

DESIGN AND IMPLEMENTATION OF A DISTRIBUTED SYSTEM FOR THE MONITORING OF AN AIR-GROUND WIRELESS SENSOR NETWORK FOR CROP MONITORING

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

By

K. NAGA NARENDRA (Reg.No: 40130266)

V. AVINASH (Reg.No: 40130244)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING**

SCHOOL OF ELECTRICAL AND ELECTRONICS ENGINEERING

**SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
CATEGORY -1 UNIVERSITY BY UGC**

**Accredited with Grade "A++" by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119**

APRIL - 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

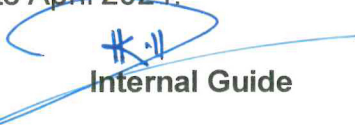
Accredited "A++" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of KALAPALA NAGA NARENDRA (40130266), VALLURI AVINASH (40130244), who carried out the project entitled "DESIGN AND IMPLEMENTATION OF A DISTRIBUTED SYSTEM FOR THE MONITORING OF AN AIR-GROUND WIRELESS SENSOR NETWORK FOR CROP MONITORING" under my supervision from November 2023 to April 2024.


Internal Guide

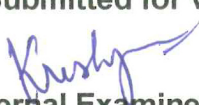
DR. KRUTIDEEPA BHOL, M.TECH., Ph.D.,

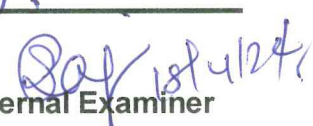

Head of the Department

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

Submitted for Viva voice Examination held on

18/4/24


Internal Examiner


External Examiner

DECLARATION

We, KALAPALA NAGA NARENDRA (40130266), VALLURI AVINASH (40130244), hereby declare that the Project Report entitled **“DESIGN AND IMPLEMENTATION OF A DISTRIBUTED SYSTEM FOR THE MONITORING OF AN AIR-GROUND WIRELESS SENSOR NETWORK FOR CROP MONITORING”** done by us under the guidance of **DR. KRUTIDEEPA BHOL, M.TECH., Ph.D.**, is submitted in partial fulfillment of the requirements for the award of Bachelor Engineering degree in Electronics and Communication Engineering.

1) 

2) 

DATE: 18/04/2024

PLACE: Chennai

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge our sincere thanks to **The Board of Management of Sathyabama Institute of Science and Technology** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

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

I would like to express our sincere and deep sense of gratitude to our Project Guide **Dr. T. RAVI, M.E., Ph.D., Head of the Department, Department of Electronics and Communication Engineering** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of the project work.

I 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.

I express our gratitude to our parents for their constant encouragement and support for the completion of the project.

ABSTRACT

India's population is reached beyond 1.2 billion and the population rate is increasing day by day then after 25-30 years there will be serious problem of food, so the development of agriculture is necessary. Today, the farmers are suffering from the lack of rains and scarcity of water. The main objective of this paper is to provide an automatic irrigation system thereby saving time, money & power of the farmer. The traditional farm-land irrigation techniques require manual intervention. With the automated technology of irrigation, the human intervention can be minimized. Whenever there is a change in temperature and humidity of the surroundings these sensors sense the change in temperature and humidity and gives an interrupt signal to the micro- controller. At present, labor-saving and water-saving technology is a key issue in irrigation. A wireless solution for intelligent field irrigation system, based on IOT (Internet of Things) technology was proposed in this paper. Instead of conventional wired connection, the wireless design made the system easy installation and maintenance. The hardware architecture and software algorithm of wireless sensor/actuator node and portable controller, acting as the end device and coordinator in Internet network respectively, were elaborated in detail. The performance of the whole system was evaluated in the end. The long-time smooth and proper running of the system in the field proved its high reliability and practicability. As an explorative application of wireless sensor network in irrigation management, this paper offered a methodology to establish large scale remote intelligent irrigation system. People now are working actively at intelligent irrigation systems because their advantages of labor-saving and water saving. Wireless technology, known for its easy installation and maintenance, is thought preponderant to develop automatic irrigation network and becoming hot research.

TABLE OF CONTENTS

Chapter No.	TITLE	Page No.
	ABSTRACT	5
1	INTRODUCTION	1
	1.1 EMBEDDED SYSTEM	2
	1.2 DESIGN OF EMBEDDED SYSTEM	2
	1.3 CLASSIFICATION	3
2	LITERATURE SURVEY	4
	2.1 Literature Survey	4
3	AIM AND SCOPE OF PROJECT	6
	3.1 EXISTING SYSTEM	6
	3.2 PROPOSED SYSTEMS	6
	3.3 ARCHITECTURE DIAGRAM	7
	3.4 MONITORING SECTION	8
4	MATERIALS AND METHODS	9
	4.1 ARDUINO UNO AND ITS PROGRAMMING	10
	4.1.1 Overview	11
	4.1.2 Specifications	13

	4.1.3 Pin Configuration	13
	4.1.4 Communication	14
	4.2 ARDUINO SOFTWARE	16
	4.3 TEMPERATURE SENSOR	19
	4.4 OPERATION IN LMS	20
	4.5 SOIL NPK SENSOR	20
	4.6 DC MOTOR OPERATION	21
	4.7 PROJECT DESCRIPTION	23
	4.8 STANDARDS	23
	4.9 CONSTRAINTS	25
	4.10 TRADE OFF	27
5	RESULTS AND CONCLUSION	28
	5.1 INTRODUCTION	28
	5.2 NULL MODULE	28
	5.3 COMPATIBILITY ISSUES	29
6	SUMMARY AND CONCLUSION	30
	6.1 SOURCE CODE	30
	6.2 SCREENSHOT OF HARDWARE	31
	6.3 CONCLUSION	31
	6.4 FUTURE SCOPE	32
	REFERENCES	33

TABLE OF FIGURE

FIG No.	FIGURE	Page No.
1.1	Embedded Product Development Life Cycle	1
3.1	Architectural Diagram	6
3.2	Monitoring system	7
4.1	ARDUINO UNO	8
4.2	ARDUINO UNO	9
4.3	Open Arduino IDE	15
4.4	Select the COM Port	16
4.5	Select the required Arduino board	16
4.6	Sketch in Arduino IDE	17
4.7	Upload the Sketch to Arduino board	18
4.8	LM35	19
4.9	Soil sensor	21
4.10	DC Motor Operation	22
6.1	Source code	30
6.2, 6.3	Working model	30-31

CHAPTER-1

INTRODUCTION

1.1 EMBEDDED SYSTEM:

A system is something that maintains its existence and functions as a whole through the interaction of its parts. E.g. Body, Mankind, Access Control, etc A system is a part of the world that a person or group of persons during some time interval and for some purpose choose to regard as a whole, consisting of interrelated components, each component characterized by properties that are selected as being relevant to the purpose.

- Embedded System is a combination of hardware and software used to achieve a single specific task.
- Embedded systems are computer systems that monitor, respond to, or control an external environment.
- Environment connected to systems through sensors, actuators and other I/O interfaces.
- Embedded system must meet timing & other constraints imposed on it by environment.

An embedded system is a microcontroller-based, software driven, reliable, real-time control system, autonomous, or human or network interactive, operating on diverse physical variables and in diverse environments and sold into a competitive and cost-conscious market. An embedded system is not a computer system that is used primarily for processing, not a software system on PC or UNIX, not a traditional business or scientific application. High-end embedded & lower end embedded systems. High-end embedded system - Generally 32, 64 Bit Controllers used with OS. Examples Personal Digital Assistant and Mobile phones etc. Lower end embedded systems - Generally 8,16 Bit Controllers used with a minimal operating systems and hardware layout designed for the specific purpose. Examples Small controllers and devices in our everyday life like Washing Machine, Microwave ovens, where they are embedded .

