A Real-Time (or) Field-based Research Project Report

on

**An Intelligent system an environment quality remote monitoring system using IOT**

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

**Bachelor of Technology**

in

**COMPUTER SCIENCE AND ENGINEERING**

by

O.SATHWIKA [227R1A05A5]

J.NITHIN [227R1A0590]

Y.SANDEEP [227R1A05C8]

Under the guidance of

**MR B P Deepak Kumar**

**Assistant professor**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

# **CMR TECHNICAL CAMPUS**

**UGC AUTONOMOUS**

**Accredited by NBA & NAAC with ‘A’ Grade**

**Approved by AICTE, New Delhi and JNTUH Hyderabad Kandlakoya (V), Medchal Road, Hyderabad - 501401**

**December, 2023**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**CERTIFICATE**

This is to certify that the Real-Time (or) Field-based Research Project Report entitled **“ An Intelligent system an environment quality remote monitoring system using IOT”** being submitted by **O.Sathwika(227r1a05a5), J.Nithin(227r1a0590), Y.Sandeep (227r1a05c8)** in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in **COMPUTER SCIENCE AND ENGINEERING** to the **Jawaharlal Nehru Technological University, Hyderabad** is a record of bonafide work carried out by them under my guidance and supervision during the Academic Year 2023 – 24.The results embodied in this thesis have not been submitted to any other University or Institute for the award of any other degree or diploma.

**< Signature of the Supervisor> <Signatureof the HOD>**

**MR B P DEEPAK KUMAR Dr.K.Srujan Raju**

**Assistant professor Head of the Department**

**< Signature of the Director>**

**Dr. A. RajiReddy**

**Director**

**Abstract**

Air quality, water pollution, and radiation pollution are major factors that pose genuine challenges in the environment. Suitable monitoring is necessary so that the world can achieve sustainable growth, by maintaining a healthy society. In recent the environment monitoring has turned into a smart environment monitoring (SEM) system, with the advances in the internet of things (IoT) and the development of modern sensors. Under this scenario, the present manuscript aims to accomplish a critical review of noteworthy contributions and research studies on SEM, that involve monitoring of air quality, water quality, radiation years, pollution, and agriculture systems. The review is divided on the basis of the purposes where SEM methods are applied, and then each purpose is further analyzed in terms of the sensors used, machine learning techniques involved, and classification methods used. The detailed analysis follows the extensive review which has suggested major recommendations and impacts of SEM research on the basis of discussion results and research trends analyzed. The authors have critically studied how the advances in sensor technology, IoT and machine learning methods make environment monitoring a truly smart monitoring system. Finally, the framework of robust methods of machine learning; denoising methods and development of suitable standards for wireless sensor networks (WSNs), has been suggested. In modern times, the society is heavily influenced by environmental pollution. For this reason, the scientist and Engineer are desperately focused on their environment pollution measurement system invention. This paper presents an intelligent system which has highly efficient, low cost, low power consumption, air, sound and water quality real-time monitoring

Table of Contents

* 1. Introduction…………………………………………. 5
  2. Literature Survey…………………………................. 7
  3. Analysis and Design………………………………… 13
  4. Implementation………………………………………23
  5. Testing and Debugging/Results……………………...34
  6. Conclusion…………………………………………..34
  7. Reference / Bibliography……………………………35

Introduction

Many embedded systems have substantially different design constraints than desktop computing applications. No single characterization applies to the diverse spectrum of embedded systems. However, some combination of cost pressure, long life-cycle, real¬-time requirements, reliability requirements, and design culture dysfunction can make it difficult to be successful applying traditional computer design methodologies and tools to embedded applications. Embedded systems in many cases must be optimized for life-cycle and business-driven factors rather than for maximum computing throughput. There is currently little tool support for expanding embedded computer design to the scope of holistic embedded system design. However, knowing the strengths and weaknesses of current approaches can set expectations appropriately, identify risk areas to tool adopters, and suggest ways in which tool builders can meet industrial needs. If we look around us, today we see numerous appliances which we use daily, be it our refrigerator, the microwave oven, cars, PDAs etc. Most appliances today are powered by something beneath the sheath that makes them do what they do. These are tiny microprocessors, which respond to various keystrokes or inputs. These tiny microprocessors, working on basic assembly languages, are the heart of the appliances. We call them embedded systems. Of all the semiconductor industries, the embedded systems market place is the most conservative, and engineering decisions here usually lean towards established, low risk solutions. Welcome to the world of embedded systems, of computers that will not look like computers and won’t function like anything we are familiar with.

