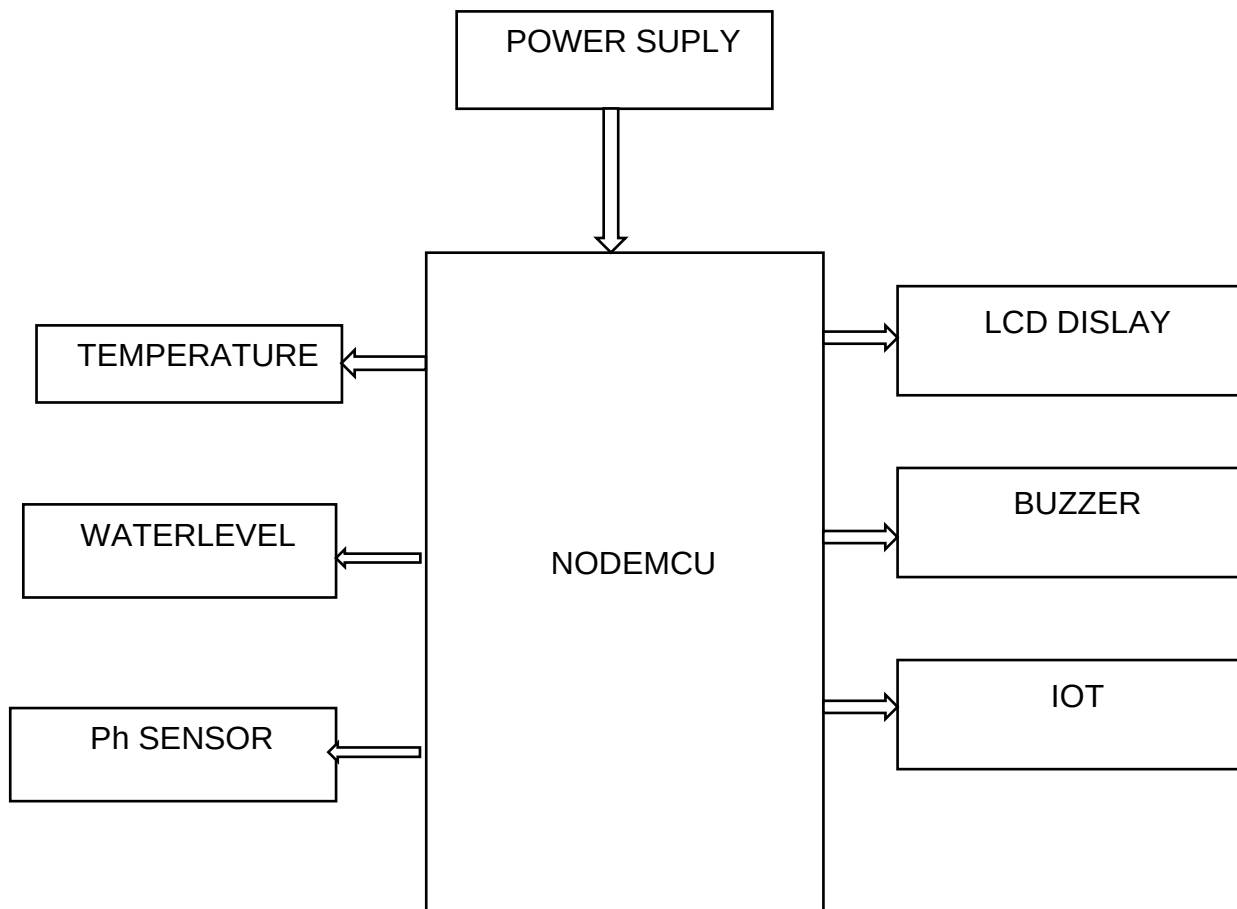


Fig:4.1: Block diagram of system

4.3 INTERNET OF THINGS(IOT)

IoT stands for Internet of Things, which means accessing and controlling daily usable equipment and devices using Internet.

Our IoT tutorial includes all topics of IoT such as introduction, features, advantage and disadvantage, ecosystem, decision framework, architecture and domains, biometric, security camera and door unlock system, devices, etc.

4.3.1 About Internet of Things (IoT)

Our mobile device contains GPS Tracking, Mobile Gyroscope, Adaptive brightness, Voice detection, Face detection etc. These components have their own individual features, but if these all communicate with each other to provide a better environment.

For example, the phone brightness is adjusted based on my GPS location or my direction.

Connecting everyday things embedded with electronics, software, and sensors to internet enabling to collect and exchange data without human interaction called as the Internet of Things (IoT).

The term “Things” in the Internet of Things refers to anything and everything in day-to-day life which is accessed or connected through the internet.

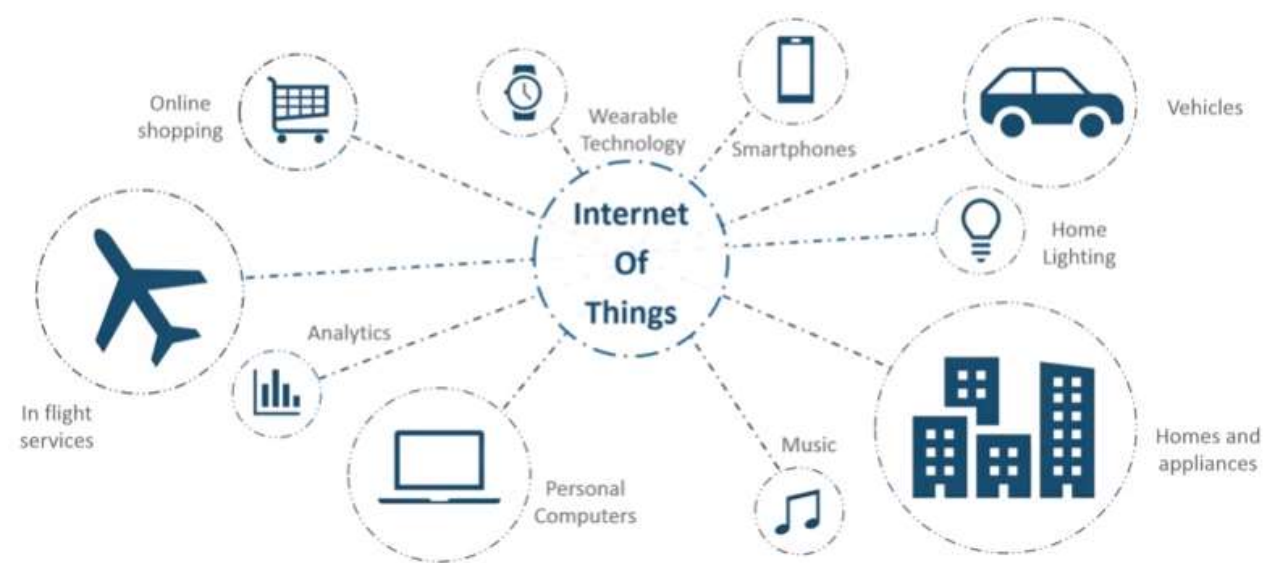


Fig: 4.2: IOT (Internet of things)

IoT is an advanced automation and analytics system which deals with artificial intelligence, sensor, networking, electronic, cloud messaging etc. to deliver complete systems for the product or services. The system created by IoT has greater transparency, control, and performance.

As a platform such as a cloud that contains all the data through which connects all the things around us. For example, a house, where people connect our home appliances such as air conditioner, light, etc. through each other and all these things are managed at the same platform. Since a platform, like iot can connect our car, track its fuel meter, speed level, and also track the location of the car.

The working of IoT is different for different IoT echo system (architecture). However, the key concept of there working is similar. The entire working process of IoT starts

with the device themselves, such as smartphones, digital watches, electronic appliances, which securely communicate with the IoT platform. This platform collects and analyze the data from all multiple devices and platforms and transfer the most valuable data with applications to devices.

