CHAPTER 1
INTRODUCTION

A lightning strike or lightning bolt is an electric discharge between the atmosphere and an Earth-bound object. They mostly originate in a cumulonimbus cloud and terminate on the ground, called cloud to ground (CG) lightning. A less common type of strike, called cloud to cloud (CC), is upward propagating lightning initiated from a tall grounded object and reaches into the clouds. About 25% of all lightning events worldwide are strikes between the atmosphere and earth-bound objects. The bulk of lightning events are intra-cloud (IC) or cloud to cloud (CC), where discharges only occur high in the atmosphere.

A single lightning event is a "flash", which is a complex, multi-stage process, some parts of which are not fully understood. Most cloud to ground flashes only "strike" one physical location, referred to as a "termination". The primary conducting channel, the bright coursing light that may be seen and is called a "strike", is only about one inch in diameter, but because of its extreme brilliance, it often looks much larger to the human eye and in photographs. Lightning discharges are typically miles long, but certain types of horizontal discharges can be upwards of tens of miles in length

Figure 1: Lightning storm

The entire flash lasts only a fraction of a second. Most of the early formative and propagation stages are much dimmer and not visible to the human eye. It is rather easy to obtain traces of lightning events, through burnt trees, far away thunder or

static in an FM radio. But it's much more difficult to actually measure all the parameters of all the flashes. The main reasons for this are:

- Lightning flash are discrete and rare events (typ. 1 flash per km² per year in France)
- Lightning strikes in different place. There is a popular saying that "lightning never strikes the same place twice", not to be trusted
- Lightning flashes occurs at random moments.
- Each lightning flash is a different event as can be seen in high speed videos, with a different path of current in the air, various attachment point and different time parameters.

1.1 Strikes and Injuries:

Lightning strikes can produce severe injuries, and have a mortality rate of between 10% and 30%, with up to 80% of survivors sustaining long-term injuries. These severe injuries are not usually caused by thermal burns, since the current is too brief to greatly heat up tissues; instead, nerves and muscles may be directly damaged by the high voltage producing holes in their cell membranes, a process called electro operation. In a direct strike, the electrical currents in the flash channel pass directly through the victim. The relatively high voltage drop around poorer electrical conductors (such as a human being), causes the surrounding air to ionize and break down, and the external flashover diverts most of the main discharge current so that it passes "around" the body, reducing injury.

Metallic objects in contact with the skin may "concentrate" the lightning's energy, given it is a better natural conductor and the preferred pathway, resulting in more serious injuries, such as burns from molten or evaporating metal. At least two cases have been reported where a strike victim wearing an iPod suffered more serious injuries as a result.

However, during a flash, the current flowing through the channel and around the body will generate large electromagnetic fields and EMPs, which may induce electrical transients (surges) within the nervous system or pacemaker of the heart, upsetting normal operations. This effect might explain cases where cardiac arrest or seizures followed a lightning strike that produced no external injuries. It

may also point to the victim not being directly struck at all, but just being very close to the strike termination. Another effect of lightning on bystanders is to their hearing.

The resulting shock wave of thunder can damage the ears. Also, electrical interference to telephones or headphones May result in damaging acoustic noise. In addition to electrical wiring damage, the other types of possible damage to consider include structural, fire, and property damage. This problems can be overcome by providing prior intimation about lightning strikes with the help of lightning detection system. This system not only provides lightning alerts but also

Figure 1.1: lightning occurs at high altitude range

weather conditions, which will be helpful for farmers in agriculture field. We know that farmer is a person engaged in agriculture, raising living organism for food or raw material. Every year, lots of loss has been taking place in agriculture to provide solution for this, information regarding weather has been displayed to that particular place people, So that further steps can be taken by people.

CHAPTER 2

LITERATURE REVIEW

2.1 Strike alert personal lightning detector :

The Strike Alert Personal Lightning Detector is an absolute necessity for single person. if he/she spend lots of time at the outdoor pool, ball field, golf course, soccer field or anywhere else that a lightning warning might be useful. It's not just a clever and compact outdoor accessory, it can literally act as a lifesaving device. This device is of palm size and can detect lightning strikes within 40miles.

Figure 2.1: Strike Alert Device.

- Device cost is of $76.07.
- Only for single user.
- Push button need to press to know the status of lightning strike.
- This system will not provide information regarding weather.
- Exact location we can’t predict.

2.2 Sky Scanp5-3 :

This system allow us to know the level of the storm. It determines if

it's moving towards away, parallel or towards our position. Though it has effective features.

Figure 2.2: Sky scanp5-3

Following are some disadvantages:

- Expensive.
- Single user can use this device.
- It shows information as a distance but not the exact location.

Though many devices are present, among those many are for single users. Always user need to check the status by using push button. Almost all devices are expensive as like above one. In order to provide solution for this, lightning detection system has been designed.

CHAPTER 3

EXISTINGAND PROPOSED SYSTEM

3.1 Existing System:

3.1.1 Vajrapaat Application:

Andhra Pradesh is coming up with an alerting system that will warn against lightning -- the first of its kind in the country. A team of five girl students of Kuppam Engineering College are working on the development of the app which alerts people about impending lightning strikes. The project is taken up by AP State Council of Higher Education (APSCHE) with ISRO. The developed app is expected to be launched in the next 10-15 days and likely to be named as Vajrapaat.