1.2 DESIGN OF EMBEDDED SYSTEM:

Like every other system development design cycle embedded system too have a design cycle. The flow of the system will be like as given below. For any design cycle these will be the implementation steps. Figure 1.1 describes about the embedded product development life cycle.

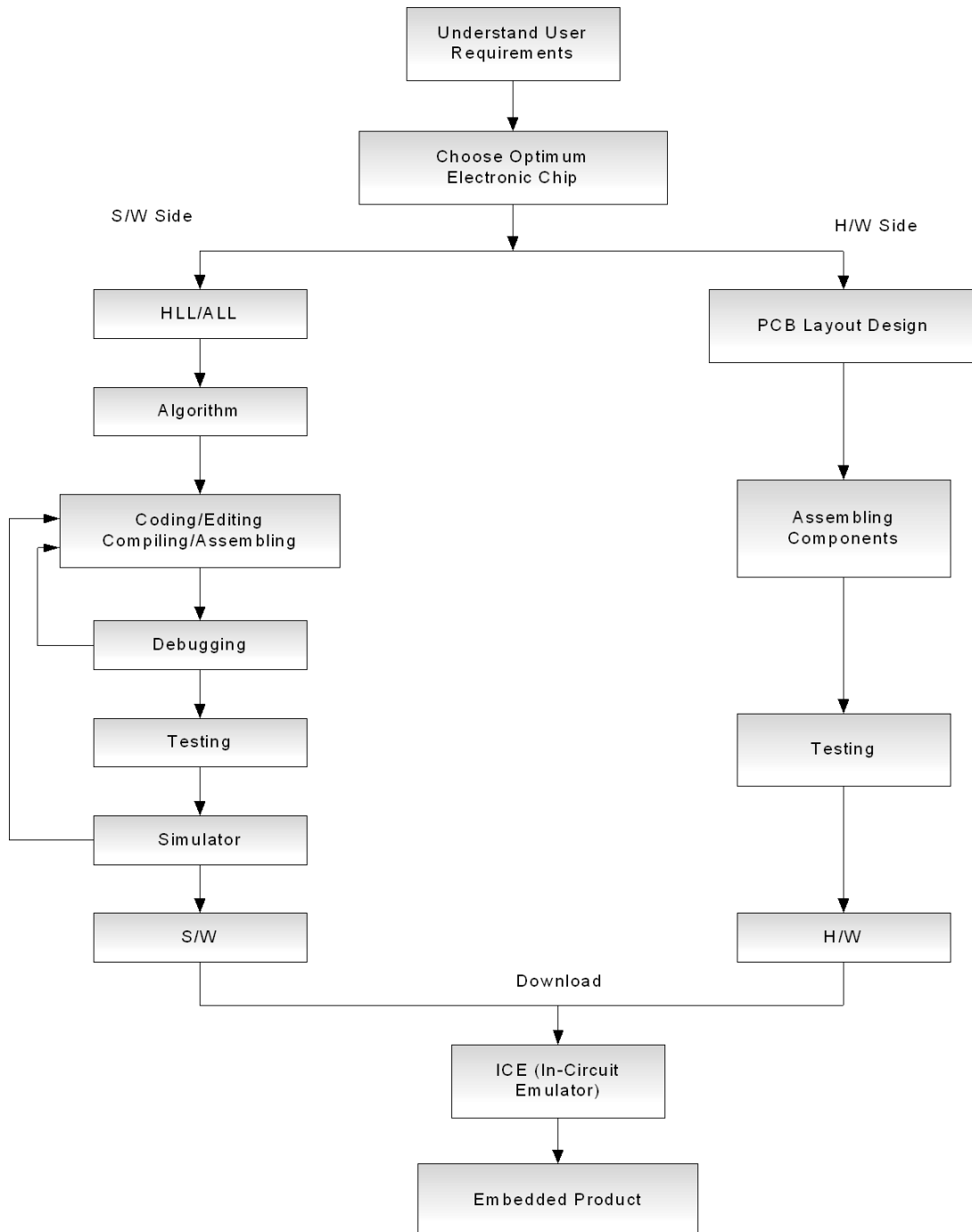


Figure 1.1 – Embedded Product Development Life Cycle

From the initial state of the project to the final fabrication the design considerations will be taken like the software consideration and the hardware components, sensor, input and output.

The electronics usually uses either a microprocessor or a microcontroller. Some large or old systems use general-purpose mainframe computers or minicomputers.

1.3 CLASSIFICATION:

- Real Time Systems.
- RTS is one which has to respond to events within a specified deadline.
- A right answer after the dead line is a wrong answer
- RTS classification
- Hard Real Time Systems
- Soft Real Time System

1.3.1 Hard Real Time Systems

1. "Hard" real-time systems have very narrow response time.
2. Example: Nuclear power system, Cardiac pacemaker.

1.3.2 Soft Real Time System

- "Soft" real-time systems have reduced constraints on "lateness" but still must operate very quickly and repeatably.
- Example: Railway reservation system – takes a few extra seconds the data remains valid.

CHAPTER-2

LITERATURE SURVEY AND OBJECTIVES

2.1 LITERATURE SURVEY

S.A. O'Shaughnessy et al. (2022) presented automated irrigation scheduling and control can be an important time management tool for a producer with multiple sprinkler irrigation systems or multiple systems relying on a single water source by prioritizing the order of irrigations field-by-field.

S. L. Davis et al. (2021) presented the field capacity and permanent wilting point were chosen for the theoretical irrigation requirement based on a soil survey. However, a controller program setting of “sand” and could mean different depths for individual controllers. Establishing differential irrigation levels using temperature–time thresholds, Wanjura, D.E., Upchurch, D.R., Mahan Data were analyzed from irrigation studies that measured the water application and cotton yield response to a range of average daily ST. All irrigation studies included a common control TT of 330 min/d above a canopy temperature threshold of 28°C as the criteria for the irrigation signal that maintained cotton in a well-watered status. In order to develop relationships between daily ST and final yield, or water input during the season, daily ST values were calculated and averaged over the irrigation season.

Mahir Dursun and Semih Ozden et al. (2020) presented the developed irrigation automation system can be proposed to be used in several commercial agricultural productions since it was obtained in low cost and in reliable operation. This application of sensor-based site-Specific irrigation has some advantages such as preventing moisture stress of trees, diminishing of excessive water usage, ensuring of rapid growing weeds and derogating scarification. If different kinds of sensors (that is, temperature, humidity, and etc.) are involved in such irrigation in future works, it can be said that an internet based remote control of irrigation automation will be possible. The developed system can also transfer fertilizer and the other agricultural chemicals (calcium, sodium, ammonium, zinc) to the field with adding new sensors and valves.

Milica,Pejanović Đurišić,Zhilbert tafa et al. (2020) This paper presents an overview of defense related applications of wireless sensor networks (WSNs). The operational context of modern military engagement has diversified into four scenarios, which set requirements and constraints upon applications of WSNs.

K.S.Vijula GraceSilja KharimP,Sivasakthi et al. (2018) In an advanced agriculture the consumption of water is very high. Due to the developments in technology, efforts are being channeled into automation of irrigation systems to facilitate remote of the irrigation to optimize crop production and cost effectiveness. practices need to intensify the rate of food crop production for the rate of population growth with available resources. This proposed work describes the automated system to make effective utilization of water resources for agriculture and crop growth monitoring using GSM.

