BMS COLLEGE OF ENGINEERING

(Autonomous College under VTU)

Bull Temple Road, Basavanagudi, Bangalore - 560019

A mini project report on

“SMART POT UISNG IOT”

Submitted in partial fulfillment of the requirements for the award of degree

MASTER OF TECHNOLOGY

IN

COMPUTER NETWORK ENGINEERING

By

(1BM22SCN01)

AMMAJI KAVALESWARI

(1BM22SCN10)

SUHASINI H S

(1BM22SCN11)

SUSHMA H P

Under the guidance of

Dr. ASHOK KUMAR R

PROFESSOR

Department of Information Science and Engineering

2022-2023

BMS COLLEGE OF ENGINEERING
(Autonomous College under VTU)
Bull Temple Road, Basavanagudi,
Bangalore – 560019

Department of Information Science and Engineering

C E R T I F I C A T E

This is to certify that the group project entitled **“SMART POT USING IOT**” is a bona-fide work carried out by **Ammaji kavalleswari (1BM22SCN01), Suhasini H S (1BM22SCN10) and Sushma H P (1BM22SCN11)** in partial fulfillment for the award of degree of Master of Technology in **Computer Network Engineering** from **Visvesvaraya Technological University, Belgaum** during the year **2022-2023**. The group project report has been approved as it satisfies the academic requirements in respect of group project prescribed for the Master of Technology Degree.

Signature of the Guide
Dr. ASHOK KUMAR R
PROFESSOR
Dept of ISE, BMSCE

Signature of the HoD
Dr. P. JAYAREKHA
PROFESSOR
Dept of ISE, BMSCE

INDEX

ABSTRACT

In our increasingly interconnected society, electronic devices have become an integral part of people's lives, from watches and cellphones to computers. This project introduces an innovative approach to smart pot and soil moisture control using the NodeMCU, a soil moisture sensor. The automatic smart pot system is designed to sense soil moisture levels and activate the water pump when necessary, addressing the pressing need for efficient water usage in plant. Traditional manual Watering for plant methods often result in either water wastage or insufficient water reaching to plant. To mitigate these issues, a microcontroller-based Automatic watering to plant system is proposed, which ensures precise water delivery based on plant requirements. This automated system employs valves, controllers and optimizing water usage and reducing runoff. By automating the watering process, peoples can provide the right amount of water at the right time, improving crop performance and maximizing profits. Automatic watering to plant, particularly in specialized greenhouse vegetable production, offers an accurate and efficient means of soil moisture control, saving time, eliminating human errors, and ultimately contributing to sustainable and profitable farming practices.

CHAPTER 1:

INTRODUCTION

A smart pot powered by NodeMCU in the realm of IoT represents an ingenious fusion of technology and horticulture. By harnessing the capabilities of NodeMCU, an affordable open-source IoT platform based on the ESP8266 WiFi module, this project introduces an efficient and convenient way to nurture plants. It involves integrating NodeMCU into a traditional plant pot, enabling remote monitoring and control of key environmental parameters like soil moisture, temperature, and light levels through a web or mobile application. This intelligent system relies on components such as soil moisture sensors, temperature and humidity sensors, and light sensors, along with optional features like water pumps for automated watering.

Users can craft custom firmware for the NodeMCU, connect sensors to capture data, and employ IoT connectivity to transmit information to a cloud platform or local server. A user-friendly interface allows real-time monitoring and provides controls for watering, making plant care more informed and accessible. By embracing this IoT-powered innovation, gardening enthusiasts can cultivate healthier and thriving plants while indulging their tech-savvy.

Furthermore, the smart pot project using NodeMCU in IoT is a holistic endeavor that empowers plant enthusiasts to optimize their gardening routines. With the help of a dedicated web or mobile application, users gain insights into their plants' well-being by effortlessly accessing real-time sensor data. This data not only aids in tracking the plant's vital signs but also facilitates the setup of personalized alerts or notifications, ensuring that no critical care requirements go unnoticed.

The system's potential for automation, particularly in watering based on soil moisture levels, alleviates the burden of manual upkeep. Regular maintenance checks for sensor accuracy and NodeMCU functionality are all that's needed to keep this technological gardening assistant operating smoothly. Overall, a smart pot with NodeMCU at its core not only brings an element of futuristic convenience to gardening but also fosters a deeper connection with nature by offering a more intimate understanding of a plant's unique needs.side.

PROBLEM STATEMENT

In the present scenario, with the increasing urbanization and fast-paced lifestyle, people often struggle to maintain and nurture indoor plants effectively. This leads to a decline in indoor air quality, aesthetics, and overall well-being. Additionally, it becomes a challenge for individuals with busy schedules or limited horticultural knowledge to provide the appropriate care and attention that plants require.

In this context, there is a pressing need for a smart pot integrated with Internet of Things (IoT) technology to address these issues. The smart pot should be capable of monitoring various environmental parameters such as soil moisture levels, ambient temperature, humidity, and light intensity. It should also have the ability to autonomously adjust these conditions to create an optimal growing environment for the plants. Furthermore, it should provide real-time notifications and insights to users through a user-friendly interface, allowing them to remotely monitor and manage their indoor plants.

The solution should be cost-effective, easy to install, and compatible with a wide range of plant species. It should also be energy-efficient and sustainable, promoting eco-conscious practices. The smart pot should be designed to enhance the overall indoor living experience, promote greenery, and contribute positively to the health and well-being of individuals in urban environments.

OBJECTIVES

- **Automated Watering:** Smart pots can monitor soil moisture levels and provide automated watering based on plant needs. The primary objective is to ensure plants receive the right amount of water, preventing overwatering or underwatering.
- **Soil Health Monitoring:** These systems often include sensors to monitor soil health parameters such as pH levels, nutrient levels, and temperature. The objective is to maintain optimal soil conditions for plant growth.
- **Temperature and Humidity Regulation:** Smart pots may include sensors to monitor temperature and humidity levels around the plants. The objective is to create an ideal environment for plant growth, optimizing conditions for specific plant species.
- **Remote Monitoring and Control:** Smart pots often come with smartphone apps or web interfaces that allow users to monitor and control their plants remotely. This objective enables users to keep an eye on their plants and adjust as needed, even when they are away from home.
- **Temperature and Humidity Regulation:** Smart pots may include sensors to monitor temperature and humidity levels around the plants. The objective is to create an ideal environment for plant growth,

optimizing conditions for specific plant species.

- **Remote Monitoring and Control:** Smart pots often come with smartphone apps or web interfaces that allow users to monitor and control their plants remotely. This objective enables users to keep an eye on their plants and adjust as needed, even when they are away from home.

Chapter 2:

REVIEW OF LITERATURE

N. P. Singh – 2016 et.al [1] proposed the indoor plants are high in demand and the number of people buying indoor plants is increasing day by day. But people face a lot of problems in raising the indoor plants as taking care of these indoor plants is not easy and they even die with a cause which is difficult to predict. According to the National Gardening Association people from all age groups ranging 20 and above are completely obsessed with indoor plants. It states that in the US alone the sale of indoor plants has incremented by 50% during the last three years which now stands at a value of $1.7 billion in 2019. Barometer of Trade report shows the sale of indoor plants in India went up by 24.1% in a span of one year from 2016 to 2017. The market of indoor plants is increasing day by day so is the problem to take care of these plants. To raise and maintain the health of the indoor plants there are various factors that need to be monitored which include- soil moisture, sunlight, temperature, humidity, and soil pH. It is really difficult to monitor all of these factors manually. Therefore, our goal is to achieve automation in taking care of the houseplants by using various components (Arduino Nano, soil moisture sensor, submersible mini water pump, ESP8266 Wi-Fi module, BH1750 sensor, DHT11 sensor, pH sensor, water level sensor, and LiPo rechargeable battery, LED light) embedded in the Smart Pot and by creating a library of indoor plants with all its data points, so, that the Smart Pot works in accordance with the plant selected. It provides functionalities for automatic water pumping into plant soil in the right amount, sends a notification when the level of water in the storage tank of the pot is low, sends a notification to put the plant in the sunlight and to brings it back, uploads data on the cloud using internet through Wi-Fi. The data includes humidity, temperature, moisture, sunlight, and pH value which can be seen through the android application and important notifications are sent if the value of a data point is not healthy. It also shows the overall health of the plant in its mobile interface.