2.1 CLASSIFICATION

Embedded systems are divided into autonomous, realtime, networked & mobile categories.

Autonomous systems

They function in standalone mode. Many embedded systems used for process control in manufacturing units& automobiles fall under this category.

Real-time embedded systems

These are required to carry out specific tasks in a specified amount of time. These systems are extensively used to carry out time critical tasks in process control.

Networked embedded systems

They monitor plant parameters such as temperature, pressure and humidity and send the data over the network to a centralized system for on line monitoring.

Mobile gadgets

Fig 2.1: Embedded Development Life Cycle

• Aerospace and defence electronics: Fire control, radar, robotics/sensors, sonar.

• Automotive: Autobody electronics, auto power train, auto safety, car information systems.

• Broadcast & entertainment: Analog and digital sound products, camaras, DVDs, Set top boxes, virtual reality systems, graphic products.

• Consumer/internet appliances: Business handheld computers, business network computers/terminals, electronic books, internet smart handheld devices, PDAs

**Literature Survey**

ARUDINO:

The Arduino is a family of microcontroller boards to simplify electronic design, prototyping and experimenting for artists, hackers, hobbyists, but also many professionals. People use it as brains for their robots, to build new digital music instruments, or to build a system that lets your house plants tweet you when they’re dry. Arduinos (we use the standard Arduino Uno) are built around an ATmega microcontroller — essentially a complete computer with CPU, RAM, Flash memory, and input/output pins, all on a single chip. Unlike, say, a Raspberry Pi, it’s designed to attach all kinds of sensors, LEDs, small motors and speakers, servos, etc. directly to these pins, which can read in or output digital or analog voltages between 0 and 5 volts.

DIGITAL PINS

In addition to the specific functions listed below, the digital pins on an Arduino board can be used for general purpose input and output via the pin Mode(), Digital Read(), and Digital Write() commands. Each pin has an internal pull-up resistor which can be turned on and off using digital Write()

ANALOG PINS

In addition to the specific functions listed below, the analog input pins support 10-bit analog-to-digital conversion (ADC) using the analog Read() function. Most of the analog inputs can also be used as digital pins: analog input 0 as digital pin 14 through analog input 5 as digital pin 19. Analog inputs 6 and 7 (present on the Mini and BT) cannot be used as digital pins.

 I2C: 4 (SDA) and 5 (SCL). Support I2C (TWI) communication using the Wire library (documentation on the Wiring website).

POWER PINS

 VIN (sometimes labeled "9V"): 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. Also note that the Lily Pad has no VIN pin and accepts only a regulated input.

OTHER PINS

 AREF: Reference voltage for the analog inputs. Used with analog Reference().

 Reset: (Diecimila-only) Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

ATMEGA328

Pin diagram

Figure 2.4 Pin Configuration of Atmega328

Pin Description

VCC:

Digital supply voltage.

GND:

Ground.

Figure 2.5 Block Diagram

OVERVIEW

This section discusses the AVR core architecture in general. The main function of the CPU core is to ensure correct program execution. The CPU must therefore be able to access

ALU – ARITHMETIC LOGIC UNIT

The high-performance AVR ALU operates in direct connection with all the 32 general purpose working registers. Within a single clock cycle, arithmetic operations between general purpose registers or between a register and an immediate are executed. The ALU operations are divided into three main categories – arithmetic, logical, and bit functions. Some implementations of the architecture also provide a powerful multiplier supporting both

signed/unsigned multiplication and fractional format. See the “Instruction Set” section for a detailed description.

STATUS REGISTER

The Status Register contains information about the result of the most recently executed arithmetic instruction. This information can be used for altering program flow in order to perform conditional operations. Note that the Status Register is updated after all ALU operations, as specified in the Instruction Set Reference.