Yandong ZhaoJunfu ZhangJinfeng Guan et al. (2020) Olympic Games in 2008 is not only a world sports event, but also a great opportunity to show our current level of economic development and modern science and technology strength. Precision water-saving irrigation automatic control system by plant physiology this paper described is one of the Olympic Games facilities projects, which takes standards of water plant physiologically need and soil water content as the basis, Through the combination of independent research and development of irrigation monitoring controller and wireless data transmission , implement a drip irrigation, sprinkler irrigation, micro-irrigation, and low-pressure pipelines, such as different modes of irrigation automatic control.

GerardRudolph Mendez,Mohd Amri,Md YunusSubhas Chandra,Mukhopadhyay (2020) The main objective of the present work is to develop a smart wireless sensor network (WSN) for an agricultural environment. Monitoring of environmental factors have increased in importance over the last decade. In particular monitoring agricultural environments for various factors such as temperature and humidity along with other factors can be of significance.

CHAPTER-3

AIM AND SCOPE OF THE PRESENT INVESTIGATION

3.1 EXISTING SYSTEM:

In existing system, we will control the water level through manual operation. It will control only the water level in the tank, so this creates our lack of concentration over the overhead tank. We will be able to miss the checking of water quality. This will create hazardous effect on our usage water.

DISADVANTAGES:

- Manual Operation
- Time Consuming

3.2 PROPOSED SYSTEM:

To overcome this, we proposed a system, to monitor our overhead tank through internet, we can monitor the water level by using ultrasonic sensor and we can be able to control the motor. We can also monitor the water purity and temperature in the yielding field by using NPK sensor and temperature sensor. So, we not only monitor but also control the motor by monitoring through internet.

ADVANTAGES:

- Maintain the temperature
- Maintain NPK level
- Maintain the water level.

3.3 ARCHITECTURE DIAGRAM

Crop Growing Section:

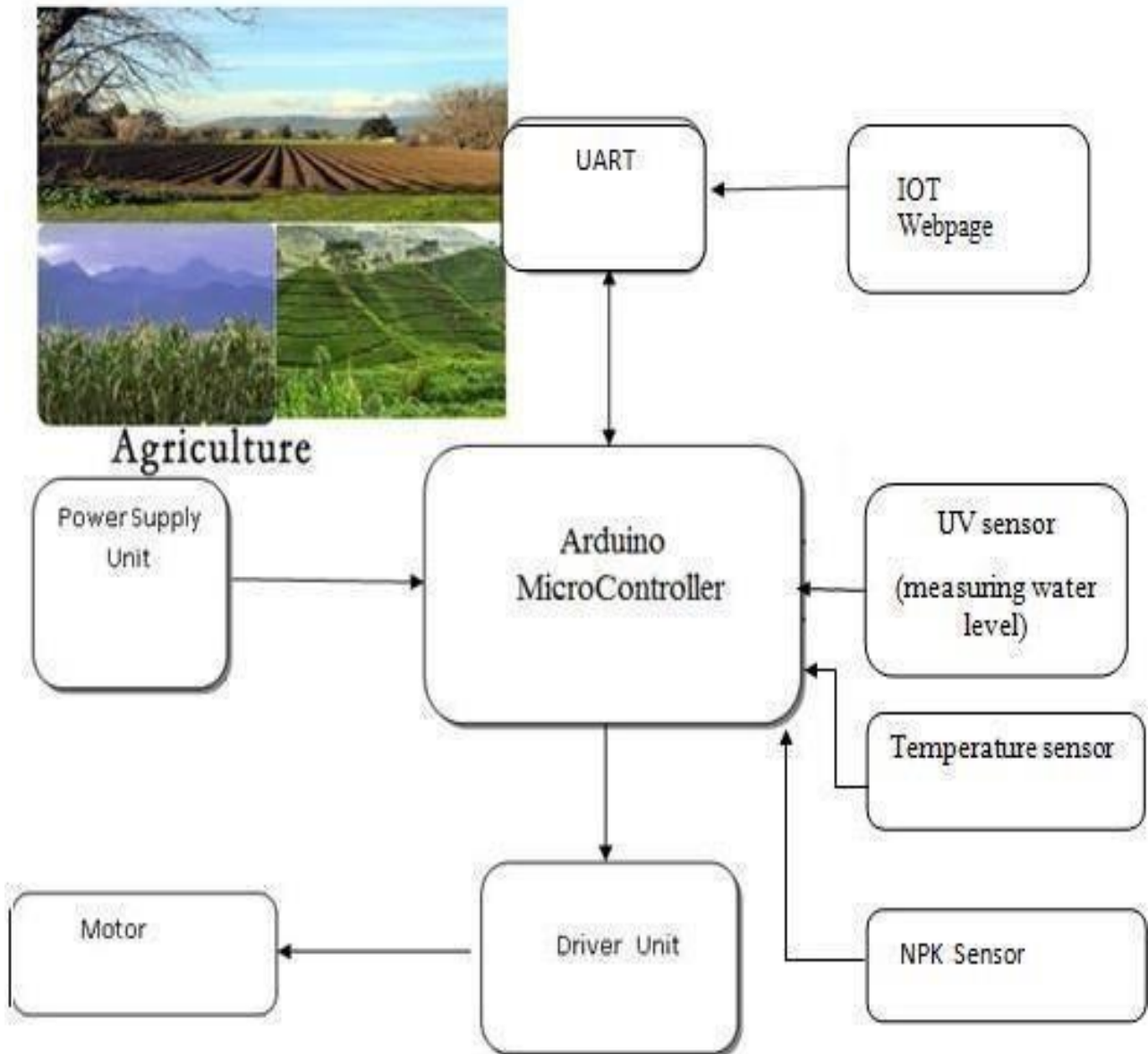


Fig. 3.1. Architectural Diagram

3.4 Monitoring section:

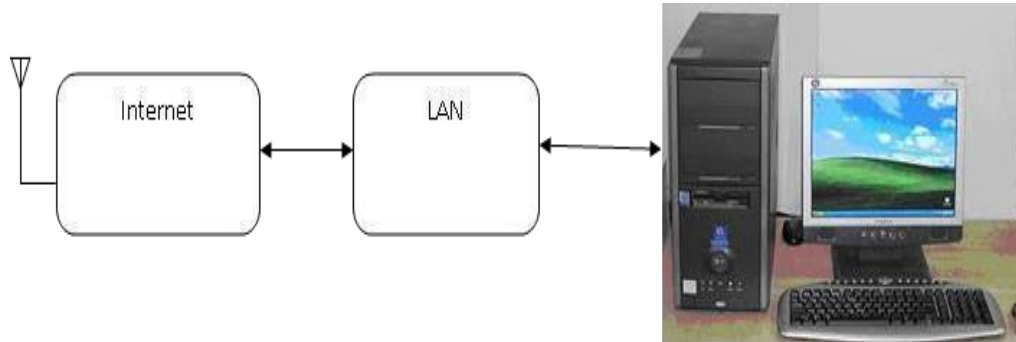


Fig. 3.2. Monitoring system

3.5 Module Description:

This smart agriculture using IOT system is powered by Arduino, it consists of Temperature sensor, Moisture sensor, water level sensor, DC motor module. When the IOT based in a agriculture monitoring system starts it checks humidity levels Sensors sense the level of water if it goes down, it automatically starts the water pump. All the datas will get updated in Webpage.