Sraboni Bhuiyan – 2019 et.al [2] proposed Smart Board for exactitude Farming victimization Wireless sensing element Network. Operating consists of FFD sensing element chiefly used for hearth detection through WIMAX. The information collected through multiple gateways is updated to server in web, to accessed anyplace within the world. The information present in web, displayed in good board and projected a wise sensing element system supported by actuators to change farming and supply exactitude farming expertise. The system helps person with inferior information of technology to grasp and maintain the system with a replacement device specifically good Board. The board helps to watch standing of the farm and to send action command to farm machinery. This work additionally helps to observe of e-governance by setting an easy however effective knowledge exchange between government and farmers.

Farmers can get benefitted because the good board can keep them up so far on government's agriculture connected announcements. It is a scientific mixture of the many technologies as well as a wise sensing element network system.

Dr. D. Saraswathi – 2018 et.al [3] proposed the technique known as farming. Farming refers to the art of growing plants in water (either saline) while not soil (land). Nutrients for the plants' area unit equipped to the roots within the style of answer that may be either within the style of static or flowing. Farming is cultivated each in inexperienced house and glass house setting. The limitation in inexperienced house setting is to take care of the temperature, pressure, timidness price at a specific level. additionally, thereto, observation on hydrogen ion concentration price and electrical physical phenomenon in farming is another challenge that must be monitored and maintained. Manual observation is in apply that could be a terribly trivial task else the plants might die out. This project, focuses on 2 tasks, the primary one is to automatize the inexperienced house setting observation. The next is automation of hydrogen ion concentration level and electrical physical phenomenon maintenance. IOT is employed to transfer the retrieved knowledge to web the net (mass storage) and mobile app is employed to speak this standing to the employment using internet to their mobile phones, in order that observation & maintenance are going to be easier.

Amandeep – 2018 et.al [4] proposed a novel wireless detector network technique in smart Farming victimization IOT to extend the productivity. The wireless detector network uses totally different detectors like temperature sensor that defects the temperature by the warmth created. Humidness detector detects the timidness according the cooling fan and wetness detector that senses the soil wetness. Extremist sonic obstacle detector for thievery detection, these data area unit sends to the most server. additionally, AVR Microcontroller At mega 16/32 want to monitor to regulate the system. Deployed ZigBee Module, that controls the pilotless vehicle and warehouse. Totally different blocks area unit utilized in this work, in block one GPS module is employed for spraying pesticides, scaring away the birds and animals and in block a pair of the AVR microcontroller is employed within which the detector detects and looking on the warmth and therefore the cooling method the motor switch ON and OFF method is allotted. Star PV panels area unit want to generate electricity which can offer inexperienced and pollution free power.

Elhassan Mohammed -2014 et.al [5] proposed the automated irrigation system designed and implementation. The system improved is beneficial and works in a very efficient apply. It decreases the water loss to a larger vary. It desires minimum maintenance the ability waste has been reduced greatly by victimization star cells. The system is applied in greenhouses. The System is incredibly helpful in fields wherever water deficiency could be a major downside. The crop's productivity improved and therefore the wastage of crops area unit greatly cut victimization this irrigation system. The automatic system is simpler and offers additional accomplishable results. The extension work is that the prediction of crop water desires victimization data processing algorithms within which we tend to area unit presently progressing. The prediction helps to provide the correct amount of irrigation to the crops.

Sudhir Rao Rupanagudi – 2019 et.al [6] proposed delineate tomato maturity grading system victimization image process technique, classification of the matures of fruit, supported its color and texture, forms a necessary method to be allotted by agriculturists and therefore the food process trade. As we've got ascertained that farmer is usually finite regarding these crops that however the expansion of the plant is going to be because it can't be picturized. Parts utilized in this system area unit soil wetness detector, photographic camera and chiefly MATLAB program is employed for Simulink. Primarily, photographic camera that is placed higher than the crop to capture the image through image process the image gets filtered according to its color. At next stage it will separate the red tomatoes because its growth is complete and divide it. The analysis been in deep trouble that remaining tomato per its maturity grading, color, texture, and image is shipped to the website or application, and additionally it'll send the small print of standing.

Savath Saypadith -2018 et.al [7] proposed a novel good crop cultivation observation system to grasp the expansion and cultivation of crops. Soil temperature humidness detector, water level detector, relay management and Wi-Fi wireless local area units' network WLAN wireless fidelity Wi-Fi local area network LAN module are wont to develop observation system. Wireless fidelity module has been connected to sensors specified soil and temperature and water level is been management led by relay control and magnet valve is been introduced to electromechanical operated valve.

REQUIREMENT SPECIFICATION:

HARDWARE:

NODEMCU ESP8266:

The NodeMCU ESP8266 is a popular development board based on the ESP8266 Wi-Fi module. It is designed to make it easier for developers and hobbyists to create IoT (Internet of Things) projects and prototypes. Here are some key details about the NodeMCU ESP8266:

1. **ESP8266 Module:** The core of the NodeMCU is the ESP8266 Wi-Fi module. The ESP8266 is a low-cost Wi-Fi chip with full TCP/IP stack and microcontroller capabilities. It is commonly used for IoT applications due to its small size, low power consumption, and affordability.
2. **Development Board:** The NodeMCU board is a development platform built around the ESP8266 module. It includes a USB-to-serial converter for programming and debugging, as well as GPIO pins for connecting sensors, actuators, and other electronic components.
3. **Arduino Compatibility:** NodeMCU has gained popularity in the Arduino community, and it can be programmed using the Arduino IDE with the help of the ESP8266 Arduino Core. This allows developers to write code using the familiar Arduino programming environment.
4. **Wi-Fi Connectivity:** The ESP8266 module on NodeMCU provides Wi-Fi connectivity, making it suitable for projects that require internet connectivity. It can connect to local Wi-Fi networks, making it easy to send and receive data over the internet.
5. **GPIO Pins:** NodeMCU features several General-Purpose Input/Output (GPIO) pins that can be used to connect sensors, LEDs, displays, and other hardware components. These pins can be controlled and read using programming, allowing you to interface with various devices.
6. **Power Supply:** NodeMCU can be powered via a micro-USB port, making it easy to connect to a computer or a USB charger. It can also be powered using an external power source or a battery.
7. **Additional Features:** Some NodeMCU variants come with additional features like built-in LED indicators, reset buttons, and voltage regulators, making it even more convenient for prototyping.
8. **Community and Libraries:** There is a vast online community of NodeMCU users and developers who share projects, tutorials, and libraries. This community support can be valuable for beginners and experienced developers alike.