STACK POINTER

The Stack is mainly used for storing temporary data, for storing local variables and for storing return addresses after interrupts and subroutine calls. Note that the Stack is implemented as growing from higher to lower memorylocations. The Stack Pointer Register always points to the top of the Stack. The Stack Pointer points to the data SRAM Stack area where the Subroutine and Interrupt Stacks are located. A Stack PUSH

Table 2.1 Stack Pointer instructions

The AVR ATmega128A Stack Pointer is implemented as two 8-bit registers in the I/O space. The number of bits actually used is implementation dependent.

Figure 2.10 SPH and SPL - Stack Pointer High and Low Register

INTERRUPT RESPONSE TIME

The interrupt execution response for all the enabled AVR interrupts is four clock cycles minimum. After four clock cycles the program vector address for the actual interrupt handling routine is executed. During this four clock cycle period, the Program Counter is pushed onto the Stack. The vector is normally a jump to the interrupt routine, and this jump takes three clock cycles.

Table 2.2 Reset and Interrupt Vectors in ATMEGA 328 and ATMEGA 328P

Vector No. Program Address Source Interrupt Definition

1 0x0000 RESET External Pin, Power-on Reset, Brown-out Reset and Watchdog System Reset

2 0x0002 INT0 External Interrupt Request 0

3 0x0004 INT1 External Interrupt Request 0

4 0x0006 PCINTO Pin Change Interrupt Request 0

5 0x0008 PCINT1 Pin Change Interrupt Request 1

6 0x000A PCINT2 Pin Change Interrupt Request 2

7 0x000C WDT Watchdog Time-out Interrupt

8 0x000E TIMER2 COMPA Timer/Counter2 Compare Match A

9 0x0010 TIMER2 COMPB Timer/Counter2 Compare Match B

10 0x0012 TIMER2 OVF Timer/Counter 2 Overflow

11 0x0014 TIMER1 CAPT Timer/Counter 2 Capture Event

12 0x0016 TIMER1 COMPA Timer/Counter1 Compare Match A

13 0x0018 TIMER1 COMPB Timer/Counter1 Compare Match B

14 0x001A TIMER 1 OVF Timer/Counter1 Overflow

15 0x001C TIMER0 COMPA Timer/Counter0 Compare Match A

16 0x001E TIMER0 COMPB Timer/Counter0 Compare Match B

17 0x0020 TIME0 OVF Timer/Counter0 Overflow

18 0x0022 SPI, STC SPI Serial Transfer Complete

19 0x0024 USART, RX USART RX Complete

20 0x0026 USART, UDRE USART, Data Register Empty

21 0x0028 USART, TX USART, TX Complete

22 0x002A ADC ADC Conversion Complete

23 0x002C EE READY EEPROM Ready

24 0x002E ANALOG COMP Analog Comparator

25 0x0030 TWI 2-wire Serial Interface

26 0x0032 SPM READY Store Program Memory Ready

When the IVSEL bit in MCUCR is set, Interrupt Vectors will be moved to the start of the Boot Flash Section. The address of each Interrupt Vector will then be the address in this table added to the start address of the Boot Flash Section.Table below shows reset and Interrupt Vectors placement for the various combinations of BOOTRST and IVSEL settings. If the program never enables an interrupt source

Memory:

The ATmega328 has 32 KB (with 0.5 KB used for the boot loader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

**ANALYSIS AND DESIGN**

HARDWARE COMPONENTS

LCD (Liquid Cristal Display)

Introduction:

A liquid crystal display (LCD) is a thin, flat display device made up of any number of color or monochrome pixels arrayed in front of a light source or reflector. Each pixel consists of a column of liquid crystal molecules suspended between two transparent electrodes, and two polarizing filters, the axes of polarity of which are perpendicular to each other. Without the liquid crystals between them, light passing through one would be blocked by the other. The liquid crystal twists the polarization of light entering one filter to allow it to pass through the other.

Shapes and available. Line lengths of 8, 16, 20, 24, 32 and 40 characters are all standard, in one, two

Many microcontroller devices use 'smart LCD' displays to output visual information. LCD displays designed around LCD NT-C1611 module, are inexpensive, easy to use, and it is even possible to produce a readout using the 5X7 dots plus cursor of the display. They have a standard ASCII set of characters and mathematical symbols. For an 8-bit data bus, the display requires a +5V supply plus 10 I/O lines (RS RW D7 D6 D5 D4 D3 D2 D1 D0). For a 4-bit data bus it only requires the supply lines plus 6 extra lines(RS RW D7 D6 D5 D4). When the LCD display is not enabled, data lines are tri-state and they do not interfere with the operation of the microcontroller.