CHAPTER-4

EXPERIMENTAL, METHODS AND ALGORITHMS USED

4.1 ARDUINO UNO AND ITS PROGRAMMING

Arduino is a tool for making computers that can sense and control more of the physical world than your 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 be stand-alone, or they can be communicated with software running on your computer. 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.



4.1 ARDUINO UNO

4.1.1 Overview

The Arduino microcontroller is an easy to use yet powerful single board computer that has gained considerable traction in the hobby and professional market. The Arduino is open- source, which means hardware is reasonably priced and development software is free. This guide is for students in ME 2011, or students anywhere who are confronting the Arduino for the first time. For advanced Arduino users, prowl the web; there are lots of resources.

This guide covers the Arduino Uno board (Spark fun DEV-09950, \$29.95), a good choice for students and educators. With the Arduino board, you can write programs and create interface circuits to read switches and other sensors, and to control motors and lights with very little effort.



Fig 4.2 ARDUINO UNO

The Demilune 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 22g solid wire or header connectors.

The Arduino programming language is a simplified version of C/C++. If you know C, programming the Arduino will be familiar. If you do not know C, no need to worry as only a few commands are needed to perform useful functions.

An important feature of the Arduino is that you can create a control program on the host PC, download it to the Arduino and it will run automatically. Remove the USB cable connection to the PC, and the program will still run from the top each time you push the reset button.

Remove the battery and put the Arduino board in a closet for six months. When you reconnect the battery, the last program you stored will run. This means that you connect the board to the host PC to develop and debug your program, but once that is done, you no longer need the PC to run the program.

the battery and put the Arduino board in a closet for six months. When you reconnect the battery, the last program you stored will run. This means that you connect the board to the host PC to develop and debug your program, but once that is done, you no longer need the PC to run the program

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 a 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.

.

4.1.2 SPECIFICATIONS:

- Microcontroller ATmega328
- Operating Voltage 5V
- Input Voltage (recommended)
- 7-12V Input Voltage (limits) 6-20V
- Digital I/O Pins 14 (of which 6 provide PWM output) Analog Input Pins 6
- DC Current per I/O Pin 40 mA DC Current for 3.3V
- Pin 50 mA
- Flash Memory 32 KB (ATmega328) of which 0.5 KB used by bootloader SRAM 2 KB (ATmega328)
- EEPROM 1 KB (ATmega328)
- Clock Speed 16 MHz

4.1.3 Pin Configuration

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall- wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the GND and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** this pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board.
- **3V3.** A 3.3-volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.
- **Memory:** The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM.
- **Input and Output:** Each of the 14 digital pins on the Uno can be used as an input or output, using pin Mode (), digital Write (), and digital Read() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 k ohms. In addition, some pins have specialized functions:
- **Serial:** 0 (RX) and 1 (TX). Used to receive (RX) and Transmit (TX) TTL serial data.
- These pins are connected to the **corresponding** pins of the ATmega8U2 USB-to- TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attach Interrupt () function for details.

- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the `analogWrite()` function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication using the **SPI library**.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default, they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the `analogReference()` function. Additionally, some pins have specialized functionality:
- **AREF.** Reference voltage for the analog inputs. Used with `analogReference()`.
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

4.1.4 Communication

Microcontrollers depend on a host computer for developing and compiling programs. The software used on the host computer is known as an integrated development environment, or IDE. For the Arduino, the development environment is based on the open source Processing platform (www.processing.org) which is described by its creators as a “programming language and environment for people who want to program images, animation, and interactions. “The Arduino programming language leverages an open source project known as Wiring (wiring.org.co). The Arduino language is based on good old- fashioned C. If you are unfamiliar with this language, don't worry; it's not hard to learn, and the Arduino IDE provides some feedback when you make mistakes in your programs. The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX).

An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A Software Serial library allows for serial communication on any of the Uno's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.

Repositories that extend core programming functionality. Libraries let you re-use code without having to physically copy and paste it into all your programs. The standard Arduino software installation comes with several libraries you may use, and you can download others from the Arduino support pages and from third-party websites that publish Arduino library code. A good example of a library you'll use with the Robot and likely many other robot projects is Servo. This library allows you to connect one or more hobby R/C servos to the Arduino's digital I/O pins. The Servo library comes with the standard Arduino installation package Library-

```
>Servo. This adds the  
line#include  
<Servo.h>
```

Which tells the Arduino IDE that you wish to include the Servo library in your sketch. With the functionality of the library now available to you, you can use its various functions to control one or more servos. For example, you can use the write function to rotate a servo to a specific position, from 0 to 180 degrees. The following code myServo.write(90).

Moves a servo to its midpoint, or 90-degree position. Structurally, Arduino sketches are very straightforward and are pretty easy to read and understand. The Arduino program contains two main parts: `setup ()` and `loop ()`. These are programming functions that do what their names suggest: `setup ()` sets up the Arduino hardware, such as specifying which I/O lines you plan to use, and whether they are inputs or outputs. The `loop ()` function is repeated endlessly when the Arduino is operating. Arduino IDE (Integrated development environment) is used to write the program and dump into the Arduino board.