Figure 3.1.1(a): Vajrapaat app

The Andhra Pradesh State Council of Higher Education (APSCHE) has tied up with the Indian Space Research Organization (ISRO) to share satellite services

required for weather forecasts, which will be used to monitor and alert against possible lightning strikes.

At present three sensors have been set up in the State at Anantapur, Kuppam and Visakhapatnam to test the lightning alert system. To add to these, eight more sensors are being shipped and may be set up in the next 10 days. Each sensor tower has a range of 200 km, which has been installed in a zigzag pattern. The partnering organization, ISRO has also successfully roped in US-based Company Earth Networks to analyze the electromagnetic waves to spot the exact location of the lightning.

Figure 3.1.1(b): Lightning Sensor

"proximity sensors assess electromagnetic waves and send data to the cloud network of Earth Networks, which is then analyses and determines details of the lightning's latitude, longitude and directions of travel and time of strike. The processed data will then be sent through Application Program Interface.

3.1.1.1 Drawbacks:

- This system is more expensive
- More time consuming process
- Internet required
- Less accurate

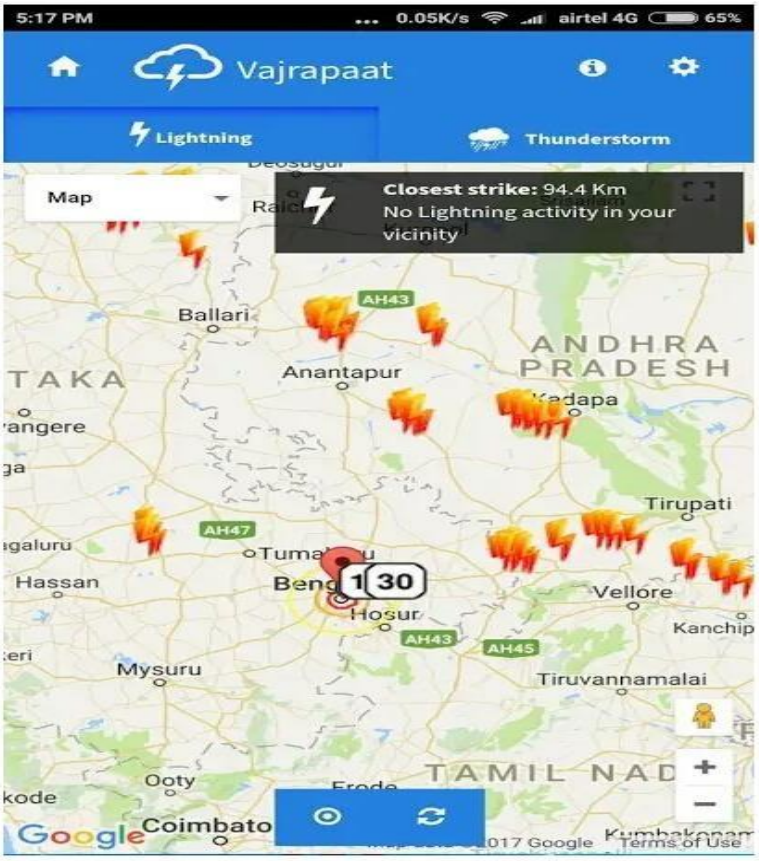

Figure 3.1.1(c): app image

3.2 PROPOSED SYSTEM:

In Proposed system, it provides lightning strikes alerts in advance. Not only alerts but also provides weather information like temperature, pressure, humidity with the help of sensor Technology. In order to help the farmers. We know that when rainy, weather condition will changes automatically like maximum humidity, low pressure and minimum temperature. So if we provide this information in advance there may possible to take further steps to protect themselves.

In lightning detection system AS3935 IC plays a major role. The AS3935 is a programmable fully integrated lightning sensor IC that detects the presence and approach of potentially hazardous lightning activity in the vicinity and provides an estimation on the distance to the head of the storm. Weather information will be providing to the people continuously in a display as well as its uploaded to an IoT platform like Thing speak, so that weather information can be access from anywhere and also lightning detection process will be continuous and has been

shown through website. Through website we can able to know the location where lightning strike is going to takes place before a couple of minutes.

- This system is accurate
- Its economic friendly
- Up to 40km range it can able to detect lightning strikes
- Analysis is simple
- weather information has been provided
- Its an effective system

CHAPTER – 4
HARDWARE AND SOFTWARE REQUIREMENTS

The Proposed system consist of sensing unit such as DHT22 to measure temperature and humidity, MQ135 to measure air quality, etc are given below list

4.1 Hardware Requirements:

4.1.1 Power Supply:

A power supply is an electronic device that supplies electric energy to an electrical load. The primary function of a power supply is to convert one form of electrical energy to another and, as a result, power supplies are sometimes referred to as electric power converters. Some power supplies are discrete, stand-alone devices, whereas others are built into larger devices along with their loads. Examples of the latter include power supplies found in desktop computers and consumer electronics devices. Every power supply must obtain the energy it supplies to its load, as well as any energy it consumes while performing that task, from an energy source. Depending on its design, a power supply may obtain energy from various types of energy sources:

- Energy storage devices such as batteries and fuel cells
- Electromechanical systems such as generators and alternators
- Solar power convertors or another power supply.

