ACKNOWLEDGEMENT

We thank our Principal Mr. Senthil Kumaran S MSc, M. phil., for the kind advice for this project.

We are extremely indebted to the Head of the Department Mr. P. Sridhar M.E., who provided us great advice, timely help, acknowledgeable assistance and useful recommendations

We own our gratitude to ours Lecturers Tmt. B. Kavitha M.E., Tmt. V. Manga M.E., Tmt. D. Juliet Thessalonica M.E., Selvi. V. Priya M.E., for their kind attention and valuable suggestions during the project.

We are also in great regard to our fellow students for their enthusiasm, during to the course of this project. Our special thanks to Mr. M. Vairam (skilled Assistant) for making this project a successful by providing excellent guidance.

ABSTRACT

The system proposed in this paper is an advanced solution for monitoring the weather conditions at a particular place and make the information visible anywhere in the world. The technology behind this is Internet of Things (IoT), which is an advanced and efficient solution for connecting the things to the internet and to connect the entire world of things in a network. The data updates from the implemented system can be accessible in the internet from anywhere in the world. In agriculture zone it will be very difficult to check and monitor the weather parameter through wires and analog devices during some weather hazards. To overcome this problem here the wireless sensors are used to check and monitor the weather parameters. The other idea is Vertical farming system. It is implemented for cultivating different crops in small area.

Now-a-days many weather reporting applications are available which gives us information about climatic changes that are going to take place by which man can be aware of present and 4 future climatic changes. Most of the weather reporting applications extracts the data from accurate weather system. Here we are building our own weather reporting system which would give us information about present temperature, humidity etc. We can setup this in our home and get time to time changes in climate which would help us in planning our daily work easily. Like it would be helpful for a farmer in this agricultural activity by which he can protect his crops climatic changes. It would help in transportation giving information of weather conditions etc.

CHAPTER-1

INTRODUCTION

Here we introduce a smart weather reporting system over the Internet. Our introduced system allows for weather parameter reporting over the Internet. It allows the people to directly check the weather states online without the need of a weather forecasting agency. System uses temperature, humidity as well as rain with humidity sensor to monitor weather and provide live reporting of the weather statistics. The system constantly monitors temperature using temperature sensor, humidity using humidity sensor and also for rain. Weather monitoring system deals with detecting and gathering various weather parameters at different locations which can be analysed or used for weather forecasting. The aim of this system is achieved by technologies such as Internet of Things (IOT) and Cloud. The idea of internet of things is to connect a device to the internet and to other required connected devices. Using internet, the information from the IOT device can easily be transferred to the cloud and then from the cloud to the end user. Weather Monitoring is an essential practical implementation of the concept of Internet of Things, it involves sensing and recording various weather parameters and using them for alerts, sending notifications, adjusting appliances accordingly and also for long term analysis. Also, we will try to identify and display trends in parameters using graphical representation. The devices used for this purpose are used to collect, organize and display information. It is expected that the internet of things is going to transform the world by monitoring and controlling the phenomenon of environment by using sensors/devices which are able to capture, process and transmit weather parameters. Cloud is availability of computer system resources like data storage, computing power without direct active management of user. The data captured is transmitted to the cloud so that the data could be further displayed. Besides this, the system consists of components such as Arduino UNO board which is a microcontroller board consisting of 14 digital pins, a USB connection and everything used to support microcontroller; DHT11 is Temperature and humidity sensor which is used for detecting these mentioned parameters; WIFI module is used to convert the data collected from the sensors and then send it to the web server. So, in this way weather conditions of any location can be monitored from any remote location in the world. The system constantly transmits this data to the micro controller which now processes this data and keeps on transmitting it to the online web server over a Wi-Fi connection. This data is live updated to be viewed on the online server system. Also, system allows user to set alerts for particular instances. In today's world many pollution monitoring systems are designed by different environmental parameters. Existing system model is presented IOT based Weather monitoring and reporting system where you can collect, process, analyse, and present your measured data on web server. Wireless sensor network management model consists of end device, router, gateway node and management monitoring centre. End device is responsible for collecting wireless sensor network data, and sending them to parent node, then data are sent to gateway node from parent node directly or by router. After receiving the data from wireless sensor network, gateway node extracts data after analysing and packaging them into Ethernet format data, sends them to the server. Less formally, any device that runs server software.

CHAPTER-2

EXISTING SYSTEM

There are many methods which are helpful to calculate the weather parameters. Manual methods need to take the readings at the place of the station by human being. This method of traditional approach is accurate and depend on the person who takes the values. Before going for any method, we must know the definitions and standard unit of the weather parameters. A manual inventory system is relying heavily on the action of the people which increases the possibilities of human error.

CHAPTER-3

LITERATURE REVIEW

Through weather monitoring system we can collect the information about humidity and temperature and according to current and previous data we can produce the results in graphical manner in the system. After reviewing many articles, there are presently no papers that mention monitoring the combination of temperature, lighting and humidity in one integrated system and have actuators to modify these settings. In addition to this, there is one research paper that has discussed monitoring these three environmental conditions; however, there has been no mention about having actuators to modify. So, our main idea was to coin a system that can sense the main components that formulates the weather and can be able to forecast the weather without human error. Ancient weather forecasting methods usually relied on observed patterns of events, also termed pattern recognition. For example, it might be observed that if the sunset was particularly red, the following day often brought fair weather. This experience accumulated over the generations to produce weather lore. However, not all of these predictions prove reliable, and many of them have since been found not to stand up to rigorous statistical testing. The simplest method of forecasting the weather, persistence, relies upon today's conditions to forecast the conditions tomorrow. This can be a valid way of forecasting the weather when it is in a steady state, such as during the summer season in the tropics. This method of forecasting strongly depends upon the presence of a stagnant weather pattern. It can be useful in both short-range forecasts and long-range forecasts. Measurements of barometric pressure and the pressure tendency (the change of pressure over time) have been used in forecasting since the late 19th century.

CHAPTER-4

PROPOSED SYSTEM

The core component of this system is the Arduino Uno which has a microcontroller i.e., Atmega 328. The Atmega has a 32KB flash, it needed to burn a boot loader and download Arduino sketches. The boot loader is programmed under ISP program controller. Arduino measures 4 weather parameters using 5 sensors in IOT Weather Monitoring System. These sensors are a Temperature and Humidity sensor, Light sensor, Air pressure sensor, Air Quality sensor. These 5 sensors are directly connected to the Arduino Uno since it has an inbuilt Analog to Digital converter.