4.2 ARDUINO SOFTWARE:

- Open Arduino IDE as shown below

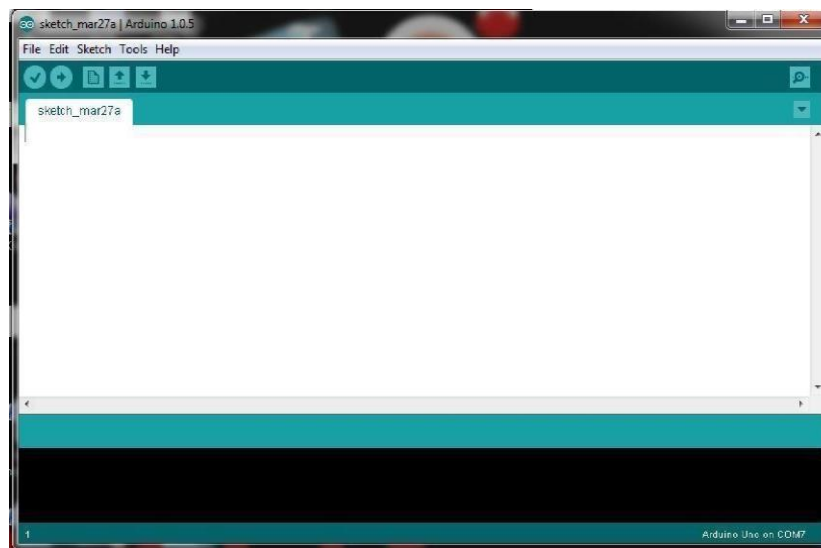


Fig 4.3 Open Arduino IDE

- Select the COM Port from tool

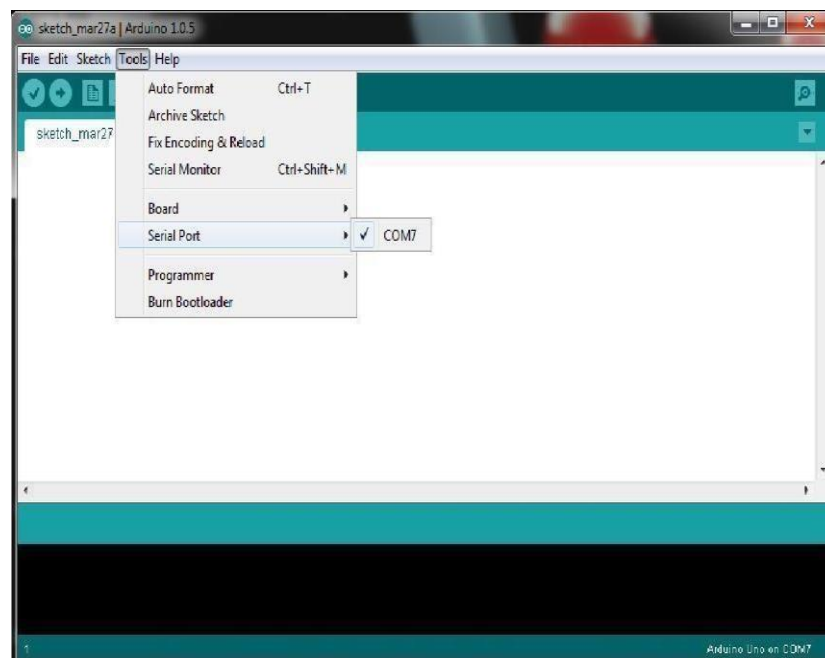


Fig 4.4 Select the COM Port

Select the required Arduino board

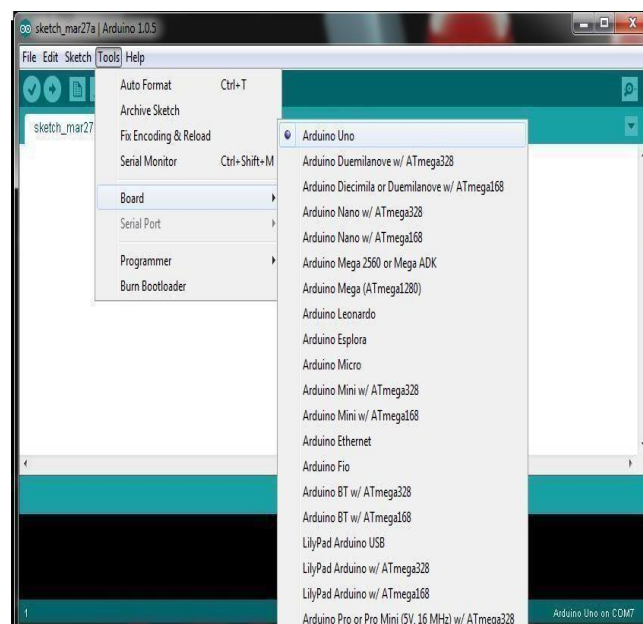


Fig 4.5. Select the required Arduino board

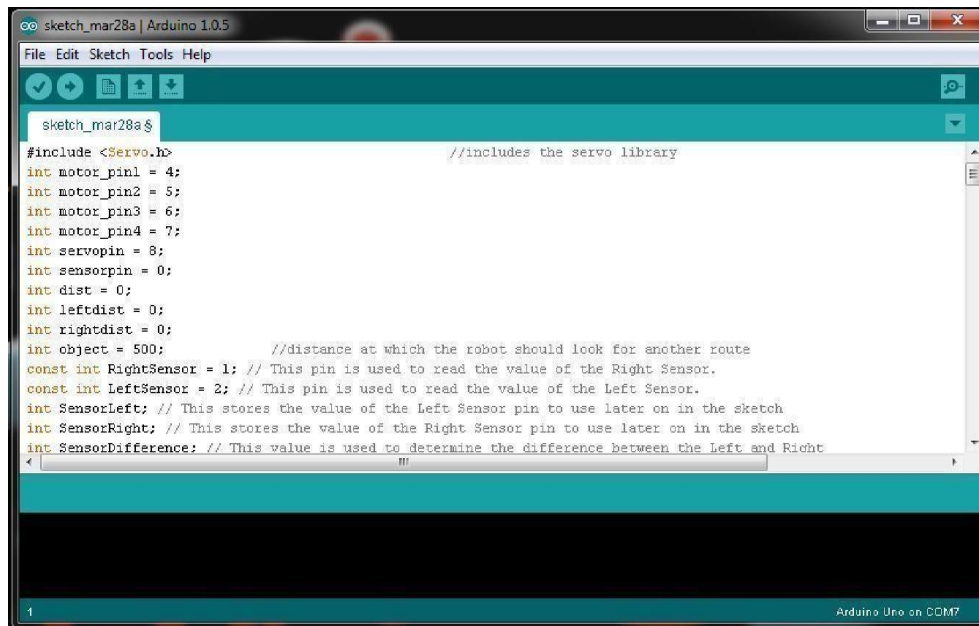


Fig 4.6: Sketch in Arduino IDE

- Compile and upload the Sketch to Arduino board upload the Sketch to Arduino board

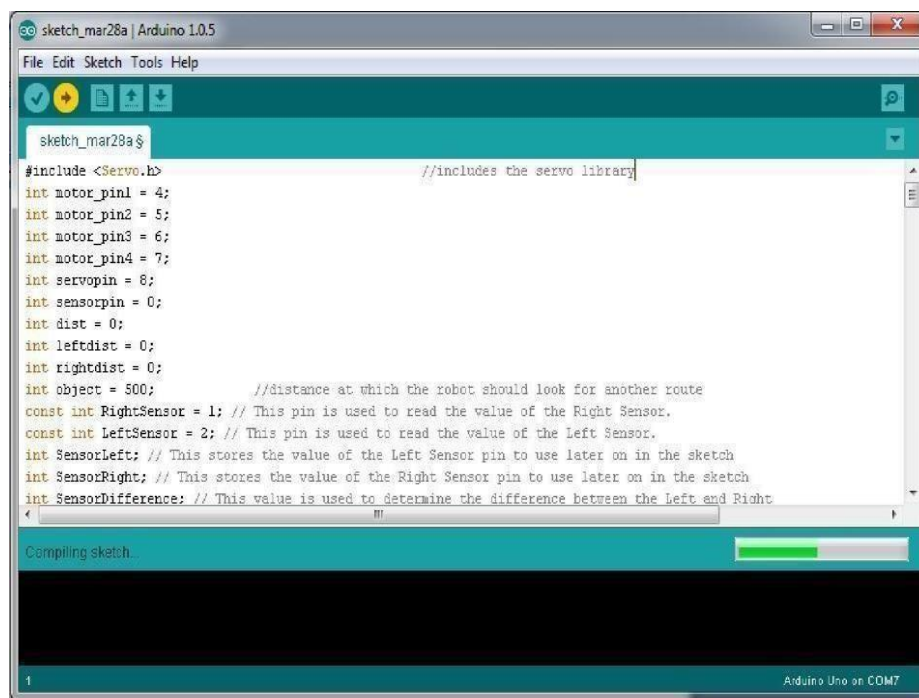


Fig 4.7: upload the Sketch to Arduino board

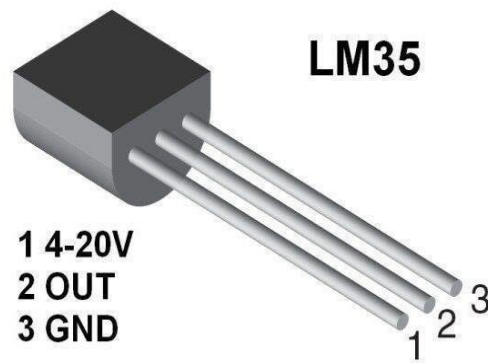


Fig. 4.8 LM35

4.3 TEMP 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 $^{\circ}$ Celsius (Centigrade)
- Rated for full -55° to $+150^{\circ}\text{C}$ rang
- $\pm 1/4^{\circ}\text{C}$ of typical nonlinearity

4.4 Operation of LM35:

The LM35 can be connected easily in the same way as other integrated circuit temperature sensors. It can be stuck or established to a 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 $\pm 0.5^{\circ}\text{C}$ accuracy. These sensors completely aligned with digital temperature readings in degree Celsius.