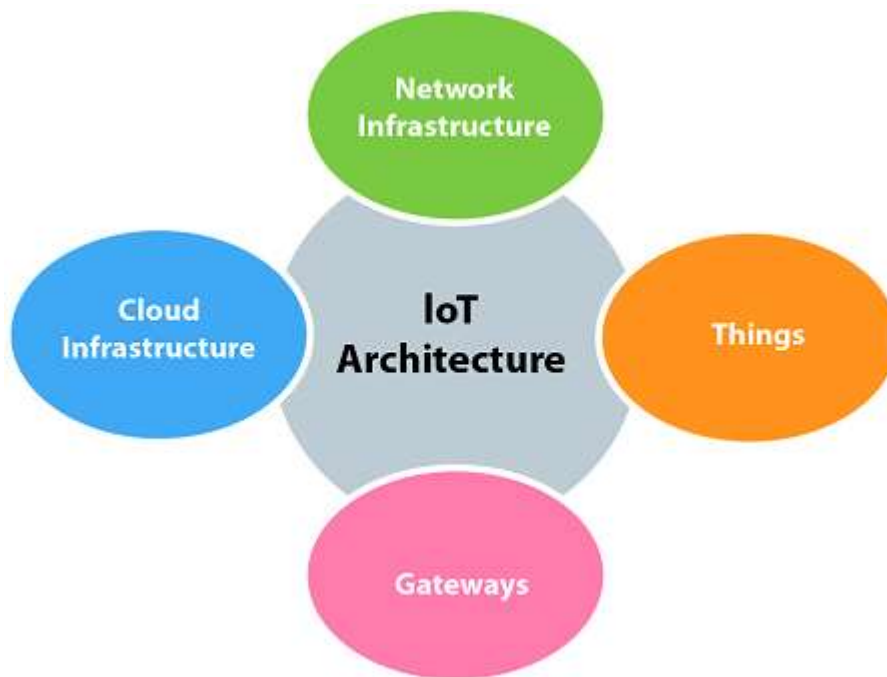


Fig:4.3: IOT Architecture

4.3.2 Features of IOT

The most important features of IoT on which it works are connectivity, analyzing, integrating, active engagement, and many more. Some of them are listed below:

Connectivity: Connectivity refers to establish a proper connection between all the things of IoT-to-IoT platform it may be server or cloud. After connecting the IoT devices, it needs a high-speed messaging between the devices and cloud to enable reliable, secure and bi-directional communication.

Analyzing: After connecting all the relevant things, it comes to real-time analyzing the data collected and use them to build effective business intelligence. Have a good insight into data gathered from all these things, call our system has a smart system.

Integrating: IoT integrating the various models to improve the user experience as well.

Artificial Intelligence: IoT makes things smart and enhances life through the use of data. For example, If the beans in our coffee maker run out, the coffee maker will order the coffee beans of the selected from the store.

Sensing: The sensor devices used in IoT technologies detect and measure any change in the environment and report on their status. IoT technology brings passive networks to active networks. Without sensors, there could not hold an effective or true IoT environment.

Active Engagement: IoT makes the connected technology, product, or services to active engagement between each other.

Endpoint Management: It is important to be the endpoint management of all the IoT system otherwise, it makes the complete failure of the system. For example, if a coffee machine itself order the coffee beans when it goes to end but what happens when it orders the beans from a retailer and not present at home for a few days, it leads to the failure of the IoT system. So, there must be a need for endpoint management.

4.4 SYSTEM ARCHITECTURE

4.4.1 Sensors

In this section six different sensors are used for checking the temperature, pH sensor and water level. The sensors transmit their information to the NodeMCU connected via UART (Universal Asynchronous Transmitter-Receiver). Each of these sensors is submerged in the pond where the measurement can be taken.

4.4.2 NODEMCU

NodeMCU is an open source IoT platform. It includes firmware which runs on the ESP8266 Wi-Fi SoC from Express if Systems, and hardware which is based on the ESP-12 module.^{[6][7]} The term "NodeMCU" by default refers to the firmware rather than the development kits. The firmware uses the Lua scripting language. It is based on the eLua project, and built on the Espressif Non-OS SDK for ESP8266. It uses many open-source projects., such as lua-cjson and SPIFFS

NodeMCU was created shortly after the ESP8266 came out. On December 30, 2013, Express Systems began production of the ESP8266. The ESP8266 is a Wi-Fi SoC integrated with a Tensilica Xtensa LX106 core, widely used in IoT applications (see related projects). NodeMCU started on 13 Oct 2014, when Hong committed the first file of nodemcu-firmware to GitHub. Two months later, the project expanded to include an open-hardware platform when developer Huang R committed the gerber file of an ESP8266 board, named devkit v0.9. Later that month, Tuan PM ported MQTT client library from Contiki to the ESP8266 SoC platform and committed to NodeMCU project, then NodeMCU was able to support the MQTT IoT protocol, using Lua to access the MQTT broker. Another important update was made on 30 Jan 2015, when Devsaurus ported the u8glib to NodeMCU project, enabling NodeMCU to easily drive LCD, Screen, OLED, even VGA displays.

In summer 2015 the creators abandoned the firmware project and a group of independent contributors took over. By summer 2016 the NodeMCU included more than 40 different modules. Due to resource constraints users need to select the modules relevant for their project and build a firmware tailored to their needs.

ESP8266 Arduino Core

As Arduino.cc began developing new MCU boards based on non-AVR processors like the ARM/SAM MCU and used in the Arduino Due and needed to modify the Arduino IDE so that it would be relatively easy to change the IDE to support alternate toolchains to allow Arduino C/C++ to be compiled for these new processors. The introduction of the Board Manager and the SAM Core. A "core" is the collection of software components required by the Board Manager and the Arduino IDE to compile an Arduino C/C++ source file for the target MCU's machine language. Some ESP8266 enthusiasts developed an Arduino core for the ESP8266 Wi-Fi SoC, popularly called the "ESP8266 Core for the Arduino IDE".^[16] This has become a leading software development platform for the various ESP8266-based modules and development boards, including NodeMCUs.

Introduction

NodeMCU is an open-source LUA based firmware developed for ESP8266 wifi chip. By exploring functionality with ESP8266 chip, NodeMCU firmware comes with ESP8266 Development board/kit i.e., NodeMCU Development board.



Fig :4.4: NodeMCU

Since NodeMCU is open-source platform, their hardware design is open for edit/modify/build.

NodeMCU Dev Kit/board consist of ESP8266 wifi enabled chip. The ESP8266 is a low-cost Wi-Fi chip developed by Espressif Systems with TCP/IP protocol. ESP8266 Wi-Fi Module is a good resource for further details on this device.

There is Version2 (V2) available for NodeMCU Dev Kit i.e., NodeMCU Development Board v1.0 (Version2), which usually comes in black colored PCB.

For more information about NodeMCU Boards available in market refer NodeMCU Development Boards

NodeMCU Dev Kit has Arduino like Analog (i.e., A0) and Digital (D0-D8) pins on its board.

It supports serial communication protocols i.e., UART, SPI, I2C etc.

Using such serial protocols and can connect it with serial devices like I2C enabled LCD display, Magnetometer HMC5883, MPU-6050 Gyro meter + Accelerometer, RTC chips, GPS modules, touch screen displays, SD cards etc.

NodeMCU Development board is featured with Wi-Fi capability, analog pin, digital pins and serial communication protocols.

To get start with using NodeMCU for IoT applications someone need to know about how to write/download NodeMCU firmware in NodeMCU Development Boards. And before that where this NodeMCU firmware will get as per our requirement.

There is online NodeMCU custom builds available using which can easily get our custom NodeMCU firmware as per our requirement.

Difference in using ESPlorer and Arduino IDE

There is a programming language difference can say while developing application for NodeMCU using ESPlorer IDE and Arduino IDE.

The need to code in C\C++ programming language can be uses Arduino IDE for developing NodeMCU applications and Lua language for using ESPlorer IDE.

Basically, NodeMCU is Lua Interpreter, so it can understand Lua script easily. When we write Lua scripts for NodeMCU and send/upload it to NodeMCU and executes sequentially. It will not build binary firmware file of code for NodeMCU to write. It will send Lua script as it is to NodeMCU to get execute.

In Arduino IDE when were write and compile code, ESP8266 toolchain in background creates binary firmware file of code which were written. And when it is upload it to NodeMCU then it will flash all NodeMCU firmware with newly generated binary firmware code. In fact, it writes the complete firmware.