BLOCK DIAGRAM

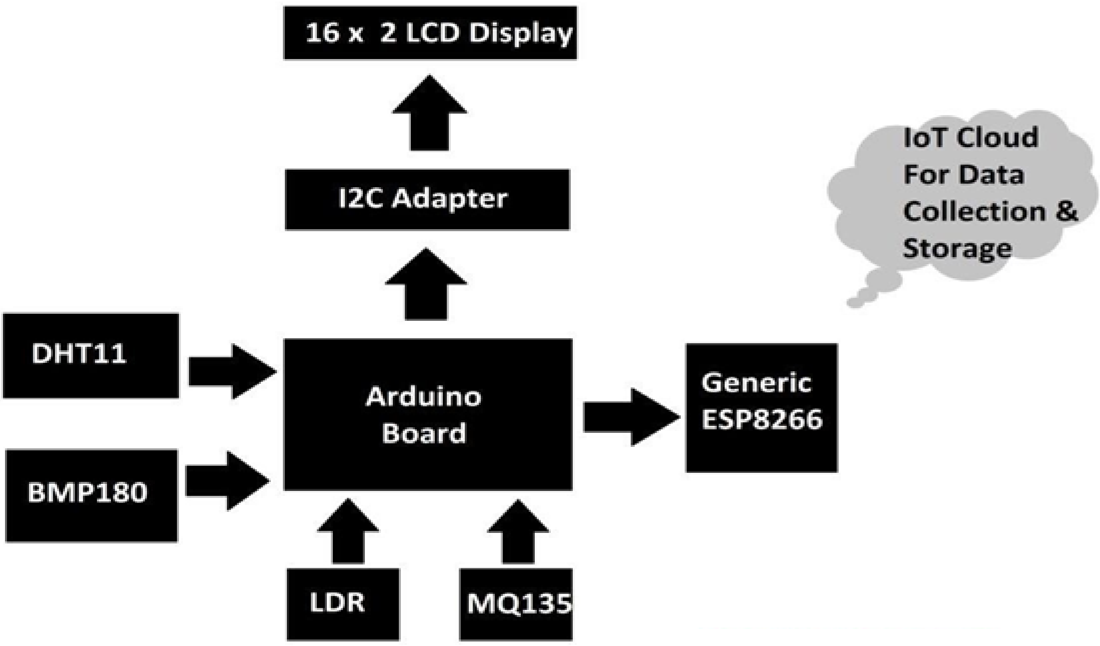


Fig 4.1 Block Diagram of IoT Weather Monitoring System

We are going to develop the weather monitoring system using the above illustrated blocks. The brain of the project is an Arduino Uno board and the surrounding blocks are Digital and Analog sensors for acquiring local weather and environment data.

A generic ESP8266 is used for interfacing the circuit setup with internet via 2.4 GHz Wi-Fi band. The ESP8266 sends the sensor data to a cloud server where the data gets updated in real time and also gets stored for future analysis. We are utilizing a 16 x 2 LCD display to showcase the sensor data, so that we can observe real-time data locally.

CHAPTER-5

HARDWARE REQUIREMENTS

5.1 ARDUINO UNO

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc and initially released in 2010. The board is equipped with sets of digital and analog input/output pins that may be interfaced to various expansion boards and other circuit.

5.1.1 PIN DIAGRAM

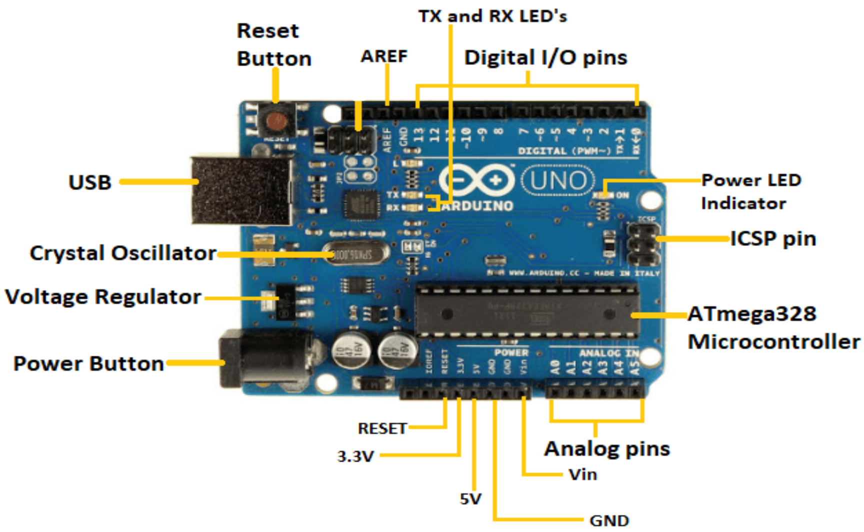


Fig 5.1Arduino Pin Diagram

5.1.2 PIN DESCRIPTION OF ARDUINO UNO

Pin Category	Pin Name	Details
Power	Vin, 3.3V, 5V, GND	Vin: input voltage to Arduino when external power source. 5V: regulated power supply used to power microcontroller and other components on the board. 3.3V: 3.3V supply generated by on board voltage regulator. Maximum current draw is 50mA. GND: ground pins.
Reset	Reset	Resets the microcontroller.
Analog pins	A0 – A5	Used to provide analog input in the range of 0-5V.

Input/output pins	Digital pins 0 - 13	Can be used as input or output pins.
serial	0(Rx), 1(Tx)	Used to receive and transmit TTL serial data.
External interrupts	2,3	To trigger an interrupt.
PWM	3, 5, 6, 9, 11	Provides 8-bit PWM output
SPI	10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK)	Used for SPI communication.
Inbuild LED	13	To turn on the inbuild LED.
TWI	A4 (SDA), A5 (SCA)	Used for TWI communication.
AREF	AREF	To provide reference voltage for input voltage.

Table 5.1 Pin Description of Arduino

5.1.3 ADVANTAGES

- Not much knowledge required to get started.
- Fairly low cost, depending on shields you need.
- Lots of sketches and shields available.
- No external programmer or power supply needed.

5.2 16x2 LCD DISPLAY

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and very commonly used in various devices and circuits. These modules are preferred over screen segments and other multi segments LEDs. The reasons being: LCDs are economical, easily programmable; have no limitation of displaying special & even custom characters (unlike in seven segments), animations and so no.

A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data.

The Command register stores the command instructions given to the LCD. A Command is an instruction given to the LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD Click to learn more about internal structure of an LCD.

5.2.1 INTRODUCTION

The most commonly used Character based LCDs are based on Hitachi's HD44780 controller or other which are compatible with HD44580. In this tutorial, we will discuss about character-based LCDs, their interfacing with various microcontroller, various interfaces (8-bit/4-bit), programming, special stuff and tricks you can do with these simple looking LCDs which can give a new look to your application.

5.2.2 PIN DIAGRAM

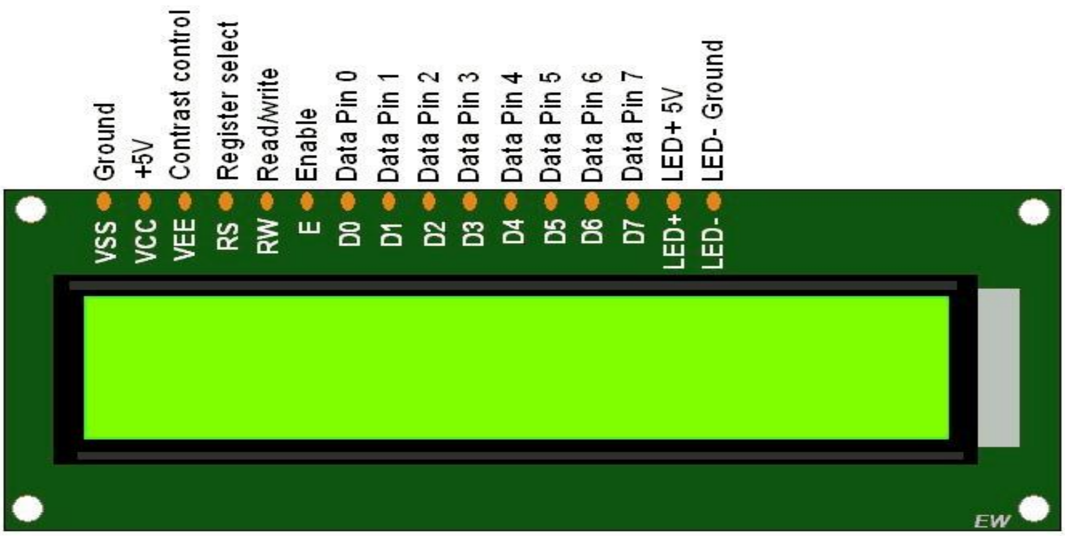


Fig 5.2 Pin Diagram of LCD

5.2.3 PIN DESCRIPTION

The most commonly used LCDs found in the market today are 1 Line, 2 Line or 4 Line LCDs which have only 1 controller and support at most of 80 characters, whereas LCDs supporting more than 80 characters make use of 2 HD44780 controllers.

Most LCDs with 1 controller have 14 pins and LCDs with 2 controllers has 16 Pins (two pins are extra in both for back-light LED connections). Pin description is shown in the table below.