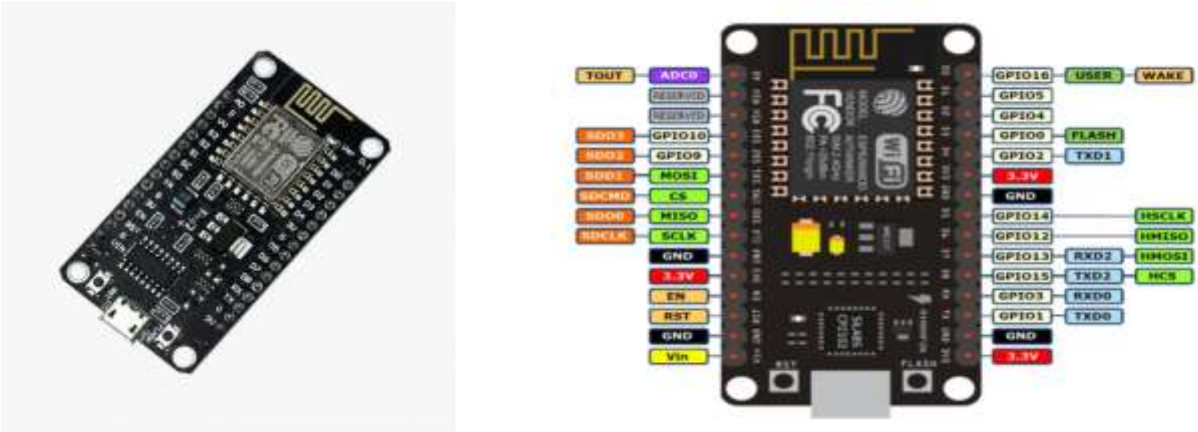


Fig 1: NodeMCU ESP8266.

<u>DHT11 SENSOR</u>:

The DHT11 is a low-cost, digital temperature and humidity sensor. It is commonly used in electronics projects and IoT (Internet of Things) applications to measure and monitor environmental conditions. Here are some key details about the DHT11 sensor:

1. **Measurement Capabilities:**
   - **Temperature:** The DHT11 can measure temperature within a range of 0°C to 50°C (32°F to 122°F) with an accuracy of ±2°C.
   - **Humidity:** It can measure relative humidity in the range of 20% to 80% with an accuracy of ±5%.

2. **Digital Output:** The DHT11 provides a digital output, which means it directly provides temperature and humidity values in a digital format that can be read by a microcontroller or other digital devices.

3. **Single-Wire Communication:** Communication with the DHT11 sensor is achieved using a single-wire protocol. This means you only need one data pin to send and receive data from the sensor.

4. **Operating Voltage:** The DHT11 typically operates at 3.3V or 5V, making it compatible with a wide range of microcontrollers and development boards.

5. **Library Support:** There are libraries available for various microcontroller platforms, such as Arduino, Raspberry Pi, and others, to simplify the process of interfacing with and reading data from

the DHT11 sensor.

6. **Applications:** The DHT11 is commonly used in DIY weather stations, home automation systems, greenhouse monitoring, and other projects where basic temperature and humidity data are needed.

Limitations:

- The DHT11 has limited accuracy compared to more advanced sensors like the DHT22 or the DHT21. If precise temperature and humidity measurements are required, you may consider using a more accurate sensor.
- It has a relatively slow update rate, which means it may not be suitable for applications that require rapid or continuous monitoring of environmental conditions.

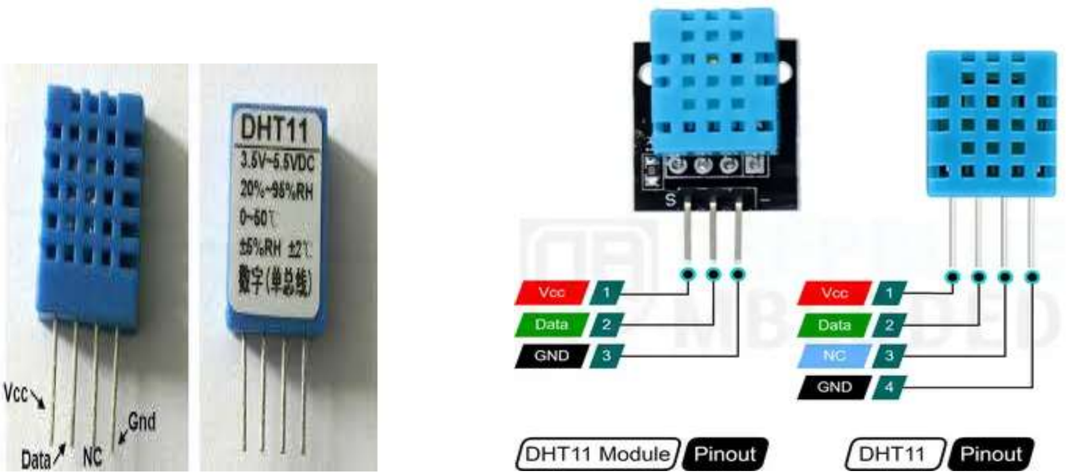


Fig 2: DHT11 Sensor.

BREADBOARD:

A breadboard (sometimes called a plug block) is used for building temporary circuits. It is useful to designers because it allows components to be removed and replaced easily. It is useful to the person who wants to build a circuit to demonstrate its action, then to reuse the components in another circuit.

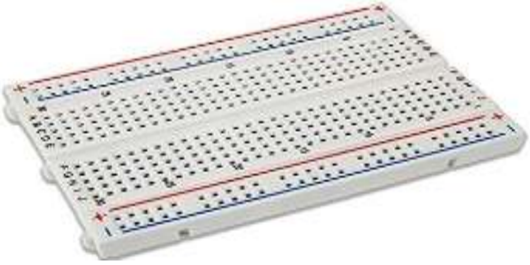


Fig 3: Breadboard.

JUMPER WIRES:

A jump wire (also known as jumper, jumper wire, DuPont wire) is an electrical wire, or group of them in a cable, with a connector or pin at each end (or sometimes without them – simply "tinned"), which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering. Individual jump wires are fitted by inserting their "end connectors" into the slots provided in a breadboard, the header connector of a circuit board, or a piece of test equipment.

Fig 4: Jumper Wires.

POT:

RELAY:

A relay is an electromechanical device that is used to control electrical circuits by opening or closing contacts in response to a signal. Relays are essential components in a wide range of applications, including industrial automation, home automation, electronics, and more. Here are the details of how a relay works and its key characteristics:

1. Basic Operation:

- A relay consists of an electromagnet (coil) and a set of electrical contacts.
- When an electrical current is applied to the coil, it creates a magnetic field around it.
- This magnetic field causes a movable armature or contact to be attracted towards the coil or, in some cases, repelled away from it.
- When the armature moves, it either closes (makes) or opens (breaks) a set of electrical contacts, depending on the type of relay.

2. Types of Relays:

- SPST (Single-Pole Single-Throw): This type of relay has one normally open (NO) or normally closed (NC) contact. When the coil is energized, it either opens or closes the contact.

- SPDT (Single-Pole Double-Throw): SPDT relays have one common terminal and two contacts one NO and one NC. When the coil is energized, it switches between the two contacts.

- DPST (Double-Pole Single-Throw): DPST relays have two separate single-pole contacts that operate together. They are used for controlling two independent circuits.

- DPDT (Double-Pole Double-Throw): DPDT relays have two separate single-pole double-throw contacts. They can be used for controlling two circuits with both NO and NC options.

Fig 4: Relay.

SOIL MOISTURE:

Soil moisture sensors measure the humidity of water content in soil. Since the direct hydrometric measuring of free-soil wetness needs removing, drying, and coefficient of a sample, soil wetness sensors live the meter water content indirectly by victimization another property of the soil, like electrical phenomenon, non-conductor constant, or interaction with neutrons, as a proxy for the wetness content.