That's the reason why NodeMCU not accept further Lua scripts/code after it is getting flashed by Arduino IDE. After getting flashed by Arduino sketch/code it will be no more Lua interpreter and any error occurs while try to upload Lua scripts. To again start with Lua script, need to flash it with NodeMCU firmware.

Since Arduino IDE compile and upload/writes complete firmware, it takes more time than ESPlorer IDE.

4.4.3 TEMPERATURE SENSOR

Temperature is the most often-measured environmental quantity. This might be expected since most physical, electronic, chemical, mechanical, and biological systems are affected by temperature. Certain chemical reactions, biological processes, and even electronic circuits perform best within limited temperature ranges. Temperature is one of the most commonly measured variables and it is therefore not surprising that there are many ways of sensing it. Temperature sensing can be done either through direct contact with the heating source, or remotely, without direct contact with the source using radiated energy instead. There are a wide variety

of temperature sensors on the market today, including Thermocouples, Resistance Temperature Detectors (RTDs), Thermistors, Infrared, and Semiconductor Sensors.

THERMOCOUPLE:

It is a type of temperature sensor, which is made by joining two dissimilar metals at one end. The joined end is referred to as the HOT JUNCTION. The other end of these dissimilar metals is referred to as the COLD END or COLD JUNCTION. The cold junction is actually formed at the last point of thermocouple material. If there is a difference in temperature between the hot junction and cold junction, a small voltage is created. This voltage is referred to as an EMF (electro-motive force) and can be measured and in turn used to indicate temperature.

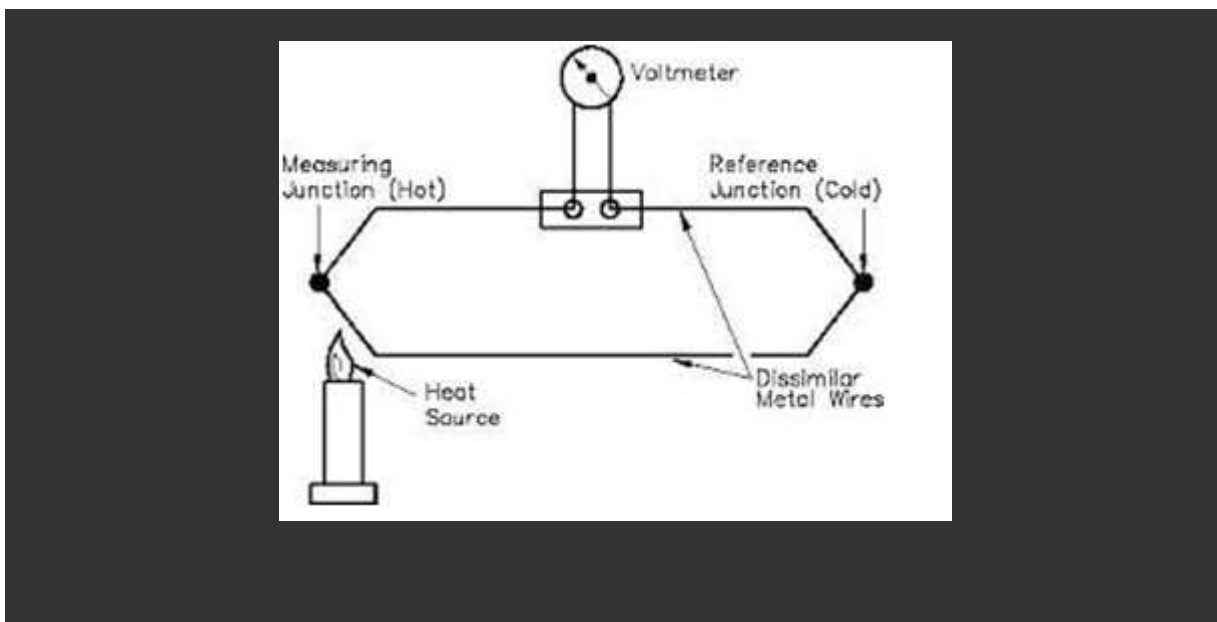


Fig :4.5: Thermocouple

The RTD is a temperature sensing device whose resistance changes with temperature. Typically built from platinum, though devices made from nickel or copper are not uncommon, RTDs can take many different shapes like wire wound, thin film. To measure the resistance across an RTD, apply a constant current, measure the resulting voltage, and determine the RTD resistance. RTDs exhibit fairly linear resistance to temperature curves over their operating regions, and any nonlinearity are highly predictable and repeatable. The PT100 RTD evaluation board uses surface

mount RTD to measure temperature. An external 2, 3 or 4-wire PT100 can also be associated with measure temperature in remote areas. The RTDs are biased using a constant current source. So as to reduce self-heat due to power dissipation, the current magnitude is moderately low. The circuit shown in figure is the constant current source uses a reference voltage, one amplifier, and a PNP transistor.

Thermistors:

Similar to the RTD, the thermistor is a temperature sensing device whose resistance changes with temperature. Thermistors, however, are made from semiconductor materials. Resistance is determined in the same manner as the RTD, but thermistors exhibit a highly nonlinear resistance vs. temperature curve. Thus, in the thermistors operating range can see a large resistance change for a very small temperature change. This makes for a highly sensitive device, ideal for set-point applications.

Semiconductor sensors:

These are classified into different types like Voltage output, Current output, Digital output, Resistance output silicon and Diode temperature sensors. Modern semiconductor temperature sensors offer high accuracy and high linearity over an operating range of about 55°C to $+150^{\circ}\text{C}$. Internal amplifiers can scale the output to convenient values, such as $10\text{mV}/^{\circ}\text{C}$. They are also useful in cold-junction compensation circuits for wide temperature range thermocouples.

Sensor ICs

There are a wide variety of temperature sensor ICs that are available to simplify the broadest possible range of temperature monitoring challenges. These silicon temperature sensors differ significantly from the above-mentioned types in a couple of important ways. The first is operating temperature range. A temperature sensor IC can operate over the nominal IC temperature range of -55°C to $+150^{\circ}\text{C}$. The second major difference is functionality.

A silicon temperature sensor is an integrated circuit, and can therefore include extensive signal processing circuitry within the same package as the sensor. There is no need to add compensation circuits for temperature sensor ICs. Some of these are analogue circuits with either voltage or current output. Others combine analogue-sensing circuits with voltage comparators to provide alert functions. Some other

sensor ICs combine analogue-sensing circuitry with digital input/output and control registers, making them an ideal solution for microprocessor-based systems.

Digital output sensor usually contains a temperature sensor, analog-to-digital converter (ADC), a two-wire digital interface and registers for controlling the IC's operation. Temperature is continuously measured and can be read at any time. If desired, the host processor can instruct the sensor to monitor temperature and take an output pin high (or low) if temperature exceeds a programmed limit. Lower threshold temperature can also be programmed and the host can be notified when temperature has dropped below this threshold. Thus, digital output sensor can be used for reliable temperature monitoring in microprocessor-based systems.

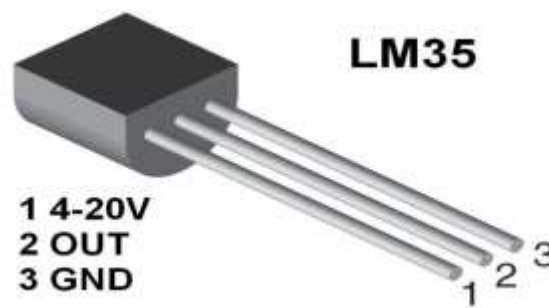


Fig :4.6: Temperature sensor

Above temperature sensor has three terminals and required Maximum of 5.5 V supply. This type of sensor consists of a material that performs the operation according to temperature to vary the resistance. This change of resistance is sensed by circuit and it calculates temperature. When the voltage increases then the temperature also rises. We can see this operation by using a diode.

Temperature sensors directly connected to microprocessor input and thus capable of direct and reliable communication with microprocessors. The sensor unit can communicate effectively with low-cost processors without the need of A/D converters.

An example for a temperature sensor is LM35. The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius temperature. The LM35 is operates at -55° to $+120^{\circ}\text{C}$. The basic centigrade temperature sensor ($+2^{\circ}\text{C}$ to $+150^{\circ}\text{C}$) is shown in figure below.