Batteries are common DC voltage source for electronic equipment especially portables like cell phones and iPods. Most non-portable equipment uses power supplies that operate from the AC power line but produce one or more outputs.

4.1.1.1 DC Power Supply:

A DC power supply is one that supplies a voltage of fixed polarity (either positive or negative) to its load. Depending on its design, a DC power supply may be powered from a DC source or from an AC source such as the power mains.

4.1.1.2 Power Supply Characteristics:

- The input is 230volt 50Hz Alternative Current(AC).
- A power supply unit converts the AC into DC and provides one or more DC output voltages.
- Some modern electronic circuits need two or more different voltages.
- Commonly required DC voltages are 48, 24, 15, 12, 9, 5, 3.3, 2.5, 1.8, 1.5, 1.2 and 1volt.

4.1.1.3 Components of Power Supply:

The major components of power supply unit are:

- Ac source
- Transformer
- Rectifier
- Filter
- Voltage Regulator

4.1.1.4 Power Supply:

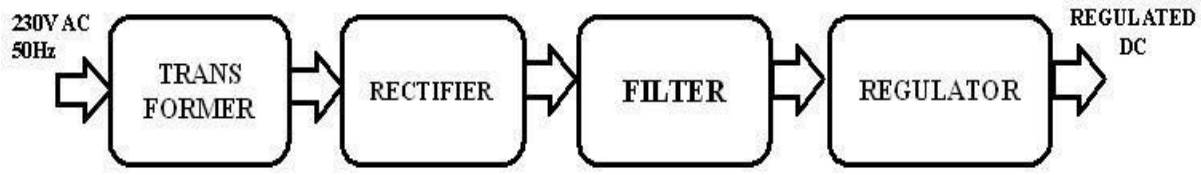

Figure 4.1.1.4: Block diagram of Power Supply

4.1.1.5 Transformer:

Transformer is a static device which transforms electrical energy from one circuit to another without any direct electrical connection with the help of mutual induction. It transforms power from one to other circuit without change in frequency but there may be change in voltage levels .It depends upon faradays law of electromagnetic mutual induction .Rate of change of flux linkage w.r.t time is directly proportional to the induced emf in the conductor or coil.

4.1.1.6 Rectifier:

A rectifier is a circuit that is used for converting AC supply into unidirectional DC supply. This rectification is achieved by using passive components such as diodes. I While choosing the diodes with PIV rating is taken into consideration.

A bridge rectifier makes use of four diodes in a bridge arrangement to achieve full-wave rectification. The AC voltage generated is passed through a circuit of four diodes arranged as shown above and emerged converted into a more useful DC output.

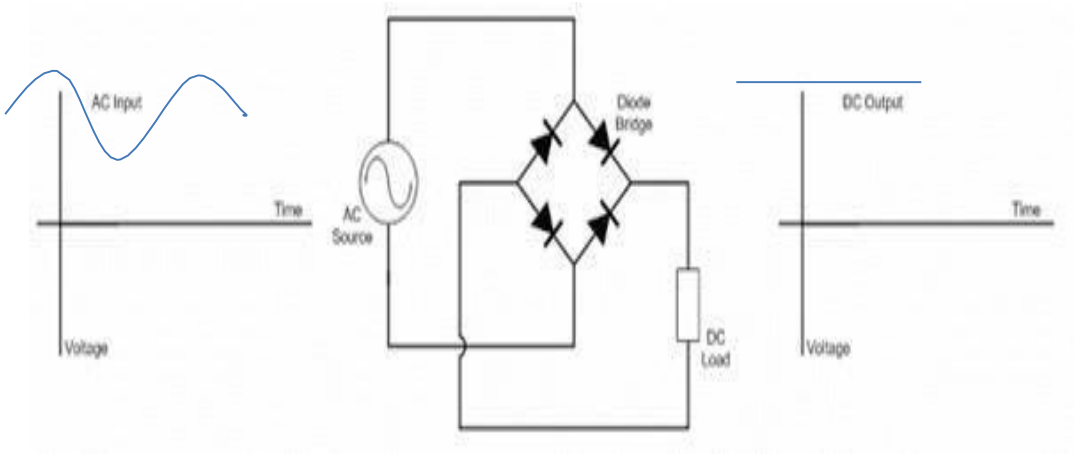

Figure 4.1.1.6: Bridge Rectifier

4 .1.1.7 Filters:

But this rectified output contains some percentage of superimposed a.c. ripples. So to filter these A.C components filter stage is built around the rectifier stage. The cheap, reliable, simple and effective filtering for low current drawing loads (say up to 50 mA) is done by using shunt capacitors. This electrolytic capacitor has polarities, take care while connecting the circuit.

Filtering is technique in which a pure DC is obtained. The output of a bridge rectifier is not pure DC and needs to be converted into pure form. Any given filter involve capacitors, inductors, &resistors in some combination.