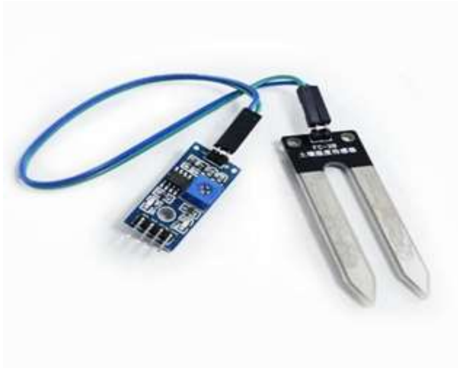


Fig 5: Soil Moisture.

BATTERY:

An electric battery is a device consisting of one or more electrochemical cells with external connections provided to power electrical devices such as flashlights, and electric cars. When a battery is supplying electric power, its positive terminal is the cathode and its negative terminal is the anode. The terminal marked negative is the source of electrons that will flow through an external electric circuit to the positive terminal. When a battery is connected to an external electric load,

Fig 6: Battery.

DC MOTOR:

An DC motor is an electrical motor driven by Associate in direct current (DC). In figure: 5, The DC motor normally consists of two basic components, an outdoor stationary stator coil having coils furnished with DC to supply a rotating flux, and an indoor rotor connected to the output shaft manufacturing a second rotating flux. The rotor flux could also be made by permanent magnets, reluctance striking, or DC or AC electrical windings.

Fig 7: Dc Motor.

PIPE:

Here it is used as a water channel, and pipe has been used for watering plant.

Fig 8: Pipe.

SOFTWARE:

Embedded C is an extension of C language and it is used to develop micro-controller-based applications. The extensions in the Embedded C language from normal C Programming Language are the I/O Hardware Addressing, fixed-point arithmetic operations, accessing address spaces, etc. Embedded C Program has five layers of Basic Structures. They are:

Comment: These are simple readable text, written in code to make it more understandableto the user. Usually comments are written in // or /* */.

Pre-processor directives: The Pre-Processor directives tell the compiler which files to lookin to find the symbols that are not present in the program.

Global Declaration: The part of the code where global variables are defined.

Local Declaration: The part of the code where local variables are defined.

Main function: Every C program has a main function that drives the whole code. It basicallyhas two parts the declaration part and the execution part. Where, the declaration part is whereall the variables are declared, and the execution part defines the whole structure of execution in the program. In nature it uses a cross-platform development scheme, i.e., the development of the application by it is platform-independent and can be used on multiple platforms.

ARDUINO IOT CLOUD:

Arduino IoT Cloud is a platform provided by Arduino that simplifies the process of connecting and managing Internet of Things (IoT) devices. It allows you to create IoT projects securely and easily and remotely monitor and control your connected devices through a user-friendly web interface or mobile app. Here's how Arduino IoT Cloud typically works:

- Device Setup:
  We start by setting up your IoT device, which can be an Arduino board (like Arduino MKR WiFi 1010 or Arduino Nano 33 IoT) equipped with various sensors, actuators, or communication modules.
- Arduino IoT Cloud Account:
  We need to create an account on the Arduino IoT Cloud platform, which provides the web interface for managing your devices and data.
- Arduino IoT Cloud Library:
  We include the Arduino IoT Cloud library in your Arduino IDE sketch. This library facilitates communication between your Arduino board and the IoT Cloud platform.
- Device Configuration:
  Within the Arduino IDE, you define the properties of your IoT device, including the variables (such as sensor readings) you want to monitor and the actuators (like LED controls) you want to control remotely.
- Upload Code:
  We write the Arduino sketch that reads sensor data and controls actuators, and you upload it to your IoT device.
- Device Connection:

IoT device connects to the Arduino IoT Cloud using the built-in Wi-Fi or other connectivity options provided by the device.

- Data Synchronization:
  The Arduino IoT Cloud platform continuously synchronizes data between your IoT device and the cloud. Sensor data is sent from the device to the cloud, and you can view it in real-time on the IoT Cloud web interface.
- Remote Control:
  Using the web interface or mobile app, you can remotely control actuators on your IoT device. For example, you can turn on or off an LED, trigger a servo motor, or control other connected devices.
- Automation and Notifications:
  Arduino IoT Cloud allows you to set up automation rules and notifications based on the data received from your devices. For example, you can create rules to receive alerts when a sensor reading exceeds a certain threshold.
- Data Visualization:
  We can visualize the historical data of your IoT devices through graphs and charts on the IoT Cloud platform. This helps in tracking trends and making informed decisions.
- Security:
  Arduino IoT Cloud provides secure communication between your devices and the cloud platform to protect your data and device control.

Overall, Arduino IoT Cloud simplifies the process of building, connecting, and managing IoT projects, making it accessible to makers, hobbyists, and developers without extensive IoT expertise. It offers an integrated ecosystem for creating IoT applications that involve data monitoring, control, and automation.

ARDUINO IDE:

The Arduino IDE is an open-supply software, that is used to write and add code to the Arduino boards. The IDE utility is appropriate for one of a kind working structuresconsisting of Windows, Mac OS X, and Linux. It supports the programming languages like Cand C++. Here, IDE stands for Integrated Development Environment.

The software design of the Sensor Interface is composed of 2 major components: the code describing the microcontroller program and the code describing the LCD application. The microcontroller code is responsible for managing the sensors that are connected to it, as well as their readings. Determining which sensor ports to read from, It should display information about all the sensors connected to the microcontroller, as well as allow the user to add sensors to monitor, delete sensors, and change sensor properties.

The testing objectives are as follows:

- Testing is the process of executing the program with the intent of finding an error.
- A successful test is the one with a high chance of detecting an error.
- Testing cannot prove that there are not any flaws.

TEST TYPES

Unit Testing: It mainly focuses on verification effort on the unit of software design. The interface of each of the module are tested to ensure proper flow of the information in and out of the modules under consideration. Boundary conditions are checked.

Integration Testing: This is the systematic technique for construction of the program structured while at the same time condition test to uncover errors associated with the interfacing. Data can be lost across an interface. One service module subsidiary function may have an unfavorable effect on another's; when combined, they may not generate the desired principal function; and worldwide data models might cause issues.

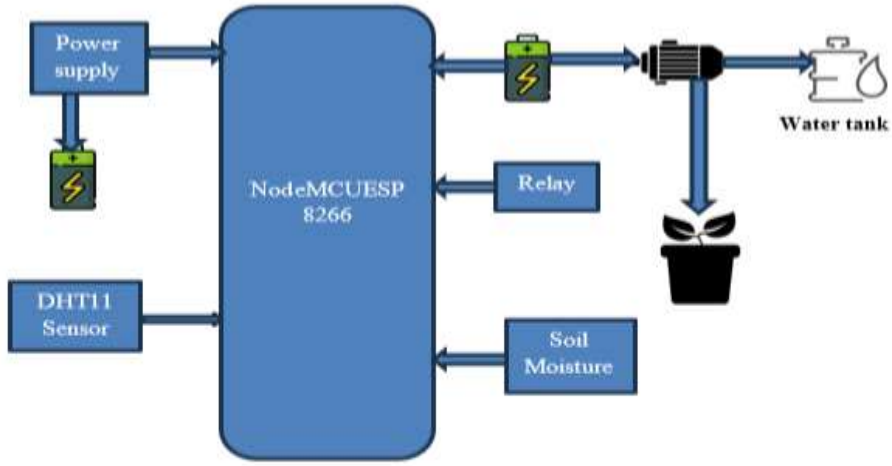
The main difficulty that arises in integration testing is localizing errors that are discovered.

System testing: After the integration testing, the software is completely assembled as package, interfacing errors has been uncovered and corrected. System testing involves putting all the modules together and checking the entire software. It is useful in checking whether for the given input desired output is achieved as a result or not. This allows to check whether all independent path within a module have been exercised at least once. Acceptance testing: User acceptance testing is a critical phase of any project and requires significant participation by end user.