Pin No	Function	Name
1	Ground(0V)	Ground
2	Supply voltage, 5V (4.7V – 5.3V)	Vcc
3	Contract adjustment, through a variable resistor	V_{EE}
4	Selects command register when low; and data register when high	Register Select
5	Low to write to the register; High to read from the register	Read/Write
6	Sends data to data pins when a high to low pulse is given	Enable
7	8-bit data pins	DB0
8		DB1
9		DB2
10		DB3
11		DB4
12		DB5
13		DB6
14		DB7
15		LED+
16		LED-

Table 5.2Pin Description of LCD

5.3 DHT11 TEMPERATURE AND HUMIDITY SENSOR

5.3.1 INTRODUCTION

The DHT11 is a basic, ultra-low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin (no Analog input pins needed). This sensor is used in various applications such as measuring humidity and temperature values in heating, ventilation and air conditioning systems. Weather stations also use these sensors to predict weather conditions. The humidity sensor is used as a preventive measure in homes where people are affected by humidity. It's compact size and sampling rate made this sensor popular among hobbyists. Some of the sensors which can be used as an alternative to DHT11 sensor are DHT22, AM2302 and SHT71.

DHT11 sensor consists of a capacitive humidity sensing element and a thermistor for sensing temperature. The humidity sensing capacitor has two electrodes with a moisture holding substrate as a dielectric between them. Change in the capacitance value occurs with the change in humidity levels. The IC measure, process this changed resistance values and change them into digital form. For measuring temperature this sensor uses a Negative Temperature coefficient thermistor, which causes a decrease in its resistance value with increase in temperature. To get larger resistance value even for the smallest change in temperature, this sensor is usually made up of semiconductor ceramics or polymers.The temperature range of DHT11 is from 0 to 50 degree Celsius with a 2-degree accuracy. Humidity range of this sensor is from 20 to 80% with 5% accuracy. The sampling rate of this sensor is 1Hz.i.e., it gives one reading for every second. DHT11 is small in size with operating voltage from 3 to 5 volts. The maximum current used while measuring is 2.5mA.

5.3.2 PIN DIAGRAM

Fig 5.3Pin diagram of DHT11

5.3.3 PIN DESCRIPTION

DHT11 is a temperature and humidity sensors, it has a dedicated NTC to measure the environment temperature, and an 8-bit microcontroller inside that can generate the output in the values of temperature as a serial data. The serial pin is connected with one of the digital PIN in ESP32 microcontroller.

Pin No	Pin Name	Pin Description
1	VCC	Power supply 3.3 to 5.5 Volt DC
2	DATA	Digital output pin
3	NC	Not in use
4	GND	Ground

Table 5.3 Pin Description of DHT11

5.3.4 FEATURES

- Ultra-low cost.
- 3 to 5V power and I/O.
- 2.5mA max current use during conversion (while requesting data)
- Good for 20-80% humidity readings with 5% accuracy.
- Good for 0-50°C temperature readings ±2°C accuracy.
- No more than 1 Hz sampling rate (once every second)
- Body size 15.5mm x 12mm x 5.5mm.

5.4 BMP180 BAROMETRIC SNESOR

A **barometric** is a scientific instrument that is used to measure air pressure in a certain environment. Pressure tendency can forecast short term changes in the weather. Many measurements of air pressure are used within surface weather analysis to help find surface troughs, pressure systems and frontal boundaries.

Barometric and pressure altimeters (the most basic and common type of altimeter) are essentially the same instrument, but used for different purposes. An altimeter is intended to be used at different levels matching the corresponding atmospheric pressure to the altitude, while a barometer is kept at the same level and measures subtle pressure changes caused by weather and elements of weather. The average atmospheric pressure on the earth's surface varies between 940 and 1040 hPa (mbar). The average atmospheric pressure at sea level is 1013 hPa (mbar).

5.4.1 INTRODUCTION

Barometric pressure and the pressure tendency (the change of pressure over time) have been used in weather forecasting since the late 19th century. When used in combination with wind observations, reasonably accurate short-term forecasts can be made. Simultaneous barometric readings from across a network of weather stations allow maps of air pressure to be produced, which were the first form of the modern weather map when created in the 19th century. Isobars, lines of equal pressure, when drawn on such a map, give a contour map showing areas of high and low pressure. Localized high atmospheric pressure acts as a barrier to approaching weather systems, diverting their course. Atmospheric lift caused by low-level wind convergence into the surface brings clouds and sometimes precipitation. The larger the change in pressure, especially if more than 3.5 hPa (0.1 inHg), the greater the change in weather that can be expected. If the pressure drop is rapid, a low pressure system is approaching, and there is a greater chance of rain. Rapid pressure rises, such as in the wake of a cold front, are associated with improving weather conditions, such as clearing skies.

With falling air pressure, gases trapped within the coal in deep mines can escape more freely. Thus, low pressure increases the risk of firedamp accumulating. Collieries therefore keep track of the pressure. In the case of the Trimdon Grange colliery disaster of 1882 the mines inspector drew attention to the records and in the report stated "the conditions of atmosphere and temperature may be taken to have reached a dangerous point". Aneroid barometers are used in scuba diving. A submersible pressure gauge is used to keep track of the contents of the diver's air tank. Another gauge is used to measure the hydrostatic pressure, usually expressed as a depth of sea water. Either or both gauges may be replaced with electronic variants or a dive computer.

5.4.2 PIN DIAGRAM

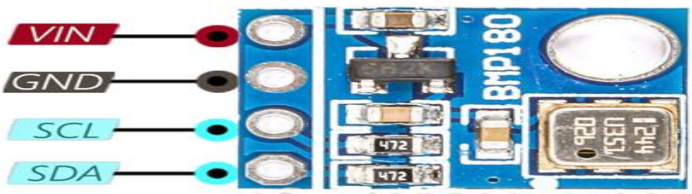


Fig 5.4 Pin Diagram of BMP180

5.4.3 PIN DESCRIPTION

Pin Name	Description
VCC	Connected to +5V
GND	Connected to ground.
SDA	Serial Data pin (I2C interface)
SCL	Serial Clock pin (I2C interface)
3.3V	If +5V is not present. Can power module by connecting +3.3V to this pin

Table 5.4 Pin Description of BMP180

5.4.4 BMP180SENSOR FEATURES

- Can measure temperature and altitude.
- Pressure range: 300 to 1100hPa
- High relative accuracy of ±0.12hPa
- Can work on low voltages
- 3.4Mhz I2C interface
- Low power consumption (3uA)
- Pressure conversion time: 5msec
- Potable size3.5 MQ-135 Air Quality Sensor

5.5 MQ-135 AIR QUALITY SENSOR

The gas sensing material used in the MQ-135 sensor is tin dioxide (SnO2), which has low conductivity in clean air. When there is polluted gas in the environment where the sensor is located, the conductivity of the sensor increases with the increase of the concentration of polluted gas in the air. The MQ-135 sensor has a high sensitivity to ammonia, sulphide, and benzene-based vapours, and is ideal for monitoring smoke and other harmful gases and is low-cost sensor suitable for a variety of applications. The operating voltage of MQ-135 is 5V and consumes around 160mA, the sensor has built-in heater for heating the sensor for its normal operation and if the sensor is exposed to strong wind, we may get incorrect readings. The sensor takes typically around 3 to 5 minutes to reach optimum temperature depending on surrounding air flow. The sensor has good sensitivity to detect the above-mentioned gases, but the disadvantage is it cannot differentiate which gas or gases have been detected.