4.5 SOIL NPK Sensor:

The soil NPK sensor is suitable for detecting the content of nitrogen, phosphorus, and potassium in the soil. It helps in determining the fertility of the soil thereby facilitating the systematic assessment of the soil condition. The sensor can be buried in the soil for a long time. It has a High-quality probe, rust resistance, electrolytic resistance, salt & alkali corrosion resistance, to ensure the long-term operation of the probe part. Therefore, it is suitable for all kinds of soil. It is suitable for the detection of alkaline soil, acid soil, seedling bed soil & coconut bran soil.



Fig. 4.9 Soil sensor

The sensor operates on 9-24V & power consumption is very low. While talking about the accuracy of the sensor, it is up to within 2%. The nitrogen, phosphorous & potassium measuring resolution is up to 1mg/kg (mg/l).

4.6 DC Motor Operation

In a dc motor, the stator poles are supplied by dc excitation current, which produces a dc magnetic field. The rotor is supplied by dc current through the brushes, commutator and coils. The interaction of the magnetic field and rotor current generates a force that drives the motor. The magnetic field lines enter into the rotor from the north pole (N) and exit toward the south pole (S). The poles generate a magnetic field that is perpendicular to the current carrying conductors. The interaction between the field and the current produces a Lorentz force; the force is perpendicular to both the magnetic field and conductor. The generated force turns the rotor until the coil reaches the neutral point between the poles. At this point, the magnetic field becomes practically zero together with the force. However, inertia drives the motor beyond the neutral zone where the direction of the magnetic field reverses. To avoid the reversal of the force direction, the commutator changes the current direction, which maintains the counterclockwise rotation.

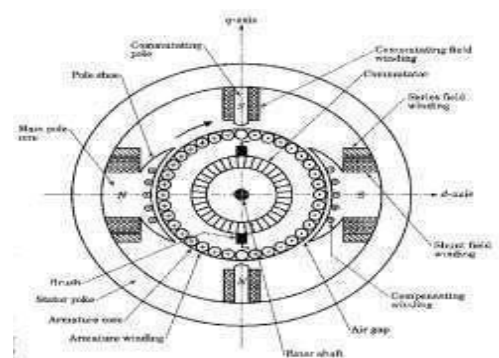
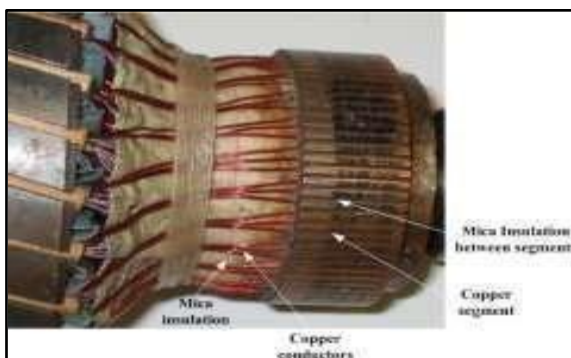


Fig. 4.10 DC Motor Operation



Fig. 4.10 Soil sensor

Specifications

- Power: 9V-24V
- Measuring Range: 0-1999 mg/kg (mg/l)
- Operating Temperature: 5-45 °C
- Resolution: 1mg/kg
- Precision: $\pm 2\%$ F.S.
- Output Signal: RS485
- Baud Rate: 2400/4800/9600

4.7 PROJECT DESCRIPTION

This project description provides a comprehensive outline for designing and implementing a distributed system for monitoring an air-ground wireless sensor network for crop monitoring. It covers various aspects including system architecture, sensor node design, data processing, user interface, implementation, and testing.

4.8 STANDARDS

When it comes to designing and implementing a distributed system for monitoring an air-ground wireless sensor network for crop monitoring, adhering to relevant standards is crucial for ensuring interoperability, reliability, and compatibility. Here are some key standards to consider:

- **Wireless Sensor Networks Standard:**

IEEE 802.15.4: This standard defines the physical and MAC layers for low-rate wireless personal area networks (LR-WPANS), which are commonly used in wireless sensor networks. IEEE 802.15.4e: This amendment to the IEEE 802.15.4 standard introduces enhancements for industrial applications, including determinism, reliability, and energy efficiency. IEEE 802.15.4g: This standard specifies the physical layer for wireless smart utility networks, which can be relevant for agricultural applications involving long-range communication.

- **Internet of Things (IoT) Standards:**

OneM2M: This is a global standard for IoT interoperability, providing a common platform for managing IoT devices and data. It ensures compatibility and seamless integration across different IoT systems and applications.

MQTT (Message Queuing Telemetry Transport): MQTT is a lightweight messaging protocol ideal for IoT applications, enabling efficient communication between sensor nodes and data processing units.

CoAP (Constrained Application Protocol): CoAP is a specialized web transfer protocol designed for constrained devices and networks, making it suitable for IoT applications in resource- constrained environments like wireless sensor networks.

Gateway: Gateway develops and promotes standards for agricultural data exchange, including data formats for precision agriculture, crop monitoring, and farm management systems.

ISO 11783 (ISOBUS): This international standard defines a communication protocol for agricultural machinery, facilitating interoperability between different equipment and systems used in farming operations.

OGC SensorThings API: The Open Geospatial Consortium (OGC) SensorThings API standardizes the way IoT sensor data is accessed and queried, providing a unified interface for managing and exchanging sensor observations.

- Security and Privacy Standards:

ISO/IEC 27001: This standard specifies requirements for information security management systems (ISMS), ensuring the confidentiality, integrity, and availability of data in the distributed system.

NIST Cybersecurity Framework: Developed by the National Institute of Standards and Technology (NIST), this framework provides guidance on managing cybersecurity risks, protecting sensitive data, and ensuring the resilience of the distributed system against cyber threats.

GDPR (General Data Protection Regulation): Compliance with GDPR is essential for protecting the privacy rights of individuals whose personal data may be collected and processed within the distributed system.

Adhering to these standards helps in ensuring the reliability, interoperability, security, and privacy of the distributed system for monitoring an air-ground wireless sensor network for crop monitoring. It also facilitates integration with existing agricultural systems and promotes compatibility with future advancements in technology and industry practices.

4.9 CONSTRAINTS

When designing and implementing a distributed system for monitoring an air-ground wireless sensor network for crop monitoring, there are various constraints and challenges that need to be addressed. Here are some common constraints:

- **Resource Constraints:**

Energy Constraints: Sensor nodes often rely on limited battery power, requiring energy-efficient communication protocols, data aggregation techniques, and power management strategies to prolong battery life.

Processing Power: Sensor nodes may have limited processing capabilities, necessitating lightweight algorithms and efficient data processing techniques.

Memory Constraints: Limited memory on sensor nodes may restrict the amount of data that can be stored locally, requiring optimized data storage and management approaches.

- **Communication Constraints:**

Bandwidth Limitations: Wireless communication in air-ground sensor networks may be subject to bandwidth constraints, especially in rural or remote areas. This requires careful selection of communication protocols and efficient data transmission strategies.

Interference and Signal Attenuation: Environmental factors such as foliage, terrain, and weather conditions can affect wireless signal propagation, leading to interference and signal attenuation. Mitigation techniques such as antenna diversity and routing protocols need to be employed.