Chapter 3

THE ANALYSIS AND DESIGN

High level:



Low-Level:

The soil moisture sensor module used here have two output pins (Digital output and Analog output). The output from the probe of the moisture sensor is compared with a reference value using a lm393 comparator. The reference value can be changed by turning the potentiometer in the module. The digital pin gives an active low output when the soil is wet. Here we are using the analog output from the module by connecting it to one of the analog pins of Node Mcu. While using the analog output the wet detection value can be set/adjusted within the program itself.

As shown in the circuit diagram, a float switch is connected to one of the analog pins of Node Mcu and a 1K Ohm resistor is used to pulled up the line. Analog pins of Node Mcu can also be used as digital inputs. The status of the tank is identified by checking the output of the float switch. Node Mcu reads the voltage dropped across the pull up resistor for sensing the level of water in the tank. Two LEDs are connected to the 2nd and 3rd pin of Node Mcu to show the moisture status and tank status respectively. And the 4th pin links to the base of a BC547 transistor which in turn drives the 12 V DC motor.

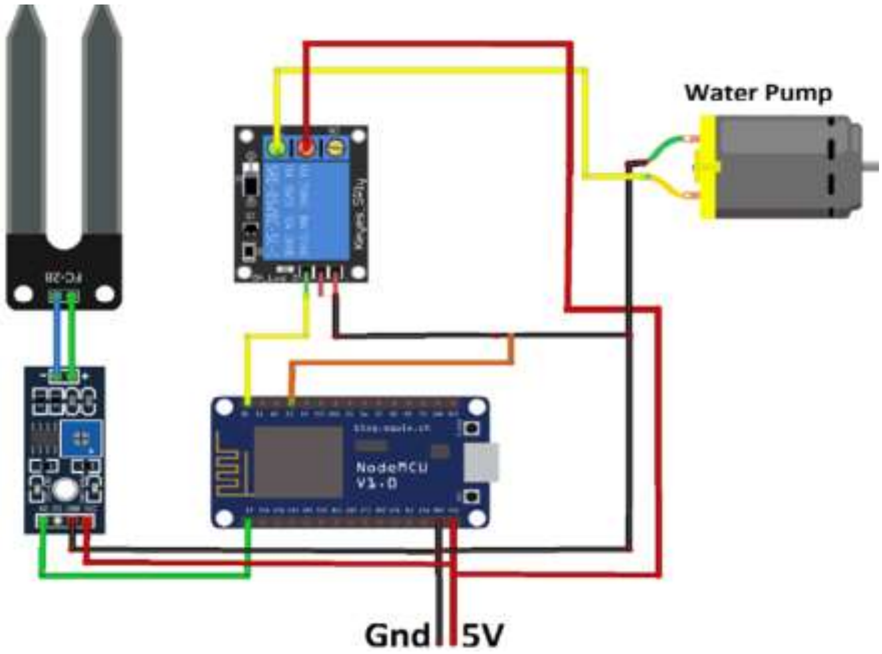


Fig 9: Circuit diagram of system architecture.

ESP8266:

ESP8266 comes with capabilities of

- 2.4 GHz Wi-Fi (802.11 b/g/n, supporting WPA/WPA2), general-purpose input/output (16 GPIO), Inter-Integrated Circuit (I²C) serial communication protocol, analog-to-digital conversion (10-bit ADC)
- Serial Peripheral Interface (SPI) serial communication protocol, I²S (Inter-IC Sound) interfaces with DMA (Direct Memory Access) (sharing pins with GPIO),
- UART (on dedicated pins, plus a transmit-only UART can be enabled on GPIO2), and pulse-width modulation (PWM).
- It employs a 32-bit RISC CPU based on the TensilicaXtensa L106 running at 80 MHz (or overclocked to 160 MHz). It has a 64 KB boot ROM, 64 KB instruction RAM and 96 KB data RAM. External flash memory can be accessed through SPI.
- To communicate with the ESP8266 module, microcontroller needs to use set of AT commands. Microcontroller communicates with ESP8266-01 module using UART having specified Baud rate.

- There are many third-party manufacturers that produce different modules based on this chip. So, the module comes with different pin availability options like, ESP-01 comes with 8 pins (2 GPIO pins) – PCB trace antenna. (Shown in above figure) ESP-02 comes with 8 pins, (3 GPIO pins) – U-FL antenna connector.
- ESP-03 comes with 14 pins, (7 GPIO pins) – Ceramic antenna.
- ESP-04 comes with 14 pins, (7 GPIO pins) – No ant.

<u>ESP8266-01 Module Pin Description</u>

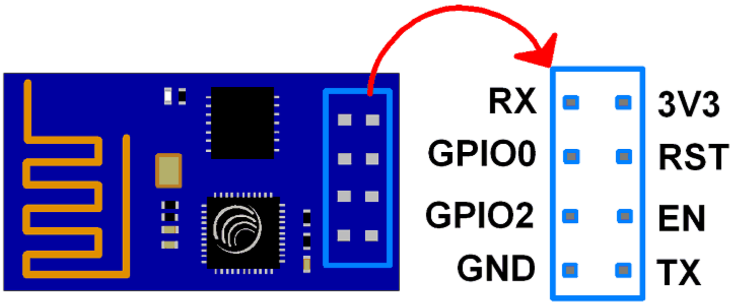


Fig 10: ESP8266-01 Module Pins.

- 3V3: - 3.3 V Power Pin.
- GND: - Ground Pin.
- RST: - Active Low Reset Pin.
- EN: - Active High Enable Pin.
- TX: - Serial Transmit Pin of UART.
- RX: - Serial Receive Pin of UART.

GPIO0 & GPIO2: General Purpose I/O Pins. These pins decide what mode (boot or normal) the module starts up in. It also decides whether the TX/RX pins are used for Programming the module or for serial I/O purpose.

To program the module using UART, Connect GPIO0 to ground and GPIO2 to VCC or leave it open. To use UART for normal Serial I/O leave both the pins open (neither VCC nor Ground).

Now let us connect ESP8266 module to computer with RS232 standard serial port (using USB to Serial converter in case of laptop) as shown in below figure.