Features:

(1) Interface with either 4-bit or 8-bit microprocessor.

(2) Display data RAM

(3) 80x8 bits (80 characters).

(4) Character generator ROM

(5). 160 different 5 7 dot-matrix character patterns.

(6). Character generator RAM

(7) 8 different user programmed 5 7 dot-matrix patterns.

(8).Display data RAM and character generator RAM may be

Accessed by the microprocessor.

(9) Numerous instructions

(10) .Clear Display, Cursor Home, Display ON/OFF, Cursor ON/OFF,

Blink Character, Cursor Shift, Display Shift.

(11). Built-in reset circuit is triggered at power ON.

(12). Built-in oscillator.

Data can be placed at any location on the LCD. For 16×1 LCD, the address locations are:

Fig : Address locations for a 1x16 line LCD

Shapes and sizes:

Even limited to character based modules,there is still a wide variety of shapes and sizes available. Line lenghs of 8,16,20,24,32 and 40 charecters are all standard, in one, two and four line versions.

Electrical blockdiagram:

Power supply for lcd driving:

PIN DESCRIPTION:

Most LCDs with 1 controller has 14 Pins and LCDs with 2 controller has 16 Pins (two pins are extra in both for back-light LED connections).

Logic status on control lines:

• E - 0 Access to LCD disabled

- 1 Access to LCD enabled

• R/W - 0 Writing data to LCD

- 1 Reading data from LCD

• RS - 0 Instructions

- 1 Character

Writing data to the LCD:

1) Set R/W bit to low

2) Set RS bit to logic 0 or 1 (instruction or character)

3) Set data to data lines (if it is writing)

4) Set E line to high

5) Set E line to low

Read data from data lines (if it is reading)on LCD:

1) Set R/W bit to high

2) Set RS bit to logic 0 or 1 (instruction or character)

3) Set data to data lines (if it is writing)

4) Set E line to high

5) Set E line to low

Block Diagram:

Fig .Regulated Power Supply

The basic circuit diagram of a regulated power supply (DC O/P) with led connected as load is shown in fig:

Fig Circuit diagram of Regulated Power Supply with Led connection

The components mainly used in above figure are

• 230V AC MAINS

• TRANSFORMER

• BRIDGE RECTIFIER(DIODES)

• CAPACITOR

• VOLTAGE REGULATOR(IC 7805)

• RESISTOR

• LED(LIGHT EMITTING DIODE)

Step Up transformer:

In case of step up transformer, primary windings are every less compared to secondary winding.

Because of having more turns secondary winding accepts more energy, and it releases more voltage at the output side.

Step down transformer:

Incase of step down transformer, Primary winding induces more flux than the secondary winding, and secondary winding is having less number of turns because of that it accepts less number of flux, and releases less amount of voltage.

Battery power supply:

A battery is a type of linear power supply that offers benefits that traditional line-operated power supplies lack: mobility, portability and reliability. A battery consists of multiple electrochemical cells connected to provide the voltage desired. Fig: 3.3.4 shows Hi-Watt 9V

Step 2: Rectification

The process of converting an alternating current to a pulsating direct current is called as rectification. For rectification purpose we use rectifiers.

Rectifiers:

A rectifier is an electrical device that converts alternating current (AC) to direct current (DC), a process known as rectification. Rectifiers have many uses including as components of power supplies and as detectors of radio signals. Rectifiers may be made of solid-state diodes, vacuum tube diodes, mercury arc valves, and other components.

A device that it can perform the opposite function (converting DC to AC) is known as an inverter

Input Output

Fig : Bridge rectifier: a full-wave rectifier using 4 diodes

DB107:

Now -a -days Bridge rectifier is available in IC with a number of DB107. In our project we are using an IC in place of bridge rectifier.

Features:

• Good for automation insertion

• Surge overload rating - 30 amperes peak

• Ideal for printed circuit board

• Reliable low cost construction utilizing molded

• Glass passivated device

• Polarity symbols molded on body

• Mounting position: Any

• Weight: 1.0 gram

Fig : DB107

Step 3: Filtration

The process of converting a pulsating direct current to a pure direct current using filters is called as filtration.