- **Environmental Constraints:**

Harsh Conditions: Agricultural environments can be harsh, with exposure to moisture, dust, extreme temperatures, and physical damage. Sensor nodes and infrastructure components must be ruggedized to withstand these conditions.

Variable Terrain: Agricultural landscapes may feature uneven terrain, which can affect the placement and deployment of sensor nodes. Adaptation mechanisms and robust network topology designs are required to accommodate variable terrain.

- Scalability Constraints:

Network Scalability: The distributed system should be scalable to accommodate a growing number of sensor nodes and an expanding monitoring area. Scalable communication protocols, data management techniques, and network architectures are essential.

Data Scalability: As the volume of monitored data increases, scalable data storage and processing solutions are needed to handle the growing data load efficiently.

- Security and Privacy Constraints:

Data Security: Agricultural data, including crop monitoring data, is valuable and may be vulnerable to security threats such as unauthorized access, data breaches, and cyber-attacks. Robust encryption, authentication, and access control mechanisms are necessary to ensure data security.

Privacy Concerns: Privacy concerns arise regarding the collection and use of sensitive data, particularly if it involves personal information or location data. Compliance with data protection regulations and privacy-preserving techniques must be considered.

4.10 TRADE OFFS

Designing and implementing a distributed system for monitoring an air-ground wireless sensor network for crop monitoring involves various trade-offs that need to be considered. Here are some key aspects to consider:

Scalability vs. Complexity: One trade-off is between scalability and complexity. A highly scalable system may require a more complex architecture, which can increase development time and maintenance overhead. Conversely, a simpler architecture may limit scalability.

Reliability vs. Cost: Ensuring high reliability often requires redundant components and fault-tolerant mechanisms, which can increase costs. Balancing reliability requirements with budget constraints is crucial.

Real-time vs. Batch Processing: Real-time data processing provides immediate insights but may require more computational resources. Batch processing can be more resource-efficient but may introduce delays in data analysis.

Centralized vs. Decentralized Architecture: A centralized architecture simplifies management but can become a single point of failure. Decentralized architectures distribute processing and storage but require more complex coordination mechanisms.

Data Consistency vs. Availability: Ensuring data consistency across distributed nodes may introduce latency due to synchronization overhead. Balancing data consistency with availability requirements is essential, especially in dynamic environments.

Security vs. Performance: Implementing robust security measures such as encryption and authentication adds overhead, which can impact system performance. Finding the right balance between security and performance is critical.

Energy Efficiency vs. Data Accuracy: Optimizing sensor node energy consumption is crucial for prolonging network lifetime, but it may require sacrificing data accuracy or frequency of data transmission.

CHAPTER 5

RESULTS, DISCUSSION AND PERFORMANCE ANALYSIS

SERIAL COMMUNICATION

5.1 INTRODUCTION:

Serial communication is basically the transmission or reception of data one bit at a time. Today's computers generally address data in bytes or some multiple thereof. A byte contains 8 bits. A bit is basically either a logical 1 or zero. Every character on this page is actually expressed internally as one byte. The serial port is used to convert each byte to a stream of ones and zeroes as well as to convert a stream of ones and zeroes to bytes. The serial port contains an electronic chip called a Universal Asynchronous Receiver/Transmitter (UART) that actually does the conversion. The serial port has many pins. We will discuss the transmit and receive pin first. Electrically speaking, whenever the serial port sends a logical one (1) a negative voltage is effected on the transmit pin. Whenever the serial port sends a logical zero (0) a positive voltage is affected. When no data is being sent, the serial port's transmit pin's voltage is negative (1) and is said to be in a MARK state. Note that the serial port can also be forced to keep the transmit pin at a positive voltage (0) and is said to be the SPACE or BREAK state. (The terms MARK and SPACE are also used to simply denote a negative voltage (1) or a positive voltage (0) at the transmit pin respectively). When transmitting a byte, the UART (serial port) first sends a START BIT which is a positive voltage (0), followed by the data (general 8 bits, but could be 5, 6, 7, or 8 bits) followed by one or two STOP Bits which is a negative (1) voltage. The sequence is repeated for each byte sent. Figure 1 shows a diagram of what a byte transmission would look like.

5.2 NULL MODEM:

Serial communications with RS232. One of the oldest and most widely spread communication methods in computer world. The way this type of communication can be performed is pretty well defined in standards. I.e. with one exception. The standards.

show the use of **DTE/DCE** communication, the way a computer should communicate with a peripheral device like a modem. For your information, **DTE** means *Data Terminal Equipment* (computers etc.) where **DCE** is the abbreviation of *Data*

When we look at the connector pin out of the RS232 port, we see two pins which are certainly used for flow control. These two pins are **RTS**, request to send and **CTS**, clear to send. With **DTE/DCE** communication (i.e. a computer communicating with a modem device) **RTS** is an output on the **DTE** and input on the **DCE**. **CTS** are the answering signal coming from the **DCE**. Before sending a character, the **DTE** asks permission by setting its **RTS** output. No information will be sent until the **DCE** grants permission by using the **CTS** line.

5.3 COMPATIBILITY ISSUES:

If you read about null modems, this three-wire null modem cable is often talked about. Yes, it is simple but can we use it in all circumstances? There is a problem, if either of the two devices checks the **DSR** or **CD** inputs. These signals normally define the ability of the other side to communicate. As they are not connected, their signal level will never go high. This might cause a problem.

Hardware Components:

- Arduino Microcontroller
- IOT
- Control unit
- NPK Sensor
- Temperature sensors
- UV Sensor

Software Components:

- Embedded C
- HTML

Application:

- Agriculture applications
- Remote control.