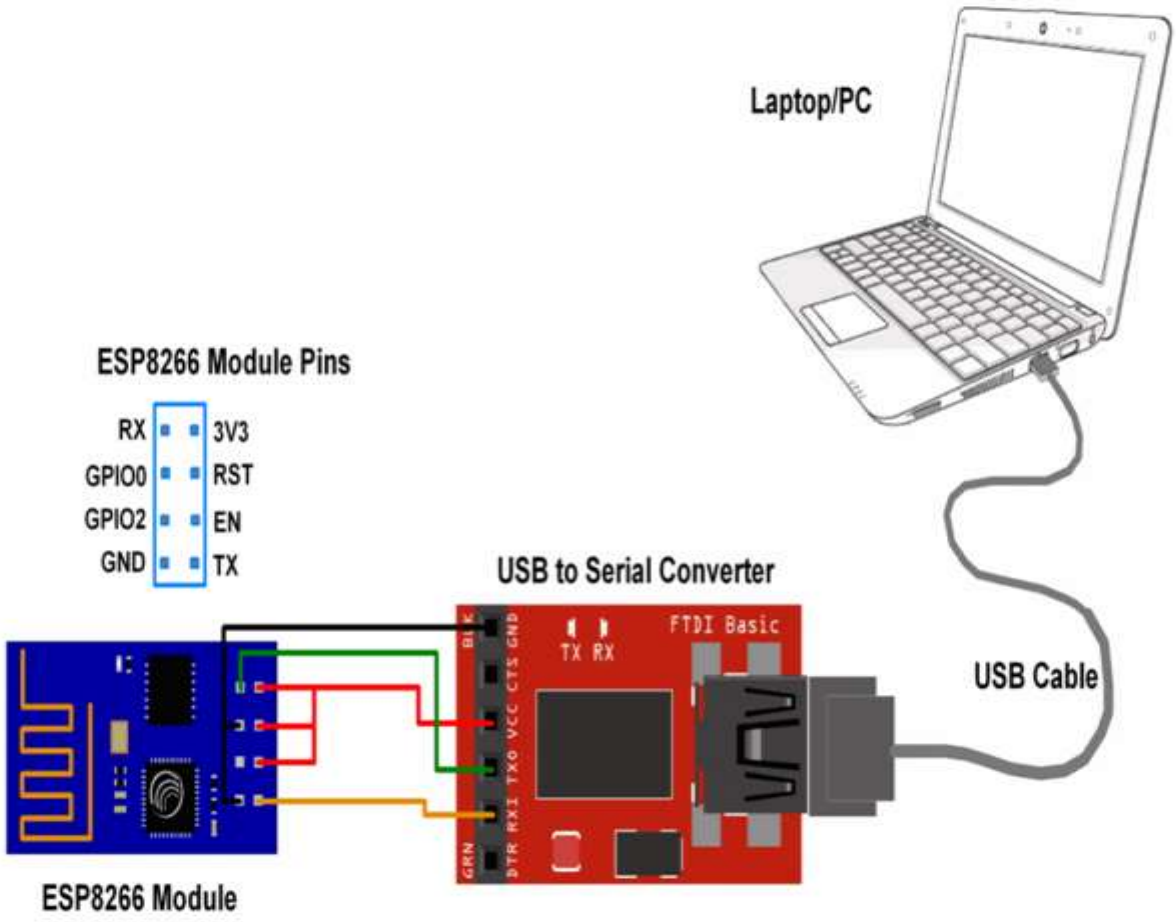


Fig 11: The connection to check and testing the NodeMCU to Port.

Pin Definition

Label	GPIO	Input	Output	Notes
D0	**GPIO16**	no interrupt	no PWM or I2C support	HIGH at boot used to wake up from deep sleep
D1	**GPIO5**	OK	OK	often used as SCL (I2C)
D2	**GPIO4**	OK	OK	often used as SDA (I2C)
D3	**GPIO0**	pulled up	OK	connected to FLASH button, boot fails if pulled LOW
D4	**GPIO2**	pulled up	OK	HIGH at boot connected to on-board LED, boot fails if pulled LOW
D5	**GPIO14**	OK	OK	SPI (SCLK)
D6	**GPIO12**	OK	OK	SPI (MISO)
D7	**GPIO13**	OK	OK	SPI (MOSI)
D8	**GPIO15**	pulled to GND	OK	SPI (CS) Boot fails if pulled HIGH
RX	**GPIO3**	OK	RX pin	HIGH at boot
TX	**GPIO1**	TX pin	OK	HIGH at boot debug output at boot, boot fails if pulled LOW
A0	**ADC0**	Analog Input	X	

Note that, to put ESP8266 in flash mode, make connections as per above figure (in between ESP8266 and USB to Serial converter) and then only connect it to PC/laptop. Do not forget to connect GPIO0 pin to ground.

Then click on START tab in ESP8266 download tool, and wait till it finishes. After finishing flash process, disconnect ESP8266 module from PC/laptop and remove ground connection at GPIO0 pin.

How to start with NodeMCU?

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 first we 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 we can easily get our custom NodeMCU firmware as per our requirement.

NodeMCU:

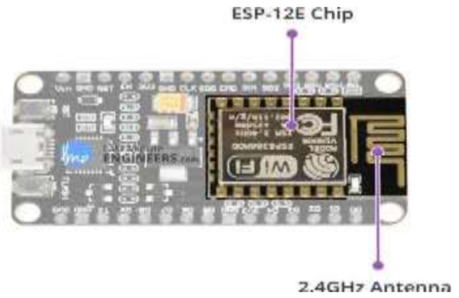
The Internet of Things (IoT) has been a trending field in the world of technology. It has changed the way we work. Physical objects and the digital world are connected now more than ever. Keeping this in mind, Espressif Systems (A Shanghai-based Semiconductor Company) has released an adorable, bite-sized Wi-Fi enabled microcontroller – **ESP8266**, at an unbelievable price! For less than $3, it can monitor and control things from anywhere in the world – **perfect for just about any IoT project**.

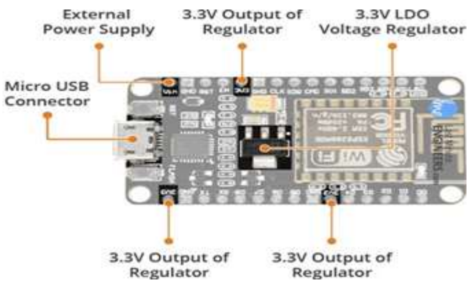
ESP-12E Module

The development board equips the ESP-12E module containing ESP8266 chip having **Tensilica Xtensa® 32-bit LX106 RISC microprocessor** which operates at **80 to 160 MHz** adjustable clock frequency and supports **RTOS**.

ESP-12E Chip

- Tensilica Xtensa® 32-bit LX106
- 80 to 160 MHz Clock Freq.
- 128kB internal RAM
- 4MB external flash.
- 802.11b/g/n Wi-Fi transceiver.





There is also **128 KB RAM and 4MB of Flash memory** (for program and data storage) just enough to cope with the large strings that make up web pages, JSON/XML data, and everything we throw at IoT devices nowadays.

The ESP8266 Integrates **802.11b/g/n HT40 Wi-Fi transceiver**, so it can not only connect to a WiFi network and interact with the Internet, but it can also set up a network of its own, allowing other devices to connect directly to it. This makes the ESP8266 NodeMCU even more versatile.

Power Requirement

As the operating voltage range of ESP8266 is **3V to 3.6V**, the board comes with a LDO voltage regulator to keep the voltage steady at 3.3V. It can reliably supply up to 600mA, which should be more than enough when ESP8266 pulls as much as **80mA during RF transmissions**. The output of the regulator is also broken out to one of the sides of the board and labelled as 3V3. This pin can be used to supply power to external components.

Power Requirement:

- Operating Voltage: 2.5V to 3.6V
- On-board 3.3V 600mA regulator

- 80mA Operating Current
- 20 µA during Sleep Mode.

Power to the ESP8266 NodeMCU is supplied via the on-board MicroB USB connector. Alternatively, if you have a regulated 5V voltage source, the VIN pin can be used to directly supply the ESP8266 and its peripherals.

Demonstration:

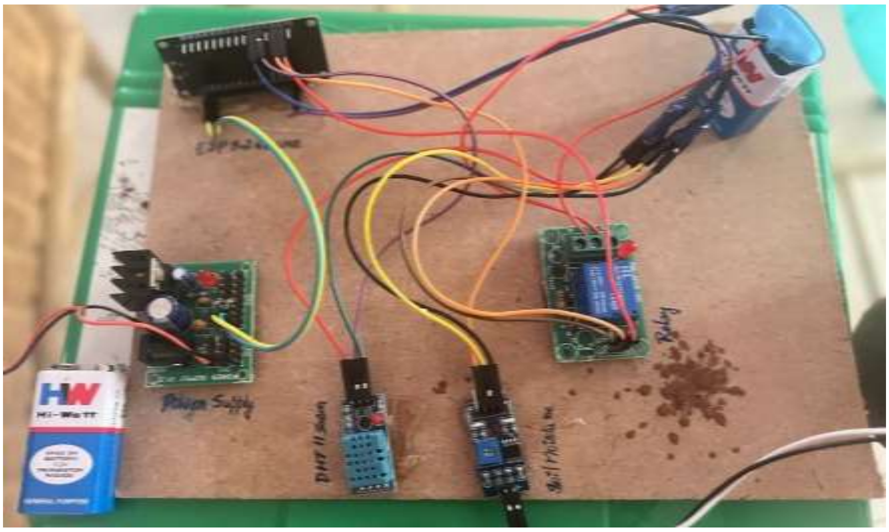


Fig 12: Shows the connection.