Features of LM35 Temperature Sensor:

Calibrated directly in ° Celsius (Centigrade)

Rated for full I –55° to +150°C range

Suitable for remote applications

Low cost due to wafer-level trimming

Operates from 4 to 30 volts

Low self-heating,

±1/4°C of typical nonlinearity

Operation of LM35:

The LM35 can be connected easily in the same way as other integrated circuit temperature sensor it established to the surface and its temperature will be within around the range of 0.01°C of the surface temperature.

This presumes that the ambient air temperature is just about the same as the surface temperature; if the air temperature were much higher or lower than the surface temperature, the actual temperature of the LM35 die would be at an intermediate temperature between the surface temperature and the air temperature.

The temperature sensors have well known applications in environmental and process control and also in test, measurement and communications. A digital temperature is a sensor, which provides 9-bit temperature readings. Digital temperature sensors offer excellent precise accuracy, these are designed to read from 0°C to 70°C and it is possible to achieve ±0.5°C accuracy. These sensors completely aligned with digital temperature readings in degree Celsius.

4.4.4 WATER LEVEL SENSOR:

Hardware Overview

The sensor has a series of ten exposed copper traces, five of which are power traces and five are sense traces. These traces are interlaced so that there is one sense trace

between every two power traces. Usually, these traces are not connected but are bridged by water when submerged.



Fig :4.7: Water level sensor

There's a Power LED on the board which will light up when the board is powered. The working of the water level sensor is pretty straightforward. The series of exposed parallel conductors, together acts as a variable resistor (just like a potentiometer) whose resistance varies according to the water level.

The change in resistance corresponds to the distance from the top of the sensor to the surface of the water.

The resistance is inversely proportional to the height of the water:

- The more water the sensor is immersed in, results in better conductivity and will result in a lower resistance.
- The less water the sensor is immersed in, results in poor conductivity and will result in a higher resistance.

The sensor produces an output voltage according to the resistance, which by measuring can be determine the water level.

Water Level Sensor Pinout

The water level sensor is super easy to use and only has 3 pins to connect.

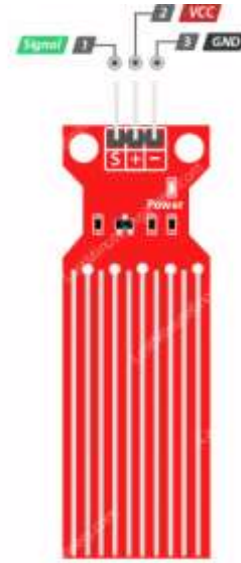


Fig :4.8: Water level sensor pinout

S (Signal) pin is an analog output that will be connected to one of the analog inputs of the Arduino.

+ (VCC) pin supplies power for the sensor. It is recommended to power the sensor with between 3.3V – 5V. Please note that the analog output will vary depending on what voltage is provided for the sensor.

– (GND) is a ground connection.

4.4.5 pH sensor

pH sensor is one of the most important tools for measuring pH and is commonly used in water quality monitoring. This type of sensor is capable of measuring alkalinity and acidity in water and other solutions. When used properly, pH sensors can ensure the safety and quality of products and processes that occur in wastewater or manufacturing plants.



Fig :4.9: pH sensor

In most cases, the standard pH range is represented by a value in the range of 0-14. When a substance has a pH value of 7, this is considered neutral. pH values above 7 represent higher alkalinity, while substances with pH values below 7 are considered more acidic. For example, toothpaste usually has a pH of 8-9. On the other hand, stomach acid has a pH of 2.

The distinction between alkaline and acidic substances is important for any company that uses cooling towers, boilers, manufacturing processes, pool control and various environmental monitoring. The standard pH of the human body is 7.4, which is essential for the body to function effectively. If the body's composition becomes too acidic or too alkaline, it will appear to revert to a neutral state.

pH sensors and transmitters are used in many industries, such as chemicals, water and waste water, food and beverage, pharmaceutical, power plants, primary and oil and gas. The choice of sensor depends on the application. Use ISFET sensors where glass breakage cannot be tolerated or where the medium contains large amounts of organic solvents. Apply enamel sensors when long-term stability and minimum maintenance requirements are required.

Benefits

Poor water quality may have a direct impact on the return on investment since it has a significant impact on the quality of the services and products that offer because it is utilized in many industrial operations. Managing water quality through different systems is also a key aspect of compliance with stringent environmental, industry and safety standards.



Fig :4.10: Water quality management

This kind of water quality management eliminate contaminants from influent water before using it in a range of industrial processes that focus in water pretreatment for manufacturing. With the removal of these impurities, operating efficiency will be improved, heating and cooling efficiency will be higher, and Everyone will be able to more successfully keep up with the required water quality requirements. If the monitoring system is effective, the benefits can be numerous and may include:

- Reduce water waste
- Energy saving
- Meet supply chain sustainability requirements
- Preventing downtime
- Maintain a healthy workplace for employees
- Reduce the use of hazardous chemicals

For everyday consumers, poor water quality can cause health problems, dull hair, rough clothing and dry and itchy skin. PH sensors play a very important role in water

quality and water treatment. When the pH of a water treatment plant becomes too high or too low, the water can be harmful.

Keep in mind that pH fluctuations are also costly to fix. Additional quantities of chemicals need to be used to remove waste, resulting in excessive downtime and higher costs. In standard wastewater treatment facilities, a pH of 6.5 to 8.5 is considered to be in the neutral range.

Keeping water at this level and constantly monitoring pH should help keep costs down. However, it is important to use different pH sensors to get the best water quality. For example, the use of a third electrode in a differential sensor makes the sensor ideal for a variety of industrial applications, such as wet scrubbers.

Advantages of our pH measuring instruments

- Our pH sensors and transmitters meet all requirements from basic functionality to multi-channel and multi-parameter measurements.
- The pH buffers, pH solutions and pH standards for the highest precision pH calibration and produced in our calibration laboratory.
- The different types of electrodes cover all types of diaphragms and reference systems, from fouling-resistant PTFE joints on highly sensitive ceramic diaphragms to anti-clogging orifices.
- The pH assemblies for several applications: fixed mount assemblies, submerged assemblies, flow-through assemblies and retractable assemblies.
- Automated systems for calibration, adjustment and cleaning ensure optimal performance of pH sensors and transmitters in chemical processes, pharmaceutical production or sanitary applications.

4.4.6 LCD Display

Liquid crystal displays (LCDs) have materials which combine the properties of both liquids and crystals. Rather than having a melting point, have a temperature range within which the molecules are almost as mobile as would be in a liquid, but are grouped together in an ordered form similar to a crystal.

An LCD consists of two glass panels, with the liquid crystal material sandwiched in between them. The inner surface of the glass plates is coated with transparent electrodes which define the character, symbols or patterns to be displayed. Polymeric layers are present in between the electrodes and the liquid crystal, which makes the liquid crystal molecules to maintain a defined orientation angle.

On each polarizer is pasted outside the two glass panels. These polarizers would rotate the light rays passing through them to a definite angle, in a particular direction.

When the LCD is in the off state, light rays are rotated by the two polarizers and the liquid crystal, such that the light rays come out of the LCD without any orientation, and hence the LCD appears transparent.

When sufficient voltage is applied to the electrodes, the liquid crystal molecules would be aligned in a specific direction. The light rays passing through the LCD would be rotated by the polarizers, which would result in activating / highlighting the desired characters.

The LCDs are lightweight with only a few millimeters thickness. Since the LCDs consume less power, compatible with low power electronic circuits, and can be powered for long durations.

The LCDs don't generate light and so light is needed to read the display. By using backlighting, reading is possible in the dark. The LCDs have long life and a wide operating temperature range.

Changing the display size or the layout size is relatively simple which makes the LCDs more customer friendly.

The LCDs used exclusively in watches, calculators and measuring instruments are the simple seven-segment displays, having a limited amount of numeric data. The recent advances in technology have resulted in better legibility, more information displaying capability and a wider temperature range. These have resulted in the LCDs being extensively used in telecommunications and entertainment electronics. The LCDs have even started replacing the cathode ray tubes (CRTs) used for the display of text and graphics, and also in small TV applications.