5.5.1 FEATURES

- Sensitive for benzene, alcohol, smoke
- Output voltage boosts along with the concentration of the measured gases increases
- Fast response and recovery
- Adjustment sensitivity
- Signal output indicator

5.5.2 SPECIFICATIONS

- Power: 2.5V~ 5.0V
- Dimension: 40.0mm * 21.0mm
- Mounting holes size: 2.0mm
- Sensitive gases: Ammonia, Nitrogen oxides, Alcohols, Aromatic compounds, Sulphides, Fumes

5.5.3 PIN DIAGRAM

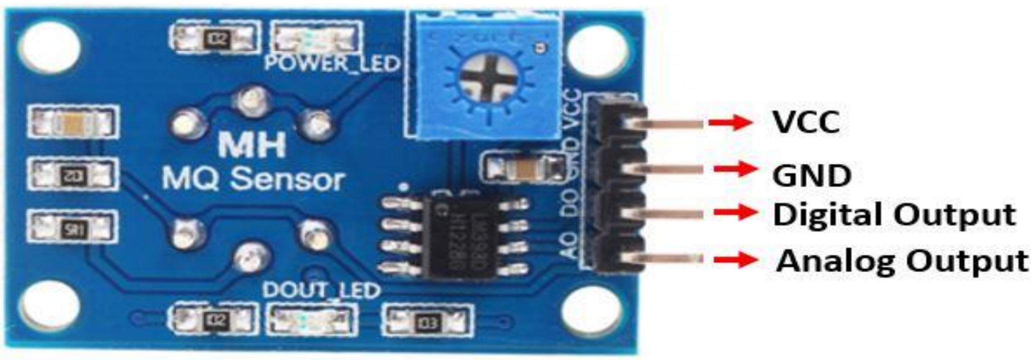


Fig 5.5 Pin Diagram of MQ-135

5.5.4 PIN DESCRIPTION

Pin	Description
VCC	Power input (2.5V ~ 5.0V)
GND	Ground
Digital Output	Digital data output
Analog Output	Analog data output

Table 5.5 Pin Description of MQ-135

5.6 LIGHT DEPENDENT RESISTOR – LDR

The photoresistor (also known as a photocell, Light Dependent Resistor (LDR) or photo-conductive cell) is a passive component that decreases resistance with respect to receiving luminosity (light) on the component's sensitive surface. The resistance of the photoresistor decreases with increases in incident light intensity; in other words, it exhibits photoconductivity. A photoresistor can be applied in light-sensitive detector circuits and light-activated and dark-activated switching circuits acting as a resistance semiconductor. In the dark, a photoresistor can have a resistance as high as several megaohms, while in the light, a photoresistor can have a resistance as low as a few hundred ohms. If incident light on a photoresistor exceeds a certain frequency, photons absorbed by the semiconductor give bound electrons enough energy to jump into the conduction band. The resulting free electrons (and their hole partners) conduct electricity, thereby lowering resistance. The resistance range and sensitivity of a photoresistor can substantially differ among dissimilar devices. Moreover, unique photoresistor may react substantially differently to photons within certain wavelength bands.

A photoelectric device can be either intrinsic or extrinsic. An intrinsic semiconductor has its own charge carriers and is not an efficient semiconductor, for example, silicon. In intrinsic devices, most of the available electrons are in the valence band, and hence the photon must have enough energy to excite the electron across the entire bandgap. Extrinsic devices have impurities, also called dopants, added whose ground state energy is closer to the conduction band; since the electrons do not have as far to jump, lower energy photons (that is, longer wavelengths and lower frequencies) are sufficient to trigger the device. If a sample of silicon has some of its atoms replaced by phosphorus atoms (impurities), there will be extra electrons available for conduction. This is an example of an extrinsic semiconductor.

5.6.1 WORKING PRINCIPAL

The working principle of an LDR is photoconductivity, which is nothing but an optical phenomenon. When the light is absorbed by the material then the conductivity of the material enhances. When the light falls on the LDR, then the electrons in the valence band of the material are eager to the conduction band. But, the photons in the incident light must have energy superior to the bandgap of the material to make the electrons jump from one band to another band (valance to conduction).

Hence, when light having ample energy, more electrons are excited to the conduction band which grades in a large number of charge carriers. When the effect of this process and the flow of the current starts flowing more, the resistance of the device decreases.

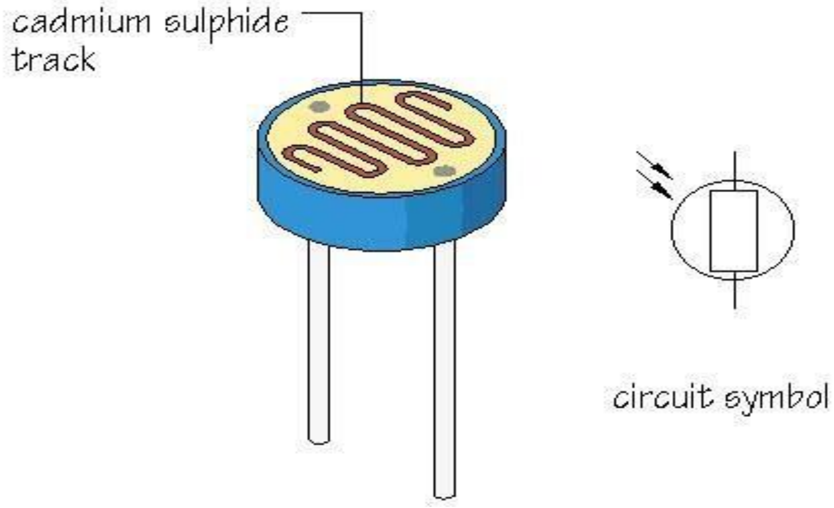


Fig 5.6 Block Diagram of LDR

5.6.2 LDR SPECIFICATIONS

The LDR specifications mainly include maximum power dissipation, maximum operating voltage, peak wavelength, dark resistance, etc. The values of these specifications mentioned below.

- Maximum power dissipation is 200mW
- The maximum voltage at 0 lux is 200V
- The peak wavelength is 600nm
- Minimum resistance at 10lux is 1.8kΩ
- Maximum resistance at 10lux is 4.5kΩ
- Typical resistance at100lux is 0.7kΩ
- Dark resistance after 1 sec is 0.03MΩ
- Dark resistance after 5 sec is 0.25MΩ

5.6.3 LDR ADVANTAGES

The **advantages of LDR** include the following.

- Sensitivity is High
- Simple & Small devices
- Easily used

- Inexpensive
- There is no union potential.
- The light-dark resistance ratio is high.
- Its connection is simple

5.6.4 LDR DISADVANTAGES

The **disadvantages of LDR** include the following.

- Spectral response is narrow
- Hysteresis effect
- Temperature stability is low for the best materials
- In stable materials, its responses very slowly
- The use of LDR is limited where the light signal changes very quickly
- It is not so much a responsive device.
- It provides incorrect result once working temperature alters

5.7 NODEMCU ESP8266 Wi-Fi MODULE

The NodeMCU (Node MicroController Unit) is an open-source software and hardware development environment built around an inexpensive System-on-a-Chip (SoC) called the ESP8266.The ESP8266, designed and manufactured by Espressif Systems, contains the crucial elements of a computer: CPU, RAM, networking (Wi-Fi), and even a modern operating system and SDK. That makes it an excellent choice for Internet of Things (IoT) projects of all kinds. However, as a chip, the ESP8266 is also hard to access and use. You must solder wires, with the appropriate analog voltage, to its pins for the simplest tasks such as powering it on or sending a keystroke to the “computer” on the chip. You also have to program it in low-level machine instructions that can be interpreted by the chip hardware. This level of integration is not a problem using the ESP8266 as an embedded controller chip in mass-produced electronics. It is a huge burden for hobbyists, hackers, or students who want to experiment with it in their own IoT projects.The ESP8266 is a low-cost Wi-Fi microchip, with built-in TCP/IP networking software, and microcontroller capability, produced by Espressif Systems in Shanghai, China.

The chip was popularized in the English-speaking maker community in August 2014 via the ESP-01 module, made by a third-party manufacturer Ai-Thinker. This small module allows microcontrollers to connect to a Wi-Fi network and make simple TCP/IP connections using Hayes-style commands. However, at first, there was almost no English-language documentation on the chip and the commands it accepted. The very low price and the fact that there were very few external components on the module, which suggested that it could eventually be very inexpensive in volume, attracted many hackers to explore the module, the chip, and the software on it, as well as to translate the Chinese documentation.