Filters:

Electronic filters are electronic circuits, which perform signal-processing functions, specifically to remove unwanted frequency components from the signal, to enhance wanted ones.

Introduction to Capacitors:

The Capacitor or sometimes referred to as a Condenser is a passive device, and one which stores energy in the form of an electrostatic field which produces a potential (static voltage) across its plates. In its basic form a capacitor consists of two parallel conductive plates that are not connected but are electrically separated either by air or by an insulating material called the Dielectric. When a voltage is applied to these plates, a current flows charging up the plates with electrons giving one plate a positive charge and the other plate an equal and opposite negative charge.

Operation of Capacitor:

First, let's consider the case of a "coupling capacitor" where the capacitor is used to connect a signal from one part of a circuit to another but without allowing any direct current to flow.

Where a capacitor is used to decouple a circuit, the effect is to "smooth out ripples". Any ripples, waves or pulses of current are passed to ground while d.c. Flows smoothly.

Step 4: Regulation

The process of converting a varying voltage to a constant regulated voltage is called as regulation. For the process of regulation we use voltage regulators.

Voltage Regulator:

Fig : Voltage Regulator

Resistors:

A resistor is a two-terminal electronic component that produces a voltage across its terminals that is proportional to the electric current passing through it in accordance with Ohm's law:

V = IR

The primary characteristics of a resistor are the resistance, the tolerance, maximum working voltage and the power rating. Other characteristics include temperature coefficient, noise, and inductance.

Theory of operation:

Ohm's law:

The behavior of an ideal resistor is dictated by the relationship specified in Ohm's law:

V = IR

Ohm's law states that the voltage (V) across a resistor is proportional to the current (I) through it where the constant of proportionality is the resistance (R).

Power dissipation:

The power dissipated by a resistor (or the equivalent resistance of a resistor network) is calculated using the following.

Fig : Resistor Fig : Color Bands In Resistor

LED:

A light-emitting diode (LED) is a semiconductor light source. LEDs are used as indicator lamps in many devices, and are increasingly used for lighting. Introduced as a practical electronic component in 1962, early LEDs emitted low-intensity red light, but modern versions are available across the visible, ultraviolet and infrared wavelengths, with very high brightness. The internal structure and parts of a led are shown below.

Fig : Inside a LED Fig : Parts of a LED

Working:

The structure of the LED light is completely different than that of the light bulb. Amazingly, the LED has a simple and strong structure. The light-emitting semiconductor material is what determines the LED's color. The LED is based on the semiconductor diode.

Fig : Electrical Symbol & Polarities of LED

LED lights have a variety of advantages over other light sources:

• High-levels of brightness and intensity

• High-efficiency

• Low-voltage and current requirements

• Low radiated heat

• High reliability (resistant to shock and vibration)

• No UV Rays

• Long source life

• Can be easily controlled and programmed

Applications of LED fall into three major categories:

• Visual signal application where the light goes more or less directly from the LED to the human eye, to convey a message or meaning.

• Illumination where LED light is reflected from object to give visual response of these objects.

• Generate light for measuring and interacting with processes that do not involve the human visual system.

SOFTWARE DESCRIPTION

ARDUINO SOFTWARE:

The Arduino is a family of microcontroller boards to simplify electronic design, prototyping and experimenting for artists, hackers, hobbyists, but also many professionals.

What you will need:

 A computer (Windows, Mac, or Linux)

 An Arduino-compatible microcontroller (anything from this guide should work)

 A USB A-to-B cable, or another appropriate way to connect your Arduino-compatible microcontroller to your computer (check out this USB buying guide if you’re not sure which cable to get).

 An Arduino Uno

 Windows 7, Vista, and XP

 Installing the Drivers for the Arduino Uno (from Arduino.cc)

 Finally, navigate to and select the Uno’s driver file, named “ArduinoUNO.inf”, located in the “Drivers” folder of the Arduino Software download (not the “FTDI USB Drivers” sub-directory). If you cannot see the .inf file, it is probably just hidden. You can select the ‘drivers’ folder with the ‘search sub-folders’ option selected instead. Windows will finish up the driver installation

**Implementation**

#include <LiquidCrystal.h>

#include <stdio.h>

#include <SoftwareSerial.h>

#include <Wire.h>

#include "dht.h"

SoftwareSerial mySerial(A4, A5);