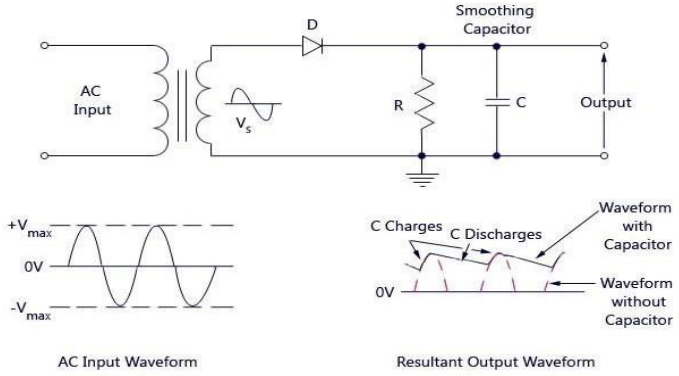

Figure 4.1.1.7: Filter circuit

4.1.2 Raspberry pi:

The Raspberry Pi is a credit card sized computer. It's basically a small PC which provides all the basic functions that are provided by a desktop PC. For example, it provides functions like word processing, gaming and playing audio/video. In this project we are using Raspberry Pi2 model B which is successor of Raspberry pi. The RaspberryPi2 model B is shown in the figure 4.1.2.1.

4.1.2.1Technical specifications:

The following are the technical specifications for Raspberry Pi2 model B.

- Processor: Broadcom BCM2836
- CPU Cores: 4 xCortex-A7
- CPU Clock: 900 MHz
- GPU: Broadcom Video CoreIV
- Built in RAM: 1 GB
- Ethernet: 100 Mbit
- USB: 4 x3.1host
- Display : HDMI, Composite
- Expansion: 40 pin Multi –I/O
- Other features: Camera input

Figure4.1.2.1: Raspberry pi 0

The software's offered are RASPBIAN, PIDORA, OPENELEC, RASPBMC, RISC OS and ARCH LINUX. All these software's can be downloaded easily and for free from the official forum under the NOOBS (new out of the box software) category.

It provides support for functioning and coding in Python as the main programming language. It also provides support for BASIC, C, C++, JAVA, Perl and Ruby.

4.1.2.2 Booting Process:

The booting method involves the following steps: Download NOOBS operating system install manager

1. Format a micro SD card.
2. Burn the NOOBS image onto a micro SD card.
3. Insert the card into the micro SD card slot on the RaspberryPi board.
4. Plug in keyboard, mouse and monitor cable onto the board and to the monitor.
5. Plug in the USB power cable.
6. The boot process has now begun and a configuration window appears to enable the camera module if present and setting the date and time.
7. The command line interface loads up asking for the username and

password submitting successful information the board is fully operational.

8. The graphical user interface can be chosen by typing start.
9. Default username and passwords for first boot are: username: pi, password: raspberry.
10. After the booting process the board can be utilized for any projects.

4.1.2.3 GPIO Pin Description:

One great feature of the RaspberryPi is the row of GPIO (general purpose input/output) pins. These pins are used to interface the RaspberryPi and the outside world. Seventeen of the 26 pins are GPIO pins; the others are power or ground pins. All these pins are programmable which are used to interact with their world. The GPIO pins are described using below figure 4.1.2.3.

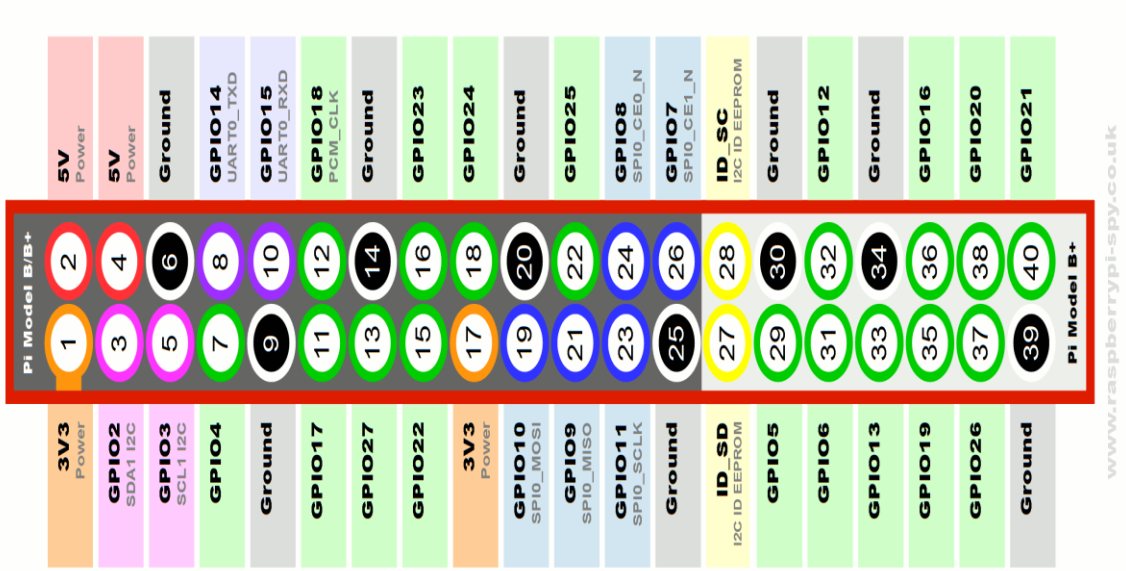

Figure 4.1.2.3: GPIO Pin description

4.1.3 Franklin Lightning Sensor(AS3935):

AS3935 Franklin Lightning Sensor IC is a programmable fully integrated lightning sensor that detects the presence and approach of potentially hazardous lightning activity in the vicinity and provides an estimation on the distance to the head of the storm. The embedded lightning algorithm checks the incoming signal pattern to

reject the potential man-made disturbers. The AS3935 can also provide information on the noise level and inform the external unit (e.g. microcontroller) in case of high noise conditions, with the noise floor generator and noise floor evaluation blocks.