Fig 13: shows the connections with running status.

Chapter 4

IMPLEMENTATION

ALGORITHM:

The code appears to be for an IoT project that involves monitoring and controlling humidity, temperature, and soil moisture using a DHT11 sensor and a relay. It also communicates with the Arduino IoT Cloud for remote monitoring and control.
Here's a step-by-step algorithm for the code:

1.Include Libraries: Include the required libraries at the beginning of the code. In this case, you're using the "thingProperties.h" library and the "DHT" library for the DHT11 sensor.

2.Declare Pin Variables: Define variables to represent the pins used for various components, such as the moisture sensor (moistsen), relay (to control a pump or other device), and the DHT11 sensor (DHT11PIN).

3.Initialize DHT Sensor: Create an instance of the DHT sensor with the specified pin and sensor type (DHT11). Initialize the sensor using HT.begin().

4.Setup Function:
Initialize Serial communication for debugging.
Set the DHT11PIN as an input.
Initialize the DHT sensor.
Set the pinMode for the moisture sensor and relay.
Set the relay to the LOW state (initially off).
Add a delay to allow time for the Serial Monitor to open.

5.Initialize IoT Cloud and Debugging:
Initialize IoT Cloud using ArduinoCloud.begin(ArduinoIoTPreferredConnection).
Set the debug message level using setDebugMessageLevel().
Print debug information using ArduinoCloud.printDebugInfo().

6.Loop Function:

Call functions to handle changes in humidity, pump state, and moisture.

Call ArduinoCloud.update() to communicate with the IoT Cloud.

7.onPumpChange Function:

This function is called when the state of the "pump" variable changes.

If the "pump" variable is true, set the relay to HIGH (turn the pump on). Otherwise, set it to LOW (turn the pump off).

8.onMoistureChange Function:

Read the analog value from the moisture sensor using analogRead().

Invert the moisture value (subtract it from 1024) to get a higher value for wetter soil.

Map the moisture value from the 0-1024 range to a 0-100 range (percentage).

Print the moisture values for debugging.

Update the "moisture" variable with the calculated moisture percentage.

Add a delay to avoid rapid updates.

9.onHumidityChange Function:

Read temperature and humidity values from the DHT11 sensor using HT.readTemperature() and HT.readHumidity().

Print the humidity and temperature values for debugging.

Update the "humidity" and "temperature" variables with the sensor readings.

Add a delay to avoid rapid updates.

10.onTemperatureChange Function: (Note: This function is empty and does not contain any code).

The code appears to be designed to work with the Arduino IoT Cloud, allowing remote monitoring and control of the connected devices. Make sure you have set up the IoT Cloud and configured the variables accordingly in the "thingProperties.h" file for your specific project requirements.

CODE:

```
#include "thingProperties.h"
#include<DHT.h>
int moistsen=A0;
int relay=D2;
 #define Type DHT11
 int DHT11PIN=D1;
 DHT HT(DHT11PIN, Type);
 float h,t;
void setup() {
  // Initialize serial and wait for port to open:
  Serial.begin(9600);
   pinMode(DHT11PIN, INPUT);
  HT.begin();
  pinMode(moistsen,INPUT);
  pinMode(relay,OUTPUT);
  digitalWrite(relay,LOW);
  // This delay gives the chance to wait for a Serial Monitor without blocking if none is found
  delay(1500);
  // Defined in thingProperties.h
  initProperties();
  // Connect to Arduino IoT Cloud
  ArduinoCloud.begin(ArduinoIoTPreferredConnection);
  /*
     The following function allows you to obtain more information
     related to the state of network and IoT Cloud connection and errors
     the higher number the more granular information you’ll get.
     The default is 0 (only errors).
     Maximum is 4
 */
  setDebugMessageLevel(2);
  ArduinoCloud.printDebugInfo();
}
```

```
void loop() {
  onHumidityChange();
  onPumpChange();
  onMoistureChange();
  ArduinoCloud.update();
  // Your code here


}
/*
  Since Pump is READ_WRITE variable, onPumpChange() is
  executed every time a new value is received from IoT Cloud.
*/
void onPumpChange()  {
  // Add your code here to act upon Pump change
  if(pump){
    digitalWrite(relay,HIGH);
  }
  else
  {
    digitalWrite(relay,LOW);
  }
}
/*
  Since Moisture is READ_WRITE variable, onMoistureChange() is
  executed every time a new value is received from IoT Cloud.
*/
void onMoistureChange()  {
 int moist_value=analogRead(moistsen);
 moist_value=1024-moist_value;
 int moist_value1=map(moist_value,0,1024,0,100);
 Serial.println("moist_val="+String(moist_value));
 Serial.println("moist_val1="+String(moist_value1));
 moisture=moist_value1;
 delay(1000);
}
/*
```

```
  Since Humidity is READ_WRITE variable, onHumidityChange() is
  executed every time a new value is received from IoT Cloud.
*/
void onHumidityChange()
{
  t = HT.readTemperature();
  h=HT.readHumidity();
  Serial.println("humidity="+String(h));
  Serial.println("temp="+String(t));
  temperature=t;
  humidity=h;
  delay(1000);

}
void onTemperatureChange(){

}
```

Chapter 5

RESULT AND DISCUSSIONS

In conclusion, the integration of IoT technology in a smart pot for monitoring and managing plant health is a remarkable step towards more efficient and informed gardening practices. By measuring crucial parameters such as moisture levels, humidity, and temperature, this system empowers gardeners to make data-driven decisions and create optimal growing conditions for their plants. Storing this valuable data in the Arduino IoT Cloud not only provides real-time access to the plant's status but also offers the convenience of remote monitoring and control. As technology continues to intersect with agriculture, solutions like the smart pot demonstrate how IoT innovations can enhance our connection with nature, foster sustainability, and nurture healthier and more vibrant plants. This combination of green thumb expertise and IoT sophistication paves the way for a greener and smarter future in gardening and plant care.