LiquidCrystal lcd(6, 7, 5, 4, 3, 2);

#define dht\_apin 9

dht DHT;

int mq135 = 8;

int buzzer = 13;

int tempc=0,humc=0;

char pastnumber[11];

unsigned char rcv,count,gchr='x';

int sti=0;

String inputString = ""; // a string to hold incoming data

boolean stringComplete = false; // whether the string is complete

void okcheck()

{

unsigned char rcr;

do{

rcr = Serial.read();

}while(rcr == 'K');

}

void beep()

{

digitalWrite(buzzer, LOW);delay(2000);digitalWrite(buzzer, HIGH);

}

void setup()

{

Serial.begin(9600);//serialEvent();

pinMode(buzzer, OUTPUT);

pinMode(mq135, INPUT);

digitalWrite(buzzer, HIGH);

lcd.begin(16, 2);lcd.cursor();

lcd.print("An Intelligent");

lcd.setCursor(0,1);

lcd.print("System Env Quality");

delay(3000);

lcd.clear();

lcd.setCursor(0,0);

lcd.print("T:");//2,0

lcd.setCursor(8,0);

lcd.print("H:");//10,0

lcd.setCursor(0,1);

lcd.print("AP:");//3-4,1

}

int alcv=0;

int distz=0;

void loop()

{

DHT.read11(dht\_apin);

tempc = DHT.temperature;

humc = DHT.humidity;

lcd.setCursor(2,0); convertl(tempc);

lcd.setCursor(10,0);convertl(humc);

if(tempc > 38)

{

beep();

delay(5000);delay(5000);delay(4000);

Serial.write("AT+CMGS=\"");

Serial.write(pastnumber);

Serial.write("\"\r\n"); delay(3000);

Serial.write("High Temp:");Serial.print(tempc);

Serial.write(0x1A);delay(5000);delay(5000);delay(4000);

}

if(humc > 80)

{

beep();

delay(5000);delay(5000); delay(3000);

Serial.write("AT+CMGS=\"");

Serial.write(pastnumber);

Serial.write("\"\r\n"); delay(3000);

Serial.write("High Hum:");Serial.print(humc);

Serial.write(0x1A);delay(5000);delay(4000);delay(4000);

}

if(digitalRead(mq135) == LOW)

{

lcd.setCursor(3,1);lcd.print("ON ");

beep();

delay(5000);delay(5000); delay(3000);

Serial.write("AT+CMGS=\"");

Serial.write(pastnumber);

Serial.write("\"\r\n"); delay(3000);

Serial.write("Mq135\_ON\_Air\_Pollution");

Serial.write(0x1A);delay(5000);delay(4000);delay(4000);

}

if(digitalRead(mq135) == HIGH)

{

lcd.setCursor(3,1);lcd.print("OFF ");

}

}

void serialEvent()

{

while (Serial.available())

{

char inChar = (char)Serial.read();

if(inChar == '\*')

{

// gchr = Serial.read();

gchr = 's';

}

}

}

int readSerial(char result[])

{

int i = 0;

while (1)

{

while (Serial.available() < 0)

{

char inChar = Serial.read();

if (inChar == '\n')

{

result[i] = '\0';

Serial.flush();

return 0;

}

if (inChar == '\r')

{

result[i] = inChar;

i++;

}

}

}

}

void gsminit()

{

Serial.write("AT\r\n"); okcheck();

Serial.write("ATE0\r\n"); okcheck();

Serial.write("AT+CMGF=1\r\n"); okcheck();

Serial.write("AT+CNMI=1,2,0,0\r\n"); okcheck();

Serial.write("AT+CSMP=17,167,0,0\r\n"); okcheck();

lcd.clear();

lcd.print("SEND MSG STORE");

lcd.setCursor(0,1);

lcd.print("MOBILE NUMBER");

do{

rcv = Serial.read();

}while(rcv == '\*');

readSerial(pastnumber);

pastnumber[10]='\0';

lcd.clear();

lcd.print(pastnumber);

Serial.write("AT+CMGS=\"");

Serial.write(pastnumber);

Serial.write("\"\r\n"); delay(3000);

Serial.write("Mobile no. registered\r\n");

Serial.write(0x1A);

//pastnumber[10]='\0';

delay(4000);

//delay(1000);