The AS3935 can be programmed via a 4-wire standard SPI or an I²C. Also if the latter is chosen it is possible to choose among four different addresses. Two clocks are internally generated by two different RC-Oscillators: TRCO and SRCO. An automatic procedure can increase the precision of those oscillators. The AS3935 can be either supplied by an internal voltage regulator or directly by VDD

4.1.3.1 Features:

- Lightning sensor warns of lightning storm activity within a radius of 40km.
- Distance estimation to the head of the storm down to 1km in 14 steps.
- Detects both cloud-to-ground and intra-cloud (cloud-to-cloud) flashes.
- Embedded man-made disturber rejection algorithm.
- Programmable detection levels enable threshold setting for optimal controls.
- SPI and I²C interface is used for control and register reading.
- Antenna Tuning to compensate variations of the external components.
- Supply voltage range: 2.4V to 5.5V.
- Power-down, listening, and active mode.
- Package: 16LD MLPQ (4x4mm).

Figure 4.1.3.1(a): AS3935 Lightning Sensor

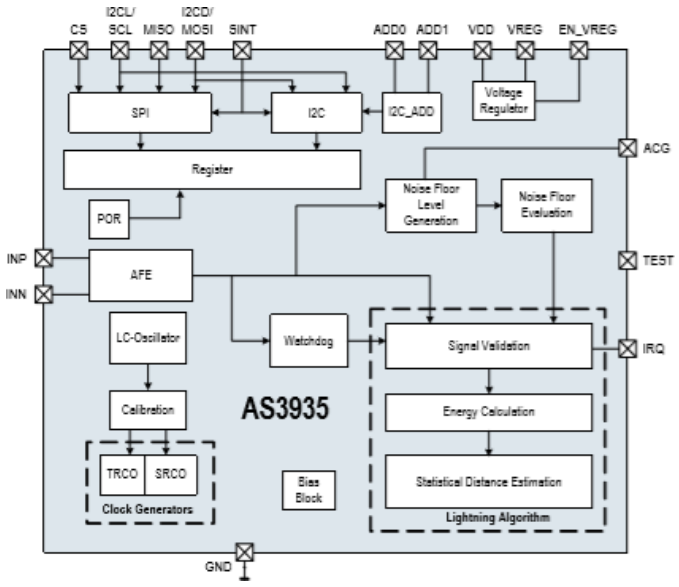

Fig4.1.3.1 (b):AS3935 internal circuit

4.1.3.2 Working Principle:

AS3935 sensor consist of antenna front end, that extracts the electromagnetic waves And that information will be given to lightning algorithm Which detects weather the signal is man-made signal, noise signal and lightning signal. If the signal is noise then it rejects .If the signal is other than noise and man-made then it sends as lightning alert. It has a pair of clock generators which compares a fixed values with a current values .Based on that it decides what type of signal has been generated.

4.1.3.3 Pin Description:

Pin description of the AS3935 lightning sensor as follows in the below figure 4.1.3.3(a):

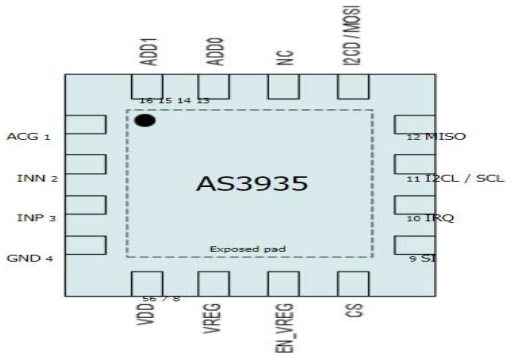

Figure 4.1.3.3(a): Pin Description

Pin Number	Pin Name	Pin Type	Description
1	AGG	Analog I/O	Ac-ground
2	INN	Analog I/O	Antenna ground
3	INP	Analog I/O	Antenna positive output
4	GND	Supply Pad	Ground
5	VDD	Supply Pad	Positive supply voltage
6	VREG	Supply Pad	Positive supply voltage/Regulated voltage
7	EN-VREG	Digital Input	Voltage regulator enable
8	CS	Digital Input	Chip select(active low)
9	SI	Digital Input	Select interface
10	IRQ	Digital output	Interface
11	I2C/SCL	Digital Input	I²C or SPI clock bus
12	MISO	Digital output	SPI data output bus
13	I2CD/MOSI	Digital I/O with Pull up/Digital input	I²C data bus or SPI data input bus
14	NC		Not connected
15	ADDO	Digital Input	I²C address selection LSB
16	ADD1	Digital Input	I²C address selection MSB
Exposed data		Supply pad	Connect to ground

Table 4.1.3.3(b): Pin Description of AS3935 IC

4.1.3.4 Applications:

- Weather stations
- Clocks
- Sports equipment
- Portables
- Pool safety
- Uninterruptible Power Supply (UPS)

- Global Positioning System (GPS)
- Cellular phones
- Watches
- Golf equipment

4.1.4 Temperature and Humidity sensor (DHT22):

4.1.4.1 Introduction:

TheDHT22isadigitalrelativehumidityandtemperaturesensor.TheoutputofDHT22 is a calibrated digital signal. The DHT22 consists of one thermistor and one capacitive humidity sensor. The thermistor is used to measure temperature and capacitive humidity sensor will measure relative humidity levels in percentage. It will give accurate output readings and also can measure high temperature and humidity levels.