Crystalonics dot –matrix (alphanumeric) liquid crystal displays are available in TN, STN types, with or without backlight. The use of C-MOS LCD controller and driver ICs result in low power consumption. These modules can be interfaced with a 4-bit or 8-bit microprocessor /Micro controller.

The built-in controller IC has the following features:

- Correspond to high speed MPU interface (2MHz)
- 80 x 8-bit display RAM (80 Characters max)
- 9,920-bit character generator ROM for a total of 240-character fonts. 208-character fonts (5 x 8 dots) 32-character fonts (5 x 10 dots)
- 64 x 8-bit character generator RAM 8-character generator RAM 8-character fonts (5 x 8 dots) 4 characters fonts (5 x 10 dots)
- Programmable duty cycles
- 1/8 – for one line of 5 x 8 dots with cursor
- 1/11 – for one line of 5 x 10 dots with cursor
- 1/16 – for one line of 5 x 8 dots with cursor
- Wide range of instruction functions display clear, cursor home, display on/off, cursor on/off, display character blink, cursor shift, display shift.



Fig: 4.11: LCD display

FUNCTIONAL DESCRIPTION OF THE CONTROLLER IC

The controller IC has two 8-bit registers, an instruction registers (IR) and a data register (DR). The IR stores the instruction codes and address information for display data RAM (DD RAM) and character generator RAM (CG RAM). The IR can be written, but not read by the MPU.

The DR temporally stores data to be written to /read from the DD RAM or CG RAM. The data written to DR by the MPU, is automatically written to the DD RAM or CG RAM as an internal operation.

When an address code is written to IR, the data is automatically transferred from the DD RAM or CG RAM to the DR. data transfer between the MPU is then completed when the MPU reads the DR. likewise, for the next MPU read of the DR, data in DD RAM or CG RAM at the address is sent to the DR automatically. Similarly, for the MPU write of the DR, the next DD RAM or CG RAM address is selected for the write operation.

The dot-matrix liquid crystal display controller and driver LSI displays alphanumeric, Japanese kana characters, and symbols. It can be configured to drive a dot-matrix liquid crystal display under the control of a 4- or 8-bit microprocessor. Since all the functions such as display RAM, character generator, and liquid crystal driver, required for driving a dot-matrix liquid crystal display are internally provided on one chip, a minimal system can be interfaced with this controller/driver.

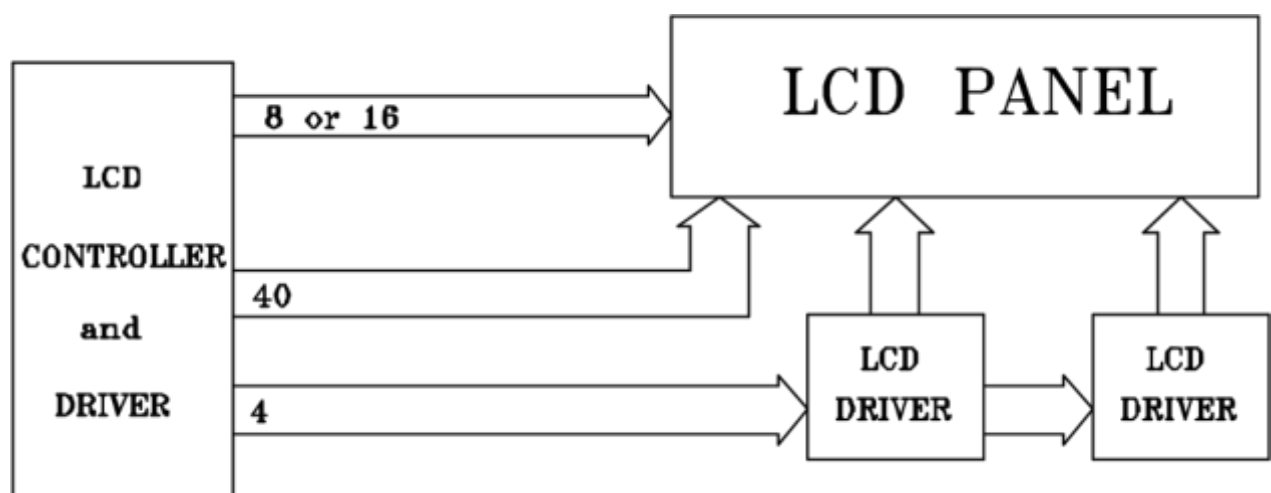


Fig :4.12: Block Diagram of LCD

4.4.7 Buzzer

There are many ways to communicate between the user and a product. One of the best ways is audio communication using a buzzer IC. So, during the design process, understanding some technologies with configurations is very helpful. So, this article discusses an overview of an audio signaling device like a beeper or a buzzer and its working with applications.

An audio signaling device like a beeper or buzzer may be electromechanical or piezoelectric or mechanical type. The main function of this is to convert the signal from audio to sound. Generally, it is powered through DC voltage and used in timers, alarm devices, printers, alarms, computers, etc. Based on the various designs, it can generate different sounds like alarm, music, bell & siren.



Fig :4.13: Buzzer

Buzzer Pin Configuration

The pin configuration of the buzzer is shown below. It includes two pins namely positive and negative. The positive terminal of this is represented with the '+' symbol or a longer terminal. This terminal is powered through 6Volts whereas the negative terminal is represented with the '-' symbol or short terminal and it is connected to the GND terminal.

Electromechanical

This buzzer was launched in the year 1831 by an American Scientist namely Joseph Henry but, this was used in doorbells until eliminated in 1930 in support of musical bells, which had a smooth tone.

Piezoelectric

These buzzers were invented by manufacturers of Japanese & fixed into a broad range of devices during the period of 1970s – 1980s. So, this development primarily came due to cooperative efforts through the manufacturing companies of Japanese. In the year 1951, the Application Research Committee of Barium Titanate that allows the corporations to be cooperative competitively & bring about numerous piezoelectric creations.

Specifications

The specifications of the buzzer include the following.

- Color is black
- The frequency range is 3,300Hz
- Operating Temperature ranges from – 20° C to +60°C
- Operating voltage ranges from 3V to 24V DC
- The sound pressure level is 85dBA or 10cm
- The supply current is below 15mA

CHAPTER 5

SOFTWARE REQUIREMENTS

5.1 Arduino uno and its programming

Arduino is a tool for making computers that can sense and control more of the physical world than the desktop computer. It's an open-source physical computing platform based on a simple microcontroller board, and a development environment for writing software for the board.

Arduino can be used to develop interactive objects, taking inputs from a variety of switches or sensors, and controlling a variety of lights, motors, and other physical outputs. Arduino projects can interact with software running on a computer or be stand-alone projects. The boards can be assembled by hand or purchased preassembled; the open-source IDE can be downloaded for free.

The Arduino programming language is an implementation of Wiring, a similar physical computing platform, which is based on the Processing multimedia programming environment.

The Duemilanove board features an Atmel ATmega328 microcontroller operating at 5 V with 2 Kb of RAM, 32 Kb of flash memory for storing programs and 1 Kb of EEPROM for storing parameters. The clock speed is 16 MHz, which translates to about executing about 300,000 lines of C source code per second. The board has 14 digital I/O pins and 6 analog input pins. There is a USB connector for talking to the host computer and a DC power jack for connecting an external 6-20 V power source, for example a 9 V battery, when running a program while not connected to the host computer. Headers are provided for interfacing to the I/O pins using 22 g solid wire or header connectors.

The Arduino programming language is a simplified version of C/C++. The Arduino can be programmed as someone is comfortable with C. There's no need to worry if one doesn't know C because only a few instructions are required to carry out useful activities

The ability to develop control programs on the host PC, download them to the Arduino, and have them run automatically is a key feature of the Arduino. Remove the USB

cable connection to the PC, and the program will still run from the top each time to push the reset button. Remove the battery and put the Arduino board in a closet for six months. The most recent programmed stored will start when the battery is reconnected. In other words, anyone connect the board to the host PC to build and debug the programme, but after that, anyone may execute the programme without the PC.