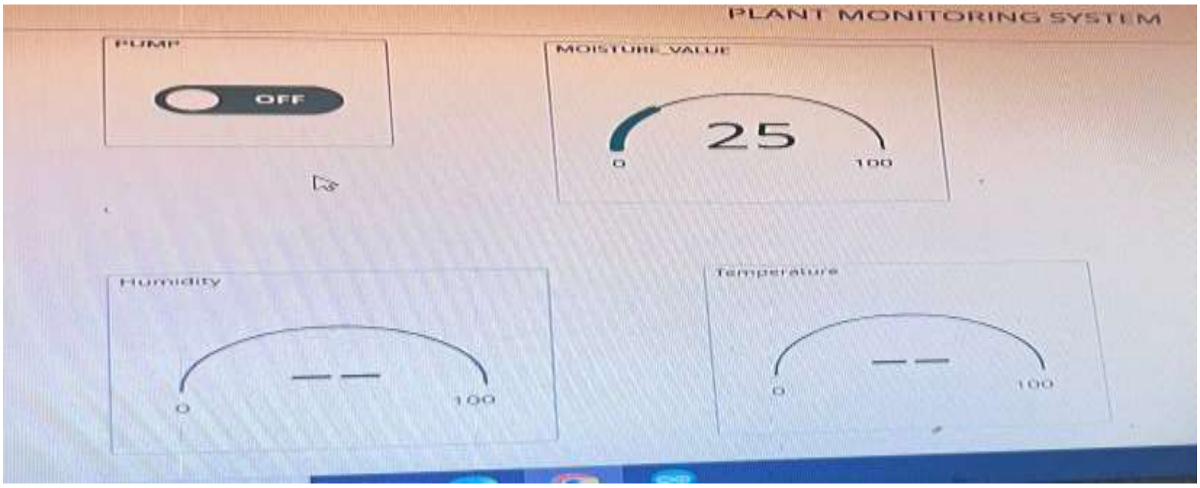


Fig 13: Monitering the mositure level when switch is in off state

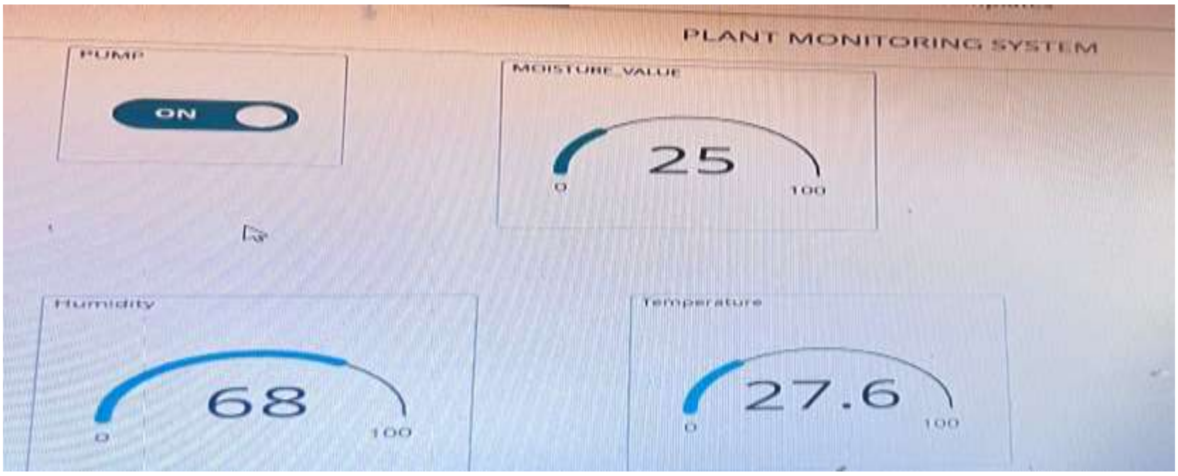


Fig 14: Monitoring the mositure level,Humidity,Temperature when switch is in on state.

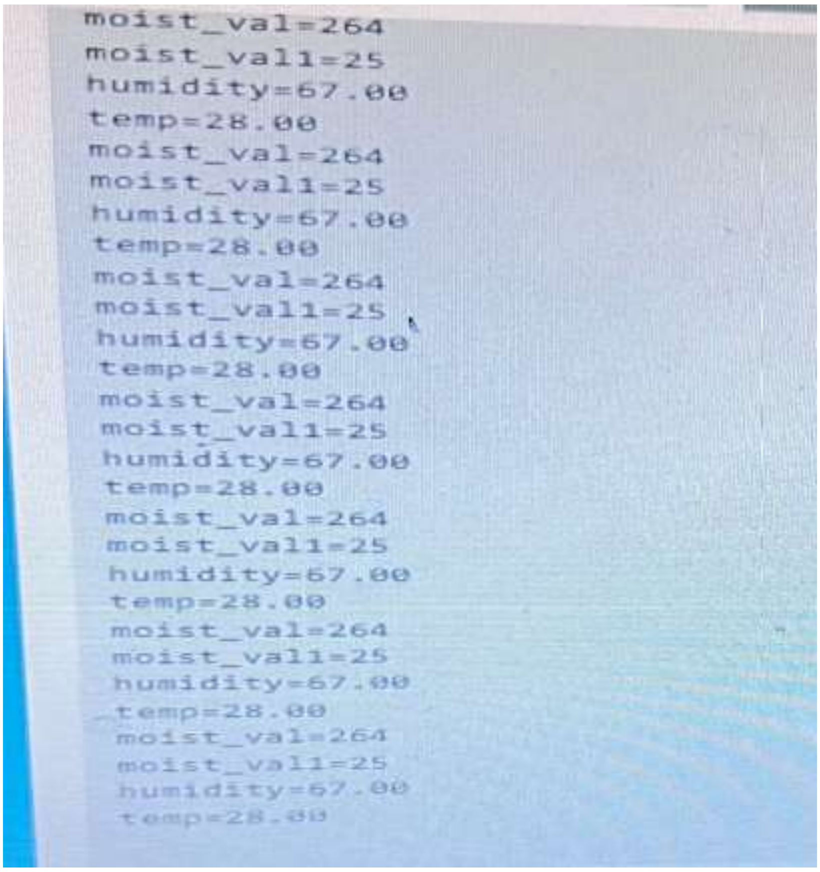


Fig 14: Monitoring the data in ArduinoIOT Cloud

Chapter 6

CONCLUSION

Thus, the soil moisture using Node Mcu has been designed and tested successfully. It has been developed by integrated features of all the hardware components used. Presence of every module has been reasoned out and placed carefully, thus contributing to the best working of the unit. Thus, the Node Mcu Based Automatic Plant Watering System has been designed and tested successfully. The system has been tested to function automatically. The moisture sensors measure the moisture level (water content) of the different plants. If the moisture level is goes to be below the desired and limited level, the moisture sensor sends the signal to the Node Mcu board which triggers the Water Pump to turn ON and supply the water to respective plant. When the desired moisture level is reached, the system halts on its own and the water Pump is turned OFF. Thus, the functionality of the entire system has been tested thoroughly and it is said to function successfully.

REFERENCES

1. Rawal, Kush and Gabrani, Goldie, IoT based Computing to Monitor Indoor Plants by using Smart Pot (March 28, 2020). Proceedings of the International Conference on Innovative Computing & Communications (ICICC) 2020, Available at SSRN: https://ssrn.com/abstract=3562964 or http://dx.doi.org/10.2139/ssrn.3562964.
2. Maherin Mizan Maha, Md Masuduzzaman and Sraboni Bhuiyan, "Smart Board for Precision Farming Using Wireless SensorNetwork" International Conference on Robotics,Electrical and Signal Processing Techniques 2019 (ICREST).
3. Dr.D.Saraswathi, P.Manibharathy and R.Gokulnath "Automation of Hydroponics Green House Farming using IOT" IEEE International Conference on System, Computation, Automation and Networking 2018 (ICSCA).
4. "Smart Farming Using IoT," by Amandeep, Arshia Bhattacharjee, and Paboni Das, 8th IEEE Annual Information Technology, Electronics, and Mobile Communication Conference 2017. (IEMCON).
5. Elhassan Mohammed Elhassan Ahmed and Ibrahim khider Eltahir "Farm Automation based on IoT" International Conference on Computer, Control, Electrical, and Electronics Engineering 2018 (ICCCEEE).
6. Sudhir Rao Rupanagudi;B.S. Ranjani;Prathik Nagaraj;Varsha G Bhat "A cost effective tomato maturity grading system using image processing for farmers" International Conference on Contemporary Computing and Informatics (IC3I) 2014.
7. Khampheth Bounnady; Poutthasone Sibounnavong and Savath Saypadith "Smart Crop Cultivation Monitoring System by Using IoT" 5th International Conference on Engineering, Applied Sciences and Technology 2019 (ICEAST).