4.1.4.2 Working principle:

DHT22 consist of a humidity sensing component, a NTC temperature sensor (thermistor) and an IC on the back side of the sensor. For measuring humidity they use the humidity sensing component which has two electrodes with moisture holding substrate between them. So as the humidity change, the conductivity of the substrate changes or the resistance between these electrodes changes. This change in

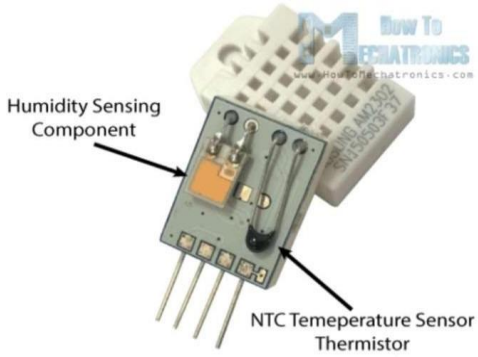

Figure 4.1.4.2(a): DHT22 Sensor

resistance is measured and processes by the IC which makes it ready to be read by a microcontroller.

On the other hand, for measuring temperature these sensors use a NTC temperature sensor or a thermistor. A thermistor is actually a variable resistor that changes its resistance with change of the temperature.

These sensors are made by sintering of semi conductive materials such as ceramics or polymers in order to provide larger changes in the resistance with just small changes in

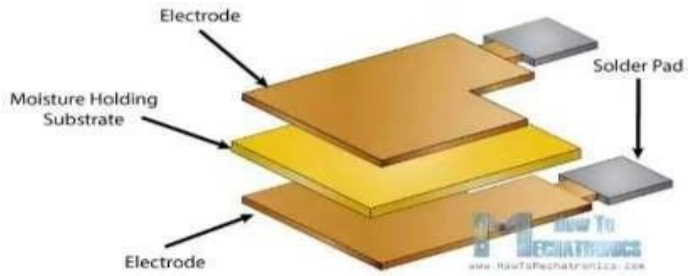

Figure 4.1.4.2(b): Semiconductor materials view

temperature. The term "NTC" means "Negative Temperature Coefficient", which means that the resistance decreases with increase of the temperature.

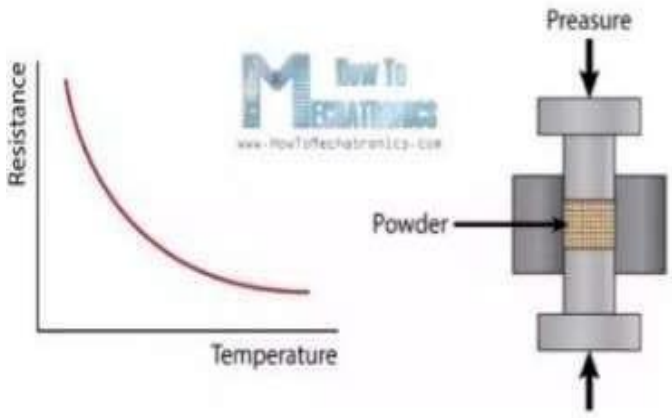

Figure 4.1.4.2(c): Graphical representation.

4.1.4.3 Features:

- Operating voltage: 3.3 -5.5vdc
- Sensing element: Polymer humidity capacitor
- Operating range: Humidity 0 – 100 % RH & Temperature: -40^0 C to 80^oC

- High precision & High stability
- Calibrated digital signal
- Capacitive type
- Fully inter change able pins
- Low power consumption
- Long transmission distance
- The below figure 3.17 shows the DHT22diagram

4.1.4.4 Pin Description:

S.NO	PIN Description
1	VCC
2	DATA
3	No Connection
4	Ground

Table 4.1.4.4: Pin description of DHT22

4.1.4.5 Calculationofhumidityandtemperatureusingdht22:

For example, MCU has received 40 bits data from DHT22 as

0000 0010 1011 1010	0000 0001 0011 0010	1110 1111
RH data	Temperature data	Check sum

If processor sends are quest signal, thenDHT22 will acknowledge a 40bit data which includes Humidity & Temperature data.

$(0000\ 001010111010)_2 = (698)_{10}$

Therefore, RH = 698 /10 = 69.8% RH

$(0000\ 000100110010)_2 = (306)_{10}$

Therefore, Temperature = 306/10 = 30.6^{o} C

Sum = 0000 0010 + 1011 1010 + 0000 0001 + 0011 0010 = 1110 1111

Check sum= last eight digits of sum = 1110 1111.

4.1.5 MQ-135 GAS SENSOR:

4.1.4.2 Introduction:

The air quality sensor is also a MQ-135 sensor for detecting venomous gases that are present in the air in homes and offices. The gas sensor layer of the sensor unit is made up of tin dioxide (SnO2); it has lower conductivity compare to clean hair and due to air pollution the conductivity is increases. The air quality sensor detects ammonia, nitrogen oxide, smoke, CO2 and other harmful gases. The air quality sensor has a small potentiometer that permits the adjustment of the load resistance of the sensor circuit. The 5V power supply is used for air quality sensor.