The Arduino Uno is a microcontroller board based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

5.2 Arduino Software (IDE)



Fig :5.1: Arduino IDE

Writing Sketches

Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension.ino. The editor has

features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The toolbar buttons allow people to verify and upload programs, create, open, and save sketches, and open the serial monitor.

NB: Versions of the Arduino Software (IDE) prior to 1.0 saved sketches with the extension. pde. It is possible to open these files with version 1.0, when saving, can save in the drawing with the.ino extension.

Additional commands are found within the five menus: File, Edit, Sketch, Tools, Help. The menus are context sensitive, which means only those items relevant to the work currently being carried out are available.

File

- New
Creates a new instance of the editor, with the bare minimum structure of a sketch already in place.
- Open
Allows to load a sketch file browsing through the computer drives and folders.
- Open
Provides a short list of the most recent sketches, ready to be opened.
- Sketchbook
Shows the current sketches within the sketchbook folder structure; clicking on any name opens the corresponding sketch in a new editor instance.
- Examples
Any example provided by the Arduino Software (IDE) or library shows up in this menu item. All the examples are structured in a tree that allows easy access by topic or library.
- Close
Closes the instance of the Arduino Software from which it is clicked.

- Save
Saves the sketch with the current name. If the file hasn't been named before, a name will be provided in a "Save as.." window.
- Save
Allows to save the current sketch with a different name.
- Page
It shows the Page Setup window for printing.
- Print
Sends the current sketch to the printer according to the settings defined in Page Setup.
- Preferences
Opens the Preferences window where some settings of the IDE may be customized, as the language of the IDE interface.
- Quit
Closes all IDE windows. The same drawings that were open when quit was selected will be opened by default the next time the IDE is launched.

Edit

- Undo/Redo
Goes back and undoes one or more editing steps; after undoing a step, someone may then redo it.
- Cut
Removes the selected text from the editor and places it into the clipboard.
- Duplicates the selected text in the editor and places it into the clipboard.
- Copy of the code will be on sketch to the clipboard in a form suitable for posting to the forum, complete with syntax coloring.
- Copy of the code will be on sketch to the clipboard as HTML, suitable for embedding in web pages.
- Paste
Puts the contents of the clipboard at the cursor position, in the editor.
- Select
Selects and highlights the whole content of the editor.

- **Comment/Uncomment**
Puts or removes the // comment marker at the beginning of each selected line.
- **Increase/Decrease**
Adds or subtracts a space at the beginning of each selected line, moving the text one space on the right or eliminating a space at the beginning.
- **Find**
Opens the Find and Replace window where one may choose from a number of parameters to determine the text to search for inside the current drawing.
- **Find**
Highlights the next occurrence - if any - of the string specified as the search item in the Find window, relative to the cursor position.
- **Find**
Highlights the previous occurrence - if any - of the string specified as the search item in the Find window relative to the cursor position.

Sketch

- **Verify/Compile**
Check the sketch for errors compiling it; it will report memory usage for code and variables in the console area.
- **Upload**
Compiles and loads the binary file onto the configured board through the configured Port.
- **Upload Using Programmer** This will replace the bootloader on the board; to restore it and resume uploading to the USB serial port, choose Tools > Burn Bootloader. On the other hand, it permits users to use the entire Flash memory for the sketch. Please note that this command will NOT burn the fuses. To do so a Tools -> Burn Bootloader command must be executed.
- **Export Compiled Binary**
Saves a .hex file that may be kept as archive or sent to the board using other tools.
- **Show Sketch Folder**
Opens the current sketch folder.

- Include Library

By including `#include` lines at the beginning of the code, users may add a library to the code they are writing. See the libraries below for further information.

This menu option also gives users access to the Library Manager, where one may import fresh libraries from .zip files.

- Add File...

Adds a source file to the sketch (it will be copied from its current location). The new file appears in a new tab in the sketch window. Files can be removed from the sketch using the tab menu accessible by clicking on the small triangle icon below the serial monitor one on the right side of the toolbar.

Tools

- Auto Format

This formats the code nicely: i.e., indents it so that opening and closing curly braces line up, and that the statements inside curly braces are indented more.

- Archive Sketch

Archives a copy of the current sketch in .zip format. The archive is placed in the same directory as the sketch.

- Fix Encoding & Reload

Fixes possible discrepancies between the editor char map encoding and other operating systems char maps.

- Serial Monitor

Opens the serial monitor window and initiates the exchange of data with any connected board on the currently selected Port. This usually resets the board, if the board supports Reset over serial port opening.

- Board

Choose the board being currently using. See below for descriptions of the various boards.

- Port

All of the serial devices on the machine, actual or virtual, are listed in this menu. Every time people use the top-level tools menu, it ought to immediately update.

- Programmer

While programming a board or chip without utilizing the inbuilt USB-serial

connection, choosing a hardware programmer. In most cases, one won't need this, but it's required if someone using a burning a bootloader to a new microcontroller.

- Burn Bootloader

Users can burn a bootloader into the Arduino board's microcontroller using the options in this menu. This is not necessary for using an Arduino or Genuino board ordinarily, but it might be helpful for those who buy a new ATmega microcontroller, which often does not come with a bootloader. Before burning the bootloader onto the target board, make sure the appropriate board was chosen from the Boards menu. The proper fuses were also set by this instruction.

Help

Several documentations that are included with the Arduino Software (IDE) are accessible here. Without an online connection, there's local access to the Getting Started, Reference, this IDE tutorial, and other publications. The papers may contain links back to our online domain and are local copies of the online versions.

Sketchbook

A sketchbook is a metaphor used by the Arduino Software (IDE) as a common location to store programmes (or sketches). One may access the sketches in the sketchbook by choosing File > Sketchbook from the menu or by clicking the Open button on the toolbar. The Arduino software will automatically create a directory in the sketchbook prior to launch it for the first time. With the Preferences dialogue, users may inspect or modify the sketchbook's location.

Beginning with version 1.0, files are saved with a .ino file extension. Previous versions use the .pde extension. Users may still open .pde named files in version 1.0 and later, the software will automatically rename the extension to .ino.

Tabs, Multiple Files, and Compilation

Allows to manage sketches with more than one file (each of which appears in its own tab). These can be normal Arduino code files (no visible extension), C files (.c extension), C++ files (.cpp), or header files (.h).

Uploading

Before uploading the sketch, the appropriate options must be chosen from the Tools > Board and Tools > Port menus. The boards are described below. On the Mac, the

serial port is probably something like `/dev/tty.usbmodem241` (for an Uno or Mega2560 or Leonardo) or `/dev/tty.usbserial-1B1` (for a Duemilanove or earlier USB board), or `/dev/tty.USA19QW1b1P1.1` (for a serial board connected with a Key span USB-to-Serial adapter). On Windows, it's probably COM1 or COM2 (for a serial board) or COM4, COM5, COM7, or higher (for a USB board) - to find out in the Windows Device Manager's port area, search for USB serial devices. It should be `/dev/ttyACMx`, `/dev/ttyUSBx`, or something like on Linux. Once user selected the correct serial port and board, press the upload button in the toolbar or select the Upload item from the Sketch menu. Current Arduino boards will reset automatically and begin the upload. With older boards (pre-Decimole) that lack auto-reset, user need to press the reset button on the board just before starting the upload. On most boards, user will see the RX and TX LEDs blink as the sketch is uploaded. The Arduino Software (IDE) will display a message when the upload is complete, or show an error.

When someone upload a sketch, the Arduino bootloader is being used, a small program that has been loaded on to the microcontroller on the board. It allows user to upload code without using any additional hardware. The bootloader is active for a few seconds when the board resets; then it starts whichever sketch was most recently uploaded to the microcontroller. The bootloader will blink the on-board (pin 13) LED when it starts (i.e., when the board resets).

Libraries

Libraries provide extra functionality for use in sketches, e.g., working with hardware or manipulating data. To use a library in a sketch, select it from the Sketch > Import Library menu. This will insert one or more `#include` statements at the top of the sketch and compile the library in the sketch. Libraries expand the size of the sketch since and submitted to the board along with it. If a sketch no longer needs a library, simply delete its `#include` statements from the top of the code.