The ESP8285 is a similar chip with a built-in 1 MiB flash memory, allowing the design of single-chip devices capable of connecting via Wi-Fi.

5.7.1 FEATURES

- Processor:L106 32-bit RISC microprocessor core based on the Tensilica Diamond Standard 106Micro running at 80 or 160 MHz
- Memory:
    - 32 KiB instruction RAM
    - 32 KiB instruction cache RAM
    - 80 KiB user-data RAM
    - 16 KiB ETS system-data RAM
- External QSPI flash: up to 16 MiB is supported (512 KiB to 4 MiB typically included)
- IEEE 802.11 b/g/n Wi-Fi
    - Integrated TR switch, balun, LNA, power amplifier and matching network
    - WEP or WPA/WPA2 authentication, or open networks
- 17 GPIO pins
- Serial Peripheral Interface Bus (SPI)
- I²C (software implementation)
- I²S interfaces with DMA (sharing pins with GPIO)
- UART on dedicated pins, plus a transmit-only UART can be enabled on GPIO2
- 10-bit ADC (successive approximation ADC)

5.7.2 PIN DIAGRAM

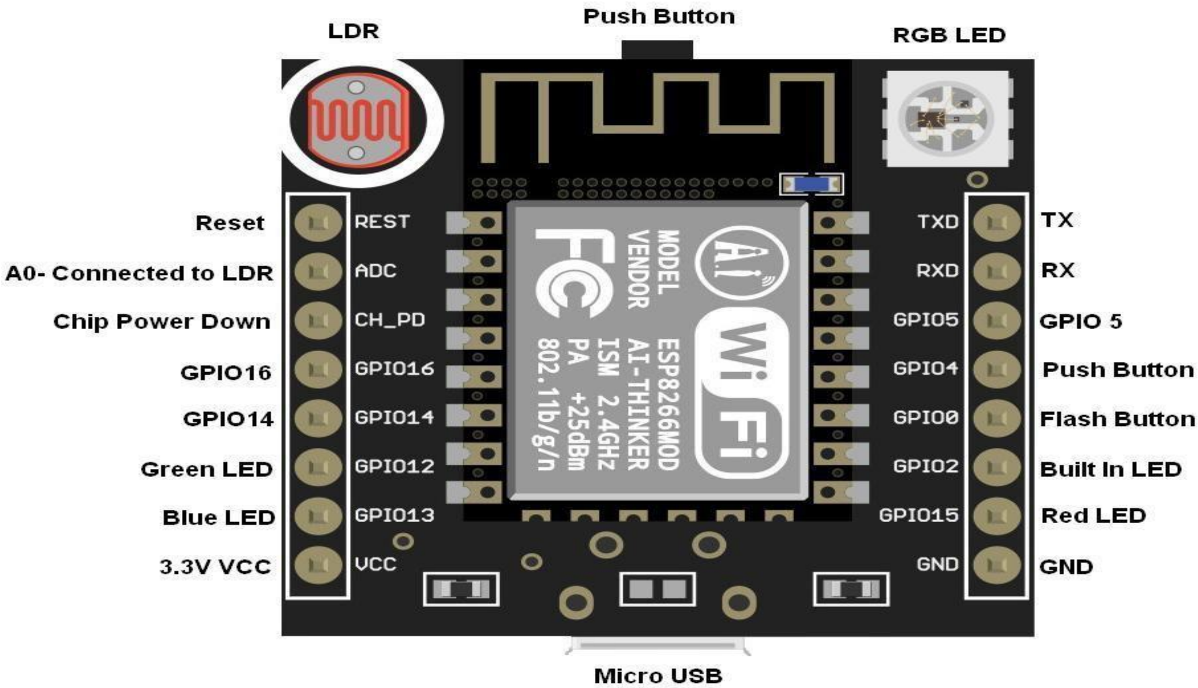


Fig 3.7 Pin Diagram of NodeMCU

5.7.3 PIN DESCRIPTION

Pin Category	Name	Description
Power	Micro-USB, 3.3V, GND, Vin	**Micro-USB:** NodeMCU can be powered through the USB port **3.3V:** Regulated 3.3V can be supplied to this pin to power the board **GND:** Ground pins **Vin:** External Power Supply
Control Pins	EN, RST	The pin and the button reset the microcontroller

Analog Pin	A0	Used to measure Analog voltage in the range of 0-3.3V
GPIO Pins	GPIO1 to GPIO16	NodeMCU has 16 general purpose input-output pins on its board
SPI Pins	SD1, CMD, SD0, CLK	NodeMCU has four pins available for SPI communication.
UART Pins	TXD0, RXD0, TXD2, RXD2	NodeMCU has two UART interfaces, UART0 (RXD0 & TXD0) and UART1 (RXD1 & TXD1). UART1 is used to upload the firmware/program.
I2C Pins		NodeMCU has I2C functionality support but due to the internal functionality of these pins, you have to find which pin is I2C.

Table 3.7 Pin Description of NodeMCU

5.8 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. A **breadboard**, **solderless breadboard**, or **protoboard** is a construction base used to build semi-permanent prototypes of electronic circuits. Unlike a perfboard or stripboard, breadboards do not require soldering or destruction of tracks and are hence reusable. For this reason, breadboards are also popular with students and in technological education.

A variety of electronic systems may be prototyped by using breadboards, from small Analog and digital circuits to complete central processing units (CPUs).

Compared to more permanent circuit connection methods, modern breadboards have high parasitic capacitance, relatively high resistance, and less reliable connections, which are subject to jostle and physical degradation. Signalling is limited to about 10 MHz, and not everything works properly even well below that frequency.

There are two major types of breadboards; these are solder and solderless boards. Solder boards are boards you have to solder components onto (per the name). These are most of your standard circuit boards, and if you flip one over, you'll notice that all of the connections are soldered to the board itself.

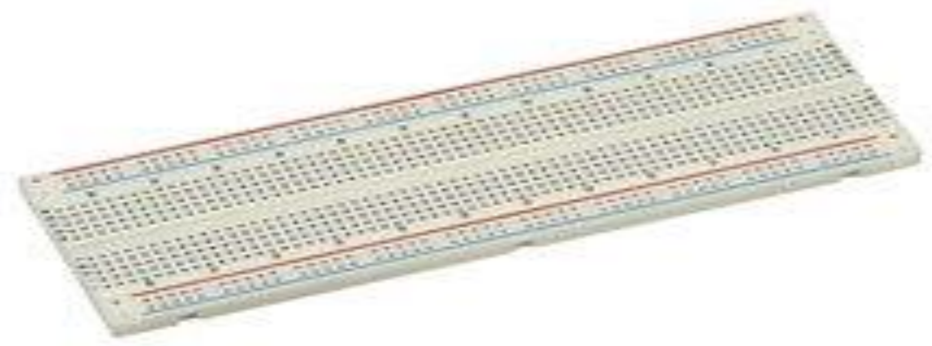


Fig 5.8 Block Diagram of Breadboard

CHAPTER-6

SOFTWARE REQUIREMENTS

6.1 ARDUINO IDE:

The Arduino Integrated Development Environment or Arduino software (IDE) contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them.

6.1.1 WRITING SKETCH

Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension. Ion. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

 - *Verify* Checks your code for errors compiling it.

 - *Upload* Compiles your code and uploads it to the configured board.

 - *New* Creates a new sketch.

 - *Open* Presents a menu of all the sketches in your sketchbook. Clicking one will open it within the current window overwriting its content.

 - *Save* Saves your sketch.

 - *Serial Monitor* Opens the serial monitor.