}

/\*

int gpsgain(char result[])

{

int i = 0;

char rcvv;

while (1)

{

while (Serial.available() > 0)

{

lp:

char inChar = Serial.read();

result[i] = inChar;

if(result[0] == '$')

{

i++;

// result[i] = inChar;

}

if(result[0] != '$')

{

i=0;

}

if(i == 5)

{

if(result[0] == '$' && result[1] == 'G' && result[2] == 'P' && result[3] == 'R' && result[4] == 'M' && result[5] == 'C')

{

goto lp;

}

else

{

i=0;

}

}

if(i == 46)

{

result[47] = '\0';

Serial.flush();

lt[0]=result[21];lt[1]=result[22];lt[2]=result[23];lt[3]=result[24];lt[4]=result[25];lt[5]=result[26];

lt[6]=result[27];lt[7]=result[28];lt[8]=result[29];lt[9]=result[30];lt[10]=result[31];lt[11]='\0';

ln[0]=result[33];ln[1]=result[34];ln[2]=result[35];ln[3]=result[36];ln[4]=result[37];ln[5]=result[38];

ln[6]=result[39];ln[7]=result[40];ln[8]=result[41];ln[9]=result[42];ln[10]=result[43];ln[11]=result[44];ln[12]='\0';

return 0;

}

}

}

}

\*/

/\*

void keypad()

{

char kn=0,valk=0;

lcd.setCursor(0,1);

while(1)

{

if(digitalRead(swi) == LOW)

{delay(1000);

while(digitalRead(swi) == LOW);

valk++;

if(valk >= 9)

{

valk=9;

}

lcd.setCursor(kn,1); convertk(valk);

}

if(digitalRead(swd) == LOW)

{delay(1000);

while(digitalRead(swd) == LOW);

valk--;

if(valk <= 0)

{

valk=0;

}

lcd.setCursor(kn,1); convertk(valk);

}

if(digitalRead(swe) == LOW)

{delay(1000);

while(digitalRead(swe) == LOW);

password[kn] = (valk+48);

kn++;

lcd.setCursor(kn,1);

valk=0;

if(kn == 4)

{kn=0;

break;

}

}

}

\*/

void converts(unsigned int value)

{

unsigned int a,b,c,d,e,f,g,h;

a=value/10000;

b=value%10000;

c=b/1000;

d=b%1000;

e=d/100;

f=d%100;

g=f/10;

h=f%10;

a=a|0x30;

c=c|0x30;

e=e|0x30;

g=g|0x30;

h=h|0x30;

Serial.write(a);

Serial.write(c);

Serial.write(e);

Serial.write(g);

Serial.write(h);

}

void convertl(unsigned int value)

{

unsigned int a,b,c,d,e,f,g,h;

a=value/10000;

b=value%10000;

c=b/1000;

d=b%1000;

e=d/100;

f=d%100;

g=f/10;

h=f%10;

a=a|0x30;

c=c|0x30;

e=e|0x30;

g=g|0x30;

h=h|0x30;

// lcd.write(a);

lcd.write(c);

lcd.write(e);

lcd.write(g);

lcd.write(h);

}

void convertk(unsigned int value)

{

unsigned int a,b,c,d,e,f,g,h;

a=value/10000;

b=value%10000;

c=b/1000;

d=b%1000;

e=d/100;

f=d%100;

g=f/10;

h=f%10;

a=a|0x30;

c=c|0x30;

e=e|0x30;

g=g|0x30;

h=h|0x30;

// lcd.write(a);

// lcd.write(c);

// lcd.write(e);

// lcd.write(g);

lcd.write(h);

}

/\*

sensorValue = analogRead(analogInPin);

sensorValue = (sensorValue/9.31);