There is a list of libraries in the reference. Some libraries are included with the Arduino software. Others can be downloaded from a variety of sources or through the Library Manager. Starting with version 1.0.5 of the IDE, import a library from a zip file and use it in an open sketch. See these instructions for installing a third-party library.

Third-Party Hardware

Everyone may add support for third-party hardware to the sketchbook directory's hardware directory. Platforms installed there may include board definitions (which appear in the board menu), core libraries, bootloaders, and programmer definitions. To install, create the hardware directory, then unzip the third-party platform into its own sub-directory. (Don't use "Arduino" as the sub-directory name or it will override the built-in Arduino platform.) To uninstall, simply delete its directory.

For details on creating packages for third-party hardware, see the Arduino IDE 1.5 3rd party Hardware specification.

Serial Monitor

This displays serial sent from the Arduino or Genuino board over USB or serial connector. To send data to the board, enter text and click on the "send" button or press enter. Choose the baud rate from the drop-down menu that matches the rate passed to Serial. Begin in the sketch. Note that on Windows, Mac or Linux the board will reset (it will rerun the sketch) when someone connect with the serial monitor. Please note that the Serial Monitor does not process control characters; if the sketch needs a complete management of the serial communication with control characters, The COM port allotted to the Arduino board can be used by an external terminal programmed.

Preferences

The preferences dialogue (located in the File or Arduino menus on Windows and Linux) allows users to set a variety of options. The preferences file, the location of which is indicated in the preference window, contains the remaining information.

5.3 Language Support

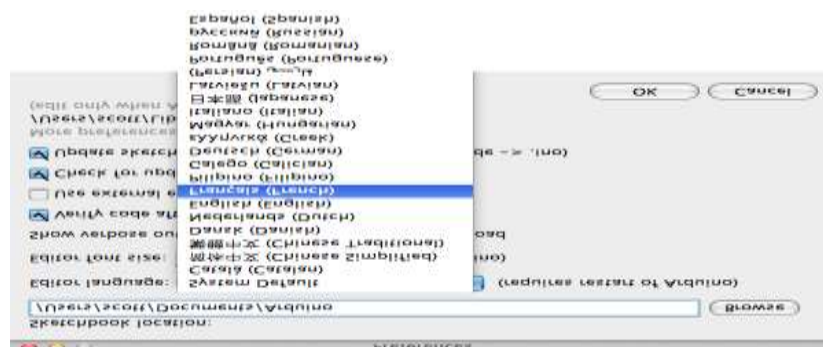


Fig :5.2: Language Support

Since version 1.0.1, the Arduino Software (IDE) has been translated into 30+ different languages. By default, the IDE loads in the language selected by the operating system. (Note: on Windows and possibly Linux, this is determined by the locale setting which controls currency and date formats, not by the language the operating system is displayed in.)

Open the Preferences window after starting the Arduino Software (IDE) if one wants to manually change the language. Next to the Editor Language there is a dropdown menu of currently supported languages. Select the preferred language from the menu, and restart the software to use the selected language. If the operating system language is not supported, the Arduino Software (IDE) will default to English.

By choosing System Default from the Editor Language drop-down menu, one can reset the software so that it picks a language depending on its operating system. As soon it's time to restart the Arduino Software (IDE), this configuration will become active. Similar to how one must restart the OS after changing the operating system's default language, one must also restart the Arduino Software (IDE).

CHAPTER 6

CONCLUSION

After successfully testing and validating the device, thus the designed system is suitable for real-time water monitoring and can be used in a variety of sectors besides aquaculture, such as hydroponics, riverbed water quality measures, and so on. This system continuously measures and displays water quality parameters such as pH, TDS, and dissolved oxygen on a webpage in real-time. The full-scale industrial trials are required to evaluate the system's efficiency before it can be employed on a large scale.

In the future, an SMS/call-based alerting system could be added to let people be notified when the quality falls below the threshold. By integrating a call/SMS-based system, farmers will be able to reduce risk and take preventative measures. By incorporating industrial quality sensors into the system, more precise readings, cheaper maintenance costs, and a longer system life can be achieved.

People would like to suggest a few things for this project's future development. The user's ability to monitor and manage the system will improve with the usage of cloud technologies to track data and obtain insights. A Nano microprocessor can be used to implement the system, which reduces the device's size and allows for easy installation in a variety of industrial applications. To save data, the system can be linked to a cloud database like SQL or Azure, which makes it easier to store big volumes of data. This data is utilized to perform predictive analysis, which will inform users about future water quality patterns for a given period of time.

6.1 Future scope

An IoT-based water level sanitation system for aquaculture applications using NodeMCU can have a promising future. NodeMCU is an open-source firmware and development board that enables easy and rapid prototyping for IoT applications. It provides a convenient way to connect to the internet and communicate with various sensors and devices.

The use of an IoT-based water level sanitary system for aquaculture can help in the efficient management of water resources, which is crucial for the growth and health of aquatic organisms. The system can monitor and control the water level, pH,

temperature, and other parameters, thereby ensuring optimal conditions for the aquaculture.

Moreover, the system can also provide alerts and notifications in case of any abnormality or deviation from the desired parameters, allowing for timely action and the prevention of any potential harm to the aquatic organisms.

With the increasing demand for sustainable and efficient aquaculture practises, the adoption of IoT-based systems is likely to increase in the future. The system's ability to provide real-time monitoring and control, along with remote access, can significantly enhance the efficiency and productivity of aquaculture operations.

6.2 Research issue

The IoT-based water level sanitary system for aquaculture applications using Nodemcu is an interesting and challenging research issue. The primary objective of such a system is to automate the monitoring and control of the water level in the aquaculture tanks to maintain optimal water quality for fish health and growth.

One of the key challenges in designing such a system is the selection of appropriate sensors to accurately measure the water level and quality parameters such as temperature, pH, dissolved oxygen, and ammonia. These sensors need to be robust, reliable, and capable of providing real-time data to the Nodemcu microcontroller for processing.

Another challenge is the development of an efficient communication protocol between the sensors and the Nodemcu microcontroller. The system should be able to transmit and receive data wirelessly over a long range and in a secure manner to prevent unauthorised access or tampering.

Finally, the development of an intuitive and user-friendly interface for the end-users is crucial to ensuring easy setup, configuration, and monitoring of the system. The interface should allow the user to view real-time data, set alarms for critical events, and remotely control the system.

6.3 Implement issues

Implementing an IoT-based water level sanitary system for aquaculture applications using Nodemcu involves several steps and potential issues. Here are some steps and considerations to keep in mind:

1. **Hardware selection:** Select the appropriate sensors, actuators, and other components that are compatible with Nodemcu and suitable for the aquaculture application. For water level monitoring, some may consider using ultrasonic or pressure sensors.
2. **Software development:** develop the firmware for Nodemcu using the Arduino IDE or other suitable programming environments. The firmware should include code to read data from the sensors, process it, and control the actuators. To transfer the data to the cloud or a local server, administrators might also need to have a communication protocol in place.
3. **Connectivity:** Ensure that Nodemcu is connected to the Internet using Wi-Fi or other suitable connectivity options. Make sure that the connectivity is reliable and secure, and consider using encryption or other security measures to protect the data.
4. **Power supply:** Ensure that the Nodemcu and other components are powered adequately and that the power supply is reliable and stable.
5. **Testing and calibration:** Test the system thoroughly and calibrate the sensors and actuators to ensure accurate measurements and control.

Some potential issues may encounter during the implementation process include:

1. **Compatibility issues:** Ensure that all the components and software are compatible with each other and work seamlessly. If compatibility problems occur, users might need to alter the firmware or use other components.
2. **Connectivity issues:** Connectivity can be a major issue, especially if the system is deployed in remote locations. Ensure that the Wi-Fi or other connectivity options are reliable and stable and have a sufficient range.
3. **Power supply issues:** The power supply can be a bottleneck, especially if the system is deployed in areas with limited access to electricity. Consider using solar or other renewable energy sources to power the system.

4. Sensor calibration issues: Sensors may require periodic calibration to maintain accuracy. Ensure that the system has a mechanism for sensor calibration, and schedule regular calibration checks.
5. Security issues: IoT systems are vulnerable to cyberattacks and other security breaches. Implement suitable security measures to protect the data and the system from unauthorised access.