The air quality sensor is a signal output indicator instruction. It has two outputs: analog output and TTL output. The TTL output is low signal light which can be accessed through the IO ports on the Microcontroller. The analog output is an concentration, i.e. increasing voltage is directly proportional to increasing concentration. This sensor has a long life and reliable stability as well.

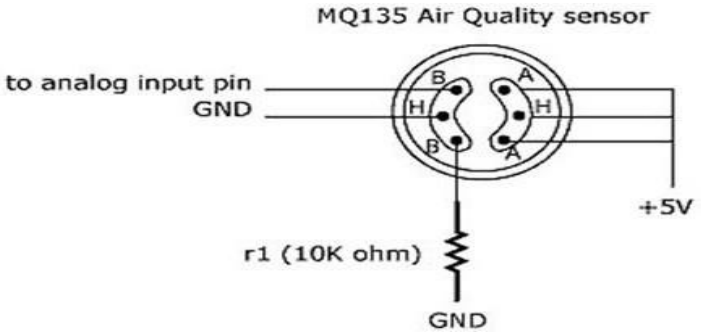

Figure 4.1.5.1: MQ 135 Gas Sensor

4.1.5.2 Characteristics of MQ 135:

- Good sensitivity to harmful gases in wide range.
- It has long life and low cost.
- Possesses high sensitivity to ammonia, benzene, sulfide gases.
- It is a simple drive circuit

4.1.5.3 Applications:

- Air quality monitor
- Detection of harmful gases
- Domestic air pollution detection
- Industrial pollution detection
- Portable air pollution detection

4.1.5.4 Pin description:

Arduino	G
5V	V
GND	G
NC	D
Analog A0	AO

Table 4.1.5.4: Pin description of Gas sensor

4.1.6 LCD DISPLAY:

A liquid-crystal display (LCD) is a flat-panel display or other electronically modulated optical device that uses the light-modulating properties of liquid crystals. Liquid crystals do not emit light directly, instead using a backlight or reflector to produce images in colour or monochrome. LCDs are available to display arbitrary images (as in a general- purpose computer display) or fixed images with low information content, which can be displayed or hidden, such as preset words, digits, and 7-segment displays, as in a digital clock. They use the same basic technology, except that arbitrary images are made up of a large number of small pixels, while other displays have larger elements.

LCDs are used in a wide range of applications including computer monitors, televisions, instrument panels, aircraft cockpit displays, and indoor and outdoor signage. Small LCD screens are common in portable consumer devices such as digital cameras, watches, calculators, and mobile telephones, including Smartphone's.

LCD screens are also used on consumer electronics products such as DVD players, video game devices and clocks. LCD screens have replaced heavy, bulky cathode ray tube (CRT) displays in nearly all applications. LCD screens are available in a wider range of screen sizes than CRT and plasma displays, with LCD screens available in sizes ranging from tiny digital watches to huge, big-screen television set.

Since LCD screens do not use phosphors, they do not suffer image burn-in when a static image is displayed on a screen for a long time (e.g., the table frame for an aircraft schedule on an indoor sign). LCDs are, however, susceptible to image persistence. The LCD screen is more energy-efficient and can be disposed of more safely than a CRT can.

Its low electrical power consumption enables it to be used in battery- powered electronic equipment more efficiently than CRTs can be. By 2008, annual sales of televisions with LCD screens exceeded sales of CRT units worldwide, and the CRT became obsolete for most purposes. An LCD display screen used as a notification panel for travellers.

Figure 4.1.6: LCD display

4.1.6.1 Working Principle:

The main principle behind liquid crystal molecules is that when an electric current is applied to them, they tend to untwist. This causes a change in the light angle passing through them. This causes a change in the angle of the top polarizing filter with respect to it. So little light is allowed to pass through

that particular area of LCD. Thus that area becomes darker comparing to others.

4.1.7 BMP085 Barometric and Pressure Sensor:

4.1.7.1 Introduction:

The BMP085 is a high precision, ultra-low power barometric pressure sensor for use in advance mobile applications. The BMP085 offers superior performance. At the same time the BMP085 features ultra-low power consumption of down to 3µA.The BMP085 sensor is based on piezo-resistive MEMS technology for EMC robustness, high accuracy and linearity as well as long term stability.

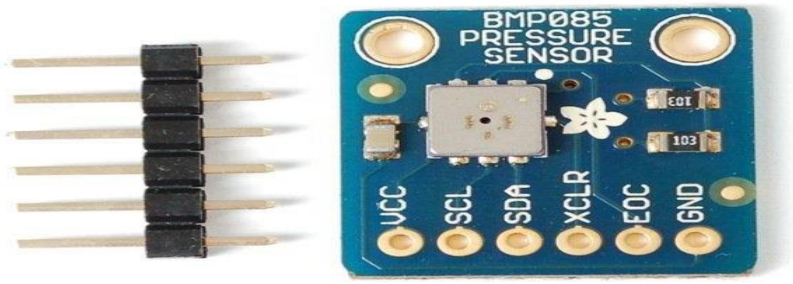

Figure 4.1.7.1: BMP085 Sensor

4.1.7.2 Working principle:

The BMP085 consists of a piezo-resistive sensor, an analog to digital converter and a control unit with E2PROM and a serial I2C interface. The BMP085 delivers the uncompensated value of pressure and temperature. The microcontroller sends a start sequence to start a pressure or temperature measurement. After converting time, the result value (pressure or temperature respectively) can be read via the I2C interface. For calculating temperature in °C and pressure in hpa ,the calibration data has to be used .These constants can be read out from the BMP085 E2PROM via the I2C interface at software initialization. The sampling rate can be increased up to 128 samples per second (standard mode) for dynamic measurement. In this case, it is sufficient to me a sure the temperature only once per second and to use this value for all pressure measurements during the same period.