CHAPTER 6

SUMMARY AND CONCLUSION

6.1 SOURCE CODE:

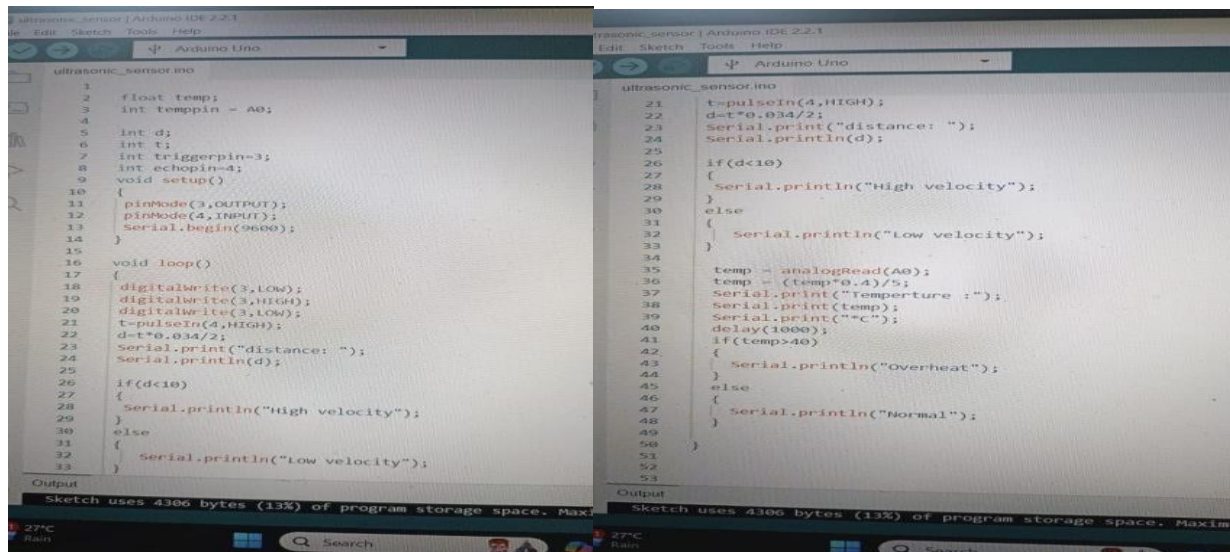


Fig. 6.1. SOURCE CODE

6.2 SCREENSHOT OF HARDWARE COMPONENTS

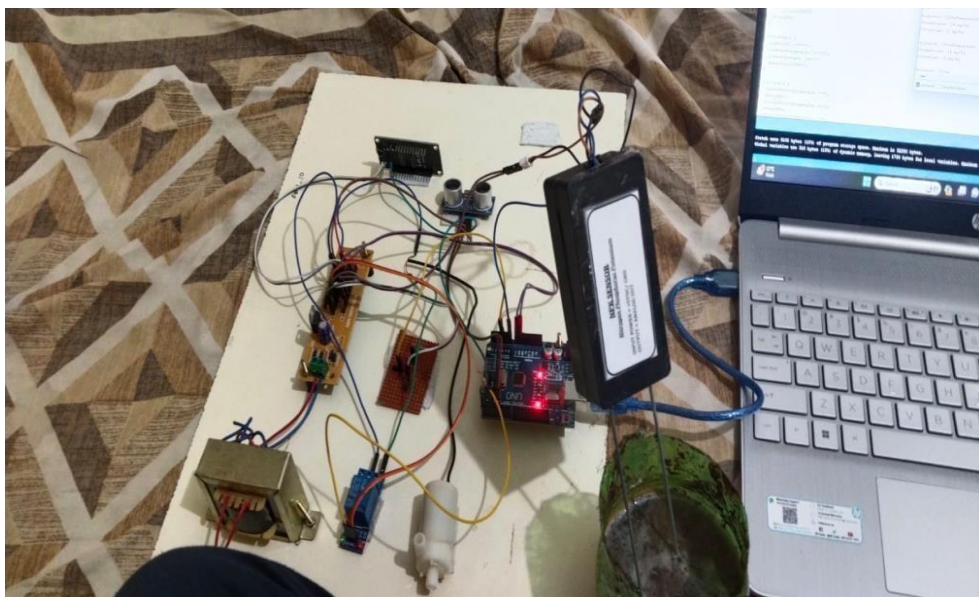


Fig. 6.2. Working model

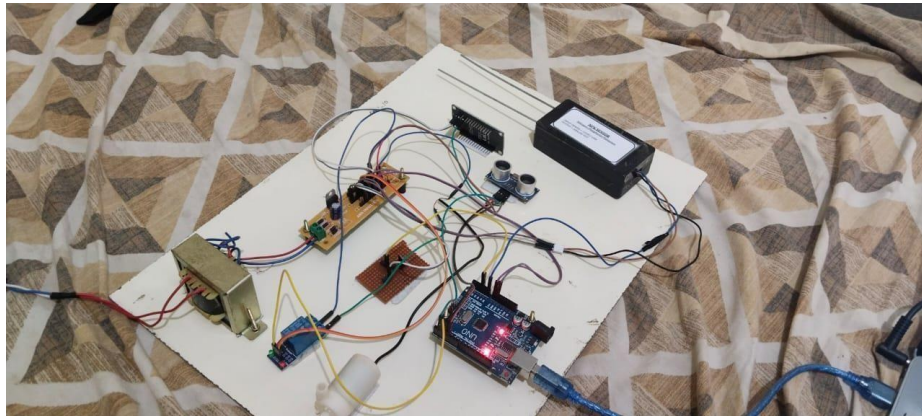


Fig. 6.3. Working model

6.3 CONCLUSION

Agriculture is always a crucial part of everyday life. Due to global environmental situation, water management and conservation is vital for human survival. Today, the farmers are suffering from the lack of rains and scarcity of water. The main objective of this paper is to provide an automatic irrigation system. In recent times, there were huge needs of consumer-based humanitarian projects that could be rapidly developed using Internet of Things (IoT) technology.

In this paper, an innovative and a general solution based on a combination of UAV and WSN to monitor variables from fragmented crop fields has been proposed. Problems related to fragmentation of the regions of interest in a wide area such as the impossibility of deploying a fully connected WSN, the large distances to cover and the continuous monitoring of the environment variables can be overcome with the proposed approach.

Although the experimental results have been obtained using a three-parcel scenario, the system is completely scalable. Experimental results have shown that this approach provides many advantages considering ground WSN, such as improvement of the WSN communication workflow, improvement of the data routing to the base station, and a better monitoring of the crop.

In conclusion, we believe that the proposed approach has a great potential for remote crop monitoring using wireless sensor networks technology. Moreover, it also overcomes many of its limitations in the kind of scenario at hand.

6.4 FUTURE SCOPE:

Improved yield prediction and planning in agriculture is achieved through the use of advanced technologies and data analytics. By leveraging historical data, weather patterns, and crop monitoring information, farmers can make more accurate predictions about potential crop yields for the upcoming season

REFERENCES

- [1] Ayaz M, Ammad-Uddin M, Sharif Z, Mansour A, Aggoune EH. Internet-of-Things (IoT)-based smart agriculture: Toward making the field stalk. *IEEE Access*. 2019 Aug 1;7:129551-83.
- [2] Shafi U, Mumtaz R, Iqbal N, Zaidi SM, Zaidi SA, Hussain I, Mahmood Z. A multi-modal approach for crop health mapping using low altitude remote sensing, internet of things (IoT) and machine learning. *IEEE Access*. 2020 Jun 16;8:112708-24.
- [3] Herwitz, S.; Dunagan, S.; Sullivan, D.; Higgins, R.; Johnson, L.; Zheng, J.; Slye, R.; Brass, J.; Leung, J.; Gallmeyer, B.; Aoyagi, M. Solar-Powered UAV Mission for Agricultural Decision Support. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, Toulouse, France, 21–25 July 2003; Volume 3, pp. 1692–1694.
- [4] Mowla MN, Mowla N, Shah AS, Rabie K, Shongwe T. Internet of Things and Wireless Sensor Networks for Smart Agriculture Applications-A Survey. *IEEE Access*. 2023 Dec 22.
- [5] Mehedi IM, Hanif MS, Bilal M, Vellingiri MT, Palaniswamy T. Remote Sensing and Decision Support System Applications in Precision Agriculture: Challenges and Possibilities. *IEEE Access*. 2024 Mar 25.
- [6] Nurlan Z, Zhukabayeva T, Othman M, Adamova A, Zhakiyev N. Wireless sensor network as a mesh: Vision and challenges. *IEEE Access*. 2021 Dec 21;10:46-67.
- [7] Kombo OH, Kumaran S, Bovim A. Design and application of a low-cost, low-power, LoRa-GSM, IoT enabled system for monitoring of groundwater resources with energy harvesting integration. *IEEE Access*. 2021 Sep 13;9:128417-33.
- [8] Hossain FF, Messenger R, Captain GL, Ekin S, Jacob JD, Taghvaeian S, O'Hara JF. Soil moisture monitoring through UAS-assisted internet of things LoRaWAN wireless underground sensors. *IEEE Access*. 2022 Sep 27;10:102107-18.
- [9] Hsiao SJ, Sung WT. Building a fish-vegetable coexistence system based on a wireless sensor network. *IEEE Access*. 2020 Oct 21;8:192119-31.
- [10] Varghese R, Sharma S. Affordable smart farming using IoT and machine learning. In *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)* 2018 Jun 14 (pp. 645-650). IEEE.