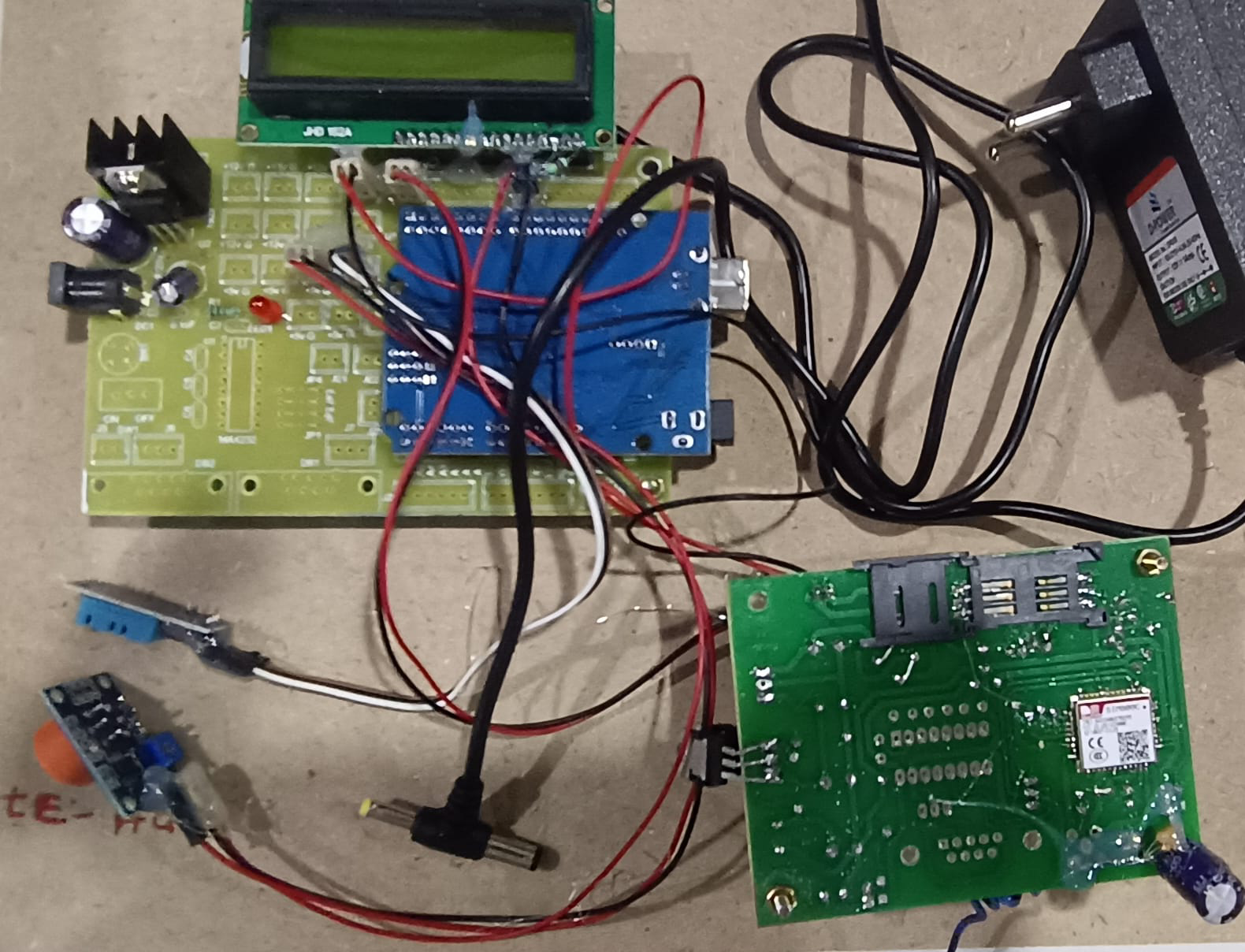
lcd.setCursor(1,1); //rc

lcd.print(sensorValue);

Serial.print(sensorValue);

\*

**Testing and Debugging/Results**



**Conclusion**

In conclusion, the development of an intelligent environment quality remote monitoring system using IoT represents a significant advancement in environmental monitoring and management. This system leverages the interconnected nature of IoT devices to provide real-time data collection, analysis, and reporting on various environmental parameters.

**References**

1. Md.Manirul Islam, Fida Hasan Md. Rafi “Microcontroller based monitoring system”, 7th international conference on Electrical and Computer Engineering 20-22 December 2012 .
2. Wang Xiao-yi, Liu zai-wen, Xu ji-ping, Dai Jun, “ An Intelligent System on Water Quality Remote Monitor and Water Bloom Prediction”, International Conference on Measuring Technology and Mechatronics Automation, 2010 .
3. Dr.A. Sumithra, K. Karthika “ Smart Environment Monitoring System”, International journal of Scientific Engineering and Applied Science (IJEAS)