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

6.1.2 SKETCHBOOK

The Arduino Software (IDE) uses the concept of a sketchbook: a standard place to store your programs (or sketches). The sketches in your sketchbook can be opened from the **File > Sketchbook** menu or from the **Open** button on the toolbar. The first time you run the Arduino software, it will automatically create a directory for your sketchbook. You can view or change the location of the sketchbook location from with the **Preferences** dialog.

6.1.3 TABS, MULTIPLE FILES, AND COMPILATION

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

Before compiling the sketch, all the normal Arduino code files of the sketch (.ino, .pde) are concatenated into a single file following the order the tabs are shown in. The other file types are left as is.

6.1.4 UPLOADING

Before uploading your sketch, you need to select the correct items from the **Tools > Board** and **Tools > Port** menus. The boards are described below. On the Mac, the serial port is probably something like **/dev/tty.usbmodem241** (for an UNO or Mega2560 or Leonardo) or **/dev/tty.usbserial-1B1**, or **/dev/tty.USA19QW1b1P1.1**. On Windows, it's probably **COM1** or **COM2** (for a serial board) or **COM4**, **COM5**, **COM7**, or higher (for a USB board) - to find out, you look for USB serial device in the ports section of the Windows Device Manager. On Linux, it should be **/dev/ttyACMx** , **/dev/ttyUSBx** or similar. Once you've selected the correct serial port and board, press the upload button in the toolbar or select the **Upload** item from the **Sketch** menu. Current Arduino boards will reset automatically and begin the upload. With older boards (pre-Diecimila) that lack auto-reset, you'll need to press the reset button on the board just before starting the upload. On most boards, you'll see the RX and TX LEDs blink as the sketch is uploaded. The Arduino Software (IDE) will display a message when the upload is complete, or show an error.

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

6.1.5 LIBRARIES

Libraries provide extra functionality for use in sketches, e.g., working with hardware or manipulating data. To use a library in a sketch, select it from the **Sketch > Import Library** menu. This will insert one or more **#include** statements at the top of the sketch and compile the library with your sketch. Because libraries are uploaded to the board with your sketch, they increase the amount of space it takes up. If a sketch no longer needs a library, simply delete its **#include** statements from the top of your code.

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

6.1.6 THIRD-PARTY HARDWARE

Support for third-party hardware can be added to the **hardware** directory of your sketchbook directory. Platforms installed there may include board definitions (which appear in the board menu), core libraries, bootloaders, and programmer definitions. To install, create the **hardware** directory, then unzip the third-party platform into its own sub-directory. (Don't use "Arduino" as the sub-directory name or you'll override the built-in Arduino platform.) To uninstall, simply delete its directory.

6.1.7 SERIAL MONITOR

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

6.1.8 LANGUAGE SUPPORT

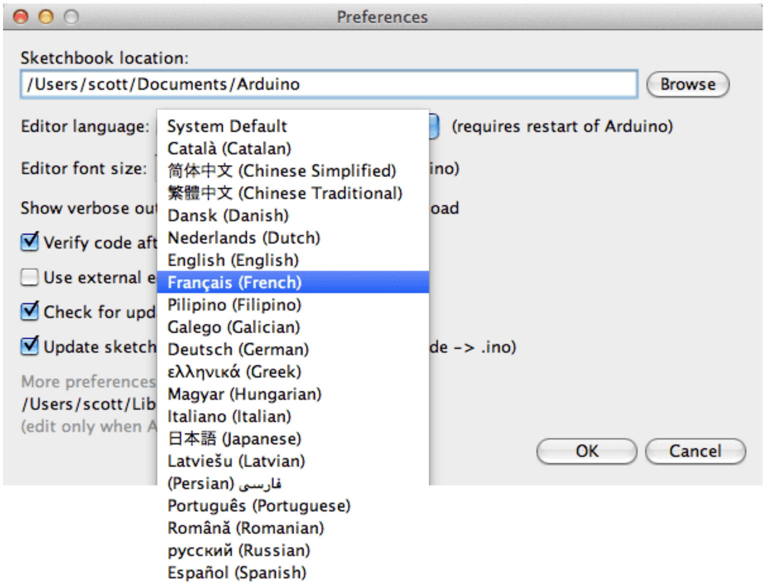


Fig 6.1 Block Diagram

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

If you would like to change the language manually, start the Arduino Software (IDE) and open the **Preferences** window. Next to the **Editor Language** there is a dropdown menu of currently supported languages. Select your preferred language from the menu, and restart the software to use the selected language. If your operating system language is not supported, the Arduino Software (IDE) will default to English.

You can return the software to its default setting of selecting its language based on your operating system by selecting **System Default** from the **Editor Language** drop-down. This setting will take effect when you restart the Arduino Software (IDE). Similarly, after changing your operating system's settings, you must restart the Arduino Software (IDE) to update it to the new default language.

6.2 THINGSPEAK

ThingSpeak is an IoT analytics platform service that allows you to aggregate, visualize and analyse live data streams in the cloud. ThingSpeak provides instant visualizations of data posted by your devices to ThingSpeak. With the ability to execute MATLAB code in ThingSpeak you can perform online analysis and processing of the data as it comes in. ThingSpeak is often used for prototyping and proof of concept IoT systems that require analytics.

6.2.1 INTRODUCTION

Internet of Things (IoT) describes an emerging trend where a large number of embedded devices (things) are connected to the Internet. These connected devices communicate with people and other things and often provide sensor data to cloud storage and cloud computing resources where the data is processed and analysed to gain important insights. Cheap cloud computing power and increased device connectivity is enabling this trend.

IoT solutions are built for many vertical applications such as environmental monitoring and control, health monitoring, vehicle fleet monitoring, industrial monitoring and control, and home automation. At a high level, many IoT systems can be described using the diagram below:

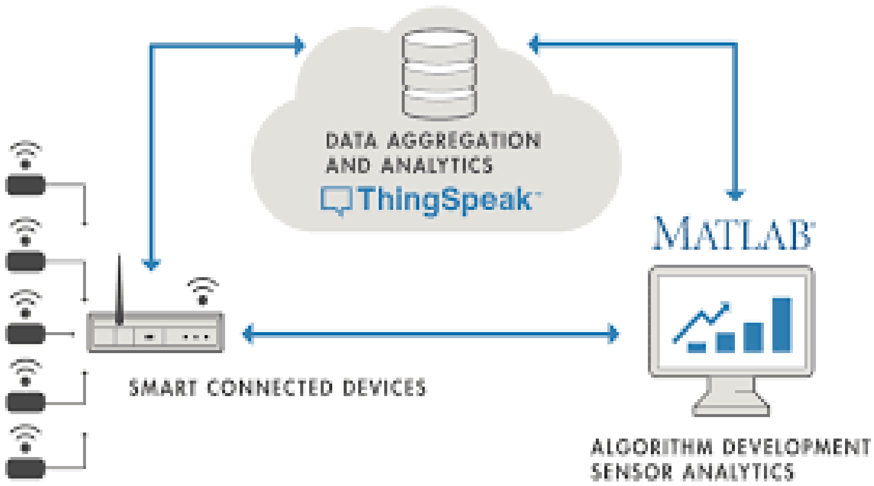


Fig 6.2 Block Diagram of ThingSpeak

On the left, we have the smart devices (the "things" in IoT) that live at the edge of the network. These devices collect data and include things like wearable devices, wireless temperatures sensors, heart rate monitors, and hydraulic pressure sensors, and machines on the factory floor.

In the middle, we have the cloud where data from many sources is aggregated and analysed in real time, often by an IoT analytics platform designed for this purpose.

The right side of the diagram depicts the algorithm development associated with the IoT application. Here an engineer or data scientist tries to gain insight into the collected data by performing historical analysis on the data. In this case, the data is pulled from the IoT platform into a desktop software environment to enable the engineer or scientist to prototype algorithms that may eventually execute in the cloud or on the smart device itself.

An IoT system includes all these elements. ThingSpeak fits in the cloud part of the diagram and provides a platform to quickly collect and analyse data from internet connected sensors.