6.4 Results

The data can be displayed on the website by using iframes, which is a simple way to exchange data with a website. First, we created a WordPress account, then we chose a domain name for our website and modified the preinstalled template. Finally, we added an HTML code block and pasted the iframes data from the Thing Speak channel into the code block. The final step was the launch of our website. The designed website can be accessed through. Figure 6.2 showing temperature of the system. Figure 6.3 shows pH level of the system, and Figure 6.1 show the time series charts of the water quality, that can be viewed on the website.



Fig: 6.1: Graph showing water level of the system

Figure 6.1 shows the graph of water level readings for the controlled aquaculture setup. Whenever the reading goes beyond the threshold level, the system automatically corrects the water.



Fig :6.2: Graph showing temperature of the system

Figure 6.2 shows the temperature sensor readings for the controlled aquaculture setup. Ideal water temperature for Nile Tilapia ranges from 25-27 °C. A water heater is used to prevent temperature from dropping.



Fig :6.3: Graph showing pH level of the system

Figure 6.3 shows the pH level for the controlled aquaculture setup. Whenever the reading goes beyond the threshold level, the system automatically corrects the water.

REFERENCES

- [1] Hong-Jun Zhu, (2010), "Global Fisheries Development Status and Future Trend Analysis", Taiwan Economic Research Monthly, 33(3).
- [2] Changhui Deng, Yanping Gao, Jun Gu, Xinying Miao, "Research on the Growth Model of Aquaculture Organisms Based on Neural Network Expert System," Sixth International Conference on Natural Computation; pg.no 1812-1815,
- [3] Sheetal Israni ,Harshal Meharkure , Parag Yelore, "Application of IoT based System for Advance Agriculture in India,"International Journal of Innovative research in Computer and Communication Engineering Vol. 3,Issue. 11, November2015.
- [4] Nikesh Gondchawar, Prof. Dr. R. S. Kawitkar, "IoT based smart Agriculture," International Journal of advanced research in Computer and Communication Engineering, Vol.5, Issue. 6, June 2016.
- [5] S.Kayalvizhi, Koushik Reddy G, Vivek Kumar P, VenkataPrasanth N, "Cyber AquaCulture Monitoring System Using ArduinioAnd Raspberry Pi," International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 4,Issue 5,Pg:2320-3765; May 2015.
- [6] Daudi S. Simbeye and Shi Feng Yang, "Water Quality Monitoring and Control for Aquaculture Based on Wireless Sensor Networks," JOURNAL OF NETWORKS, VOL. 9, NO. 4, APRIL 2014.
- [7] Mr. Kiran Patil,Mr. Sachin Patil,Mr. Sachin Patil and Mr. Vikas Patil,"Monitoring of Turbidity,pH & Temperature of Water Based on GSM,"International Journal for Research in Emerging Science and Technology,Volume-2,Issue-3,March-2015.
Bhatnagar, Anita, and Pooja Devi. "Water quality guidelines for the management of pond fish culture." International Journal of Environmental Sciences 3, no. 6 (2013):1980.
- [8] Raju, K. Raghu Sita Rama, and G. Harish Kumar Varma. "Knowledge Based RealTime Monitoring System for Aquaculture Using IoT." In Advance Computing Conference (IACC), 2017 IEEE 7th International, pp. 318-321. IEEE, 2017
- [9] Raju, K. Raghu Sita Rama, and G. Harish Kumar Varma. "Knowledge Based Real Time Monitoring System for Aquaculture Using IoT." In Advance Computing

Conference (IACC), 2017 IEEE 7th International, pp. 318-321. IEEE, 2017

[10] Boyd CE. Water quality management for pond fish culture. Elsevier Scientific Publishing Co.; 1982.

[11] Delincé, Guy. The ecology of the fish pond ecosystem: with special reference to Africa. Vol. 72. Springer Science & Business Media, 2013.

[12] Bhatnagar A, Jana SN, Garg SK, Patra BC, Singh G, Barman UK. Water quality management in aquaculture. Course Manual of summerschool on development of sustainable aquaculture technology in fresh and saline waters, CCS Haryana Agricultural, Hisar (India)

[13] Rodhouse, P. "Managing and forecasting squid fisheries in variable environments". Fisheries Research, 54, pp. 3-8.

[14]. Petersen, A., J. S. Andersen, T. Kaewmak, T. Somsiri, and A. Dalsgaard (2002). "Impact of integrated fish farming on antimicrobial resistance in a pond environment", Appl. Environ. Microbiol.

[15]. Doni, A., C. Murthy, and M. Kurian. (2018). "Survey on multi sensorbased air and water quality monitoring using IoT", (IJSR), pp. 147-153.

[16]. Zhu, X., D. Li, D. He, J. Wang, D. Ma, and F. Li. (2010). "A remote wireless system for water quality online monitoring in intensive fish culture." In Proceedings of the Computers and Electronics in Agriculture (CEA) pp. S3-S9

[17] H. N. Binti Hasim, M. Ramalingam, F. Ernawan, and R. Puviarasi, "Developing fish feeder system using Raspberry Pi," Proc. 3rd IEEE Int. Conf. Adv. Electr. Electron. Information, Commun. Bio-Informatics, AEEICB 2017, pp. 246–250, 2017.

[18] K. Meethongjan and S. Kongsong, "Aquarium Fish Smart Farming on Internet of Things (IoT) and Mobile Application Technology," Int. J. Bus. Tour. Appl. Sci., pp. 22–28, 2019.

[19] I. S. Akila, P. Karthikeyan, H. M. V. Hari, and K. J. Hari, "IoT Based Domestic Fish

[20] <http://www.ijetajournal.org/volume-5/issue-2/IJETA-V5I2P27.pdf>

- [21] S. Geetha and S. Gouthami, "Internet of things enabled real time water quality monitoring system," Smart Water, Vol. 2, No. 1, pp. 1-19, 2016
- [22] T. Bourtade and P. Janmak, "Automatic catfish farming in a cement pond," Rajamangala University of Technology Lanna Tak
- [23] Mourvika S, Monika A, Jyoti B, Trupti M (2018) IOT Based Water Quality Monitoring System, IJSRCSEIT 3; 2456-3307.
- [24] Mohammad S, Talha B, Subhasish G , Abhijit P (2019) IoT Based Real-time River Water Quality Monitoring System, Procedia Computer Science 155; 161-168.

APPENDIX

SOURCE CODE

```
#include <ESP8266WiFi.h>
#include <LiquidCrystal.h>

LiquidCrystal lcd(D0, D1, D2, D3, D4, D5);
String apiKey = "NB8770T7I2SZD2HH";
const char *ssid = "1234TEST";    key
const char *pass = "1234test";
const char* server = "api.thingspeak.com";
WiFiClient client;

void setup()
{
    pinMode(D6, OUTPUT);
    pinMode(D7, INPUT);
    digitalWrite(D6,LOW);
    lcd.begin(16, 2);
    // Print a message to the LCD.
    lcd.clear();
    lcd.print("AQUACULTURE MNTR");
    lcd.setCursor(0,1);
    lcd.print("CNTRL&SYSTEM");
    WiFi.begin(ssid, pass);
    while (WiFi.status() != WL_CONNECTED) ;
}

void loop()
{
    float temperature = analogRead(A0);
    delay(50);
    lcd.clear();
```

```

    lcd.print("TEMP=");
    lcd.print(temperature);
    if(digitalRead(D7)==HIGH)
{
    lcd.setCursor(0,1);
    lcd.print("LEVEL: EMPTY");
    digitalWrite(D6,HIGH);
    }
    if(digitalRead(D7)==LOW)
{
    lcd.setCursor(0,1);
    lcd.print("LEVEL: FULL");
    digitalWrite(D6,LOW);
    }

    if (client.connect(server,80))
    {
        String postStr = apiKey;
        postStr += "&field1=";
        postStr += String(temperature);
        delay(500);
        postStr += "&field2=";
        postStr += String(digitalRead(D6));
        client.print("POST /update HTTP/1.1\n");
        client.print("Host: api.thingspeak.com\n");
        client.print(postStr);
    }

    delay(1000);
}

```