6.2.2 THINGSPEAK KEY FEATURES

ThingSpeak allows you to aggregate, visualize and analyse live data streams in the cloud. Some of the key capabilities of ThingSpeak include the ability to:

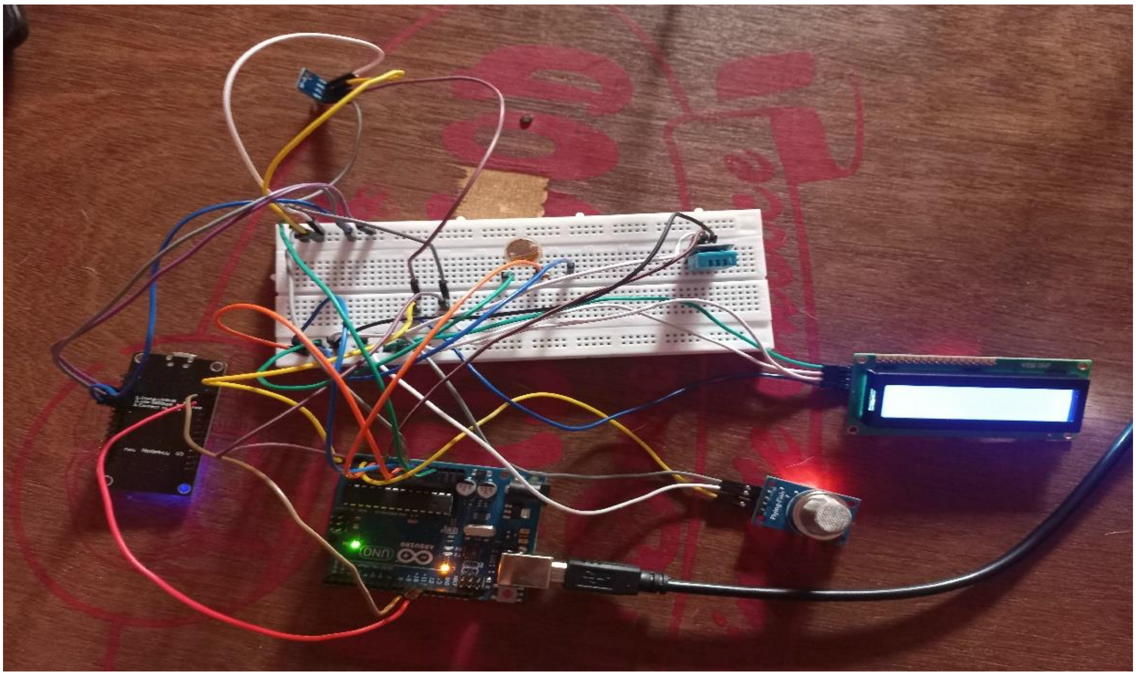
- Easily configure devices to send data to ThingSpeak using popular IoT protocols.
- Visualize your sensor data in real-time.
- Aggregate data on-demand from third-party sources.
- Use the power of MATLAB to make sense of your IoT data.
- Run your IoT analytics automatically based on schedules or events.
- Prototype and build IoT systems without setting up servers or developing web software.

CHAPTER-7

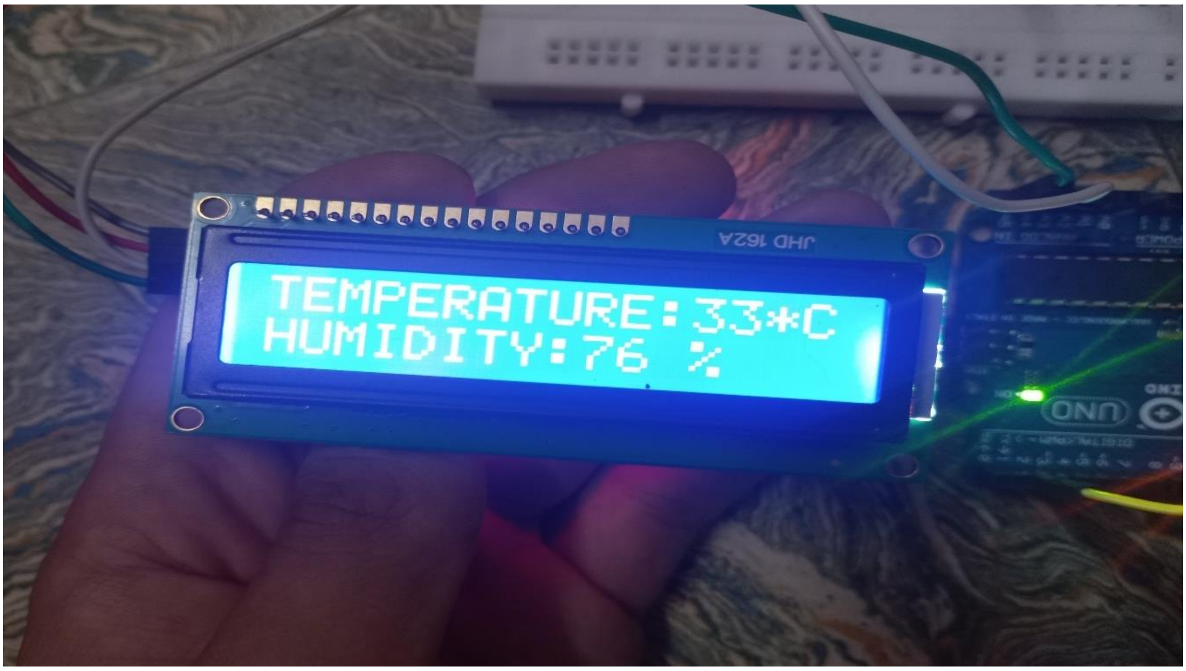
RESULT

7.1 OUTPUT FOR ARDUINO

7.1.1 OVERALL CIRCUIT

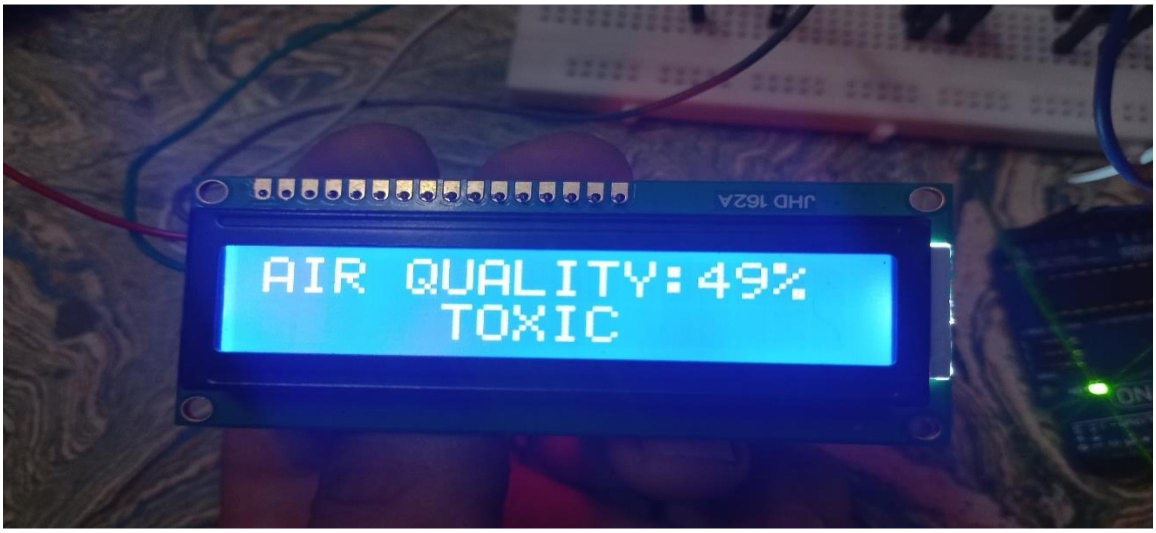


7.1.2 TEMPERATURE AND HUMIDITY



7.1.3 BMP180 BAROMETRIC SENSOR

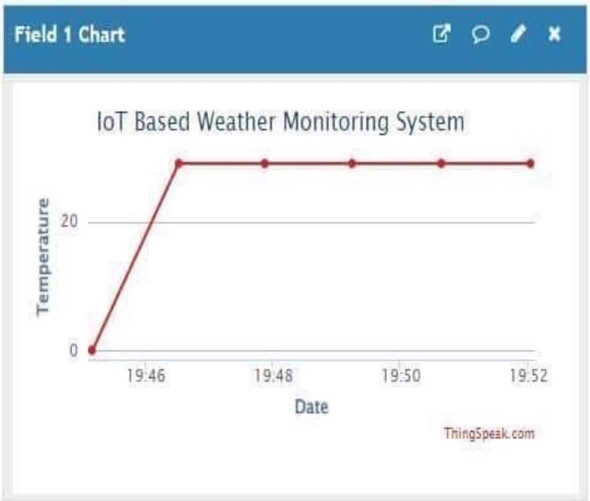
7.1.4 MQ-135 AIR QUALITY SENSOR

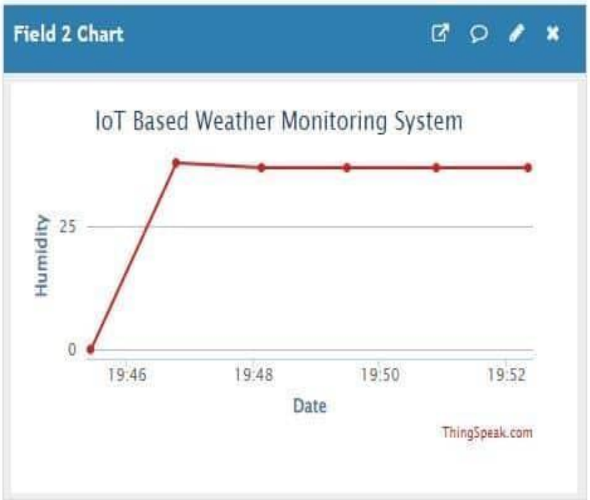


7.1.5 LIGHT DEPENDENT RESISTOR-LDR

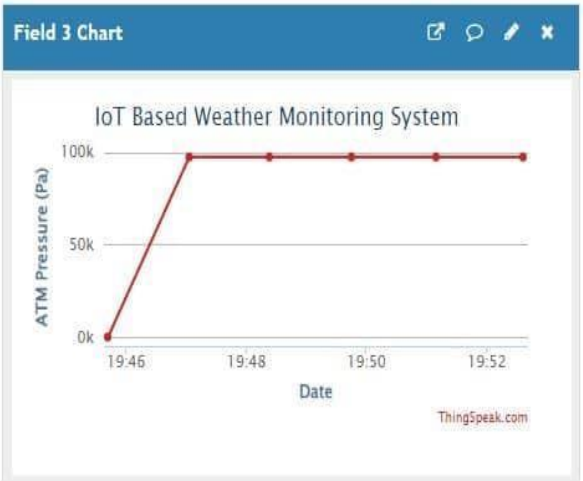
7.2 OUTPUT FOR THINGSPEAK

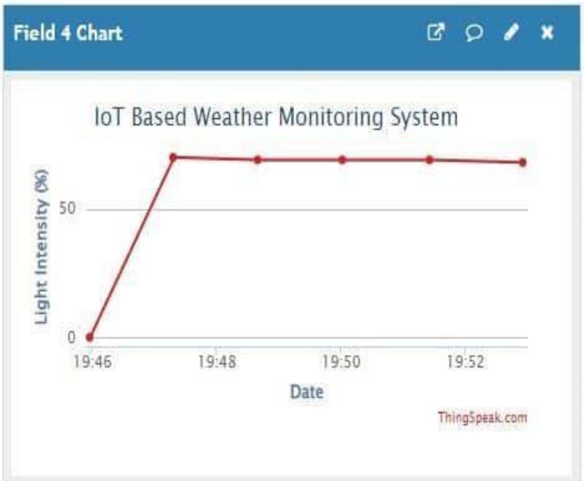
7.2.1 TEMPERATURE AND HUMIDITY



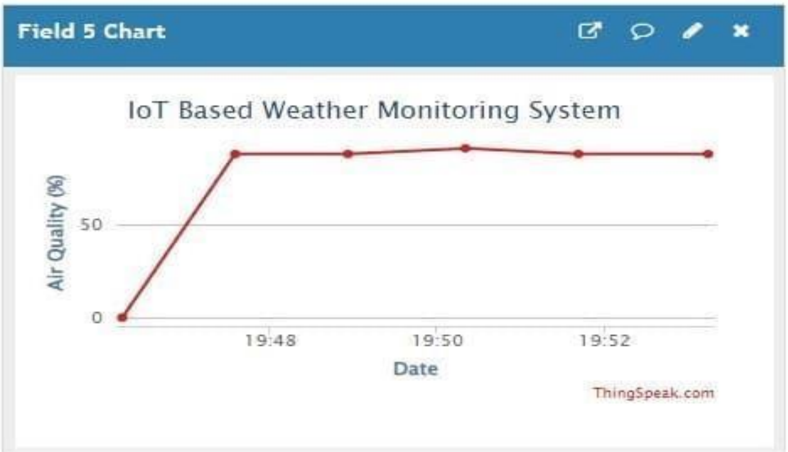


7.2.2 BMP180 SENSOR AND LDR





7.2.3 MQ-135 SENSOR



CHAPTER-8

CONCLUSION AND FUTURE ENHANCEMENT

8.1 CONCLUSION

The implementation of a system for monitoring environmental parameters using the IoT has been tentatively tested to verify weather parameters. The system provides low power consumption solution for the establishment of a station weather system. The system tested in an indoor environment and it successfully updated the environment and weather conditions from sensor data. This information will be useful for future review and tend to be shared effectively with various users. this model can be extended to the observation of contamination in new and modern urban areas. To protect the general well-being from contamination, this model provides an effective and minimal effort response for continuous observation.

8.2 FUTURE ENHANCEMENT

An alarm can be added to the circuit to notify the user in case of excess smoke conditions i.e., Smoke alarm.

An SMS can be sent to clients notifying them with the temperature/humidity/smoke parameters.

CHAPTER-9

BIBLIOGRAPHY

1. E. Welbourne, L. Battle, G. Cole, K. Gould, K. Rector, S. Raymer et al., Building the internet of things using RFID: The RFID experience, IEEE internet comput., vol. 13, no. 3, pp.48- 55, May- Jun. 2009.

2. Shifeng Fang; Li Da Xu; Yunqiang Zhu; JiaerhengAhati; Huan Pei; Jianwu Yan; Zhihui Liu., An integrated system for regional environmental monitoring and management based on internet of things, IEEE Transactions on Industrial Informatics, vol.10, no. 2,pp.1596-1605, May-Jun. 2014.

3. J. A. Stankovic, Research directions for the Internet of Things,IEEE Internet Things., vol. 1, no. 1, pp. 39, Feb. 2014

4. Shanzhi Chen; HuiXu; Dake Liu; Bo Hu; Hucheng Wang.

5. L. Atzori, A. Iera, and G. Morabito, The internet of things: A survey, Comput. Netw., vol. 54, no. 15, pp. 27872805, 2010

6. P. Bellavista, G. Cardone, A. Corradi, and L. Foschini,Convergence of MANET and WSN in IoT urban scenarios,IEEE Sens. J., vol. 13, no. 10, pp. 35583567, Oct. 2013.

7. BulipeSrinivasRao , Prof. Dr. K. SrinivasaRao , Mr. N. Ome, Internet of Things (IOT) Based Weather Monitoring system, IJARCCE Journal,vol. 5, no. 9, sept. 2016.