The BMP085 delivers the uncompensated value of pressure and

temperature after receiving the start signal from the processor. The EEPROM has stored 176 bit of individual calibration data. This is used to compensate offset, temperature dependence and other parameters of the sensor.

UP = pressure data (16 to 19 bit) UT= temperature data (16 bit)

4.1.7.3 Features:

- Wide barometric pressure range.
- Flexible supply voltage range.
- Ultra-low power consumption.
- Fully calibrated.
- Temperature measurement included.

4.1.7.4 Applications:

- BMP085 navigation enhancement.
- In and outdoor navigation.
- Leisure, sports and health monitoring.
- Weather forecast.
- Fan power control.

4.1.7.5Technical Details:

- Vin 3 to 5 Vdc
- Pressure sensing range:300-1100 hpa(9000m to – 500m above sea level)
- Up to 0.03 hpa /0.25m resolution.
- -40 to +85◦c operational range, +-2◦c temperature accuracy.
- This board /chip uses I2C 7-bit address 0X77.
- Package – LGA package.

4.1.7.6 Pin Configuration:

Sl.no	Name	Function	Type
1	VCC	Power Supply	Power
2	SCL	I²C serial bus clock input	Digital input
3	SDA	I²C Serial bus data	Digital bi-directional
4	XCLR	Master clear(low active) input	Digital input
5	EOC	End of conversion	Digital output
6	GND	ground	power

Table 4.1.7.6: Pin configuration of BMP085

4.1.8 Emulator:

4.1.8.1 Introduction:

Arduino shield, ideal for AS3935 lightning sensor development. The lightning "emulator" generates a RF signal that mimics lightning strikes. This board is in an Arduino Uno form factor, and only uses GPIO and I2C, so can be stacked on many form factors (developed on an Uno and Mega).Breakout board for the AS3935 digital lightning sensor based on the AMS reference design. Includes specially tuned antenna, SPI or I2C interfacing, and a wide 2.4V to 5.5V standard operating range. This innovative sensor is designed to interface with most current development systems and boards, including all current Arduino modules. The breakout board features an inductor (antenna) specially designed for this application, and the board ships fully calibrated. This ensures that you don't have to write a massive back-end to support low-level IC calibration, just focus on your final application.

4.1.8.2 Features:

- Can be configured to simulate close, medium and far strikes
- Effective range is ~5-15cm from emulator antenna to sensor antenna
- Arduino form factor and example code makes it easy to install and use
- Three pushbuttons with LED indicators to indicate function and status

- Male straight header included, not installed

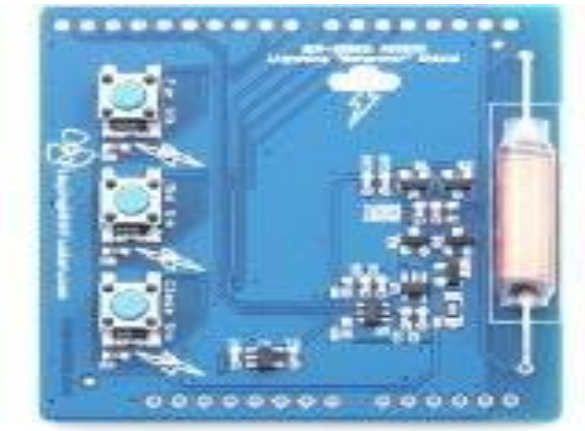

Figure 4.1.8.1: Emulator Image

4.1.8.3 Applications:

- Weather stations
- Sports electronic equipment
- Cell phones
- Smart watches
- Golf equipment

4.1.9 Analog to digital Converter:

4.1.9.1 Introduction:

The Microchip MCP300X family of 10-bit analog-to-digital converters (ADCs) combines' high performance and low power consumption in a small package – making it ideal for embedded control applications. Consisting of the MCP3001, MCP3002, MCP3004 and MCP3008, the MCP300X family features a successive approximation register (SAR) architecture and an industry standard SPITM serial interface. Devices are available with 1, 2, 4 or 8 input channels and in PDIP, SOIC and TSSOP packages. The MCP300X family offers existing Microchip customers added flexibility when incorporating analog inputs into their designs. The industry standard SPI interface allows 10-bit ADC capability to be added to any PIC micro or microcontroller. In addition, new customers will find the performance and price of the MCP300X family very attractive.

4.1.9.2 Working Principle:

The basic principle of operation is to use the comparator principle to determine whether or not to turn on a particular bit of the binary number output. it is typical for an ADC to use a digital-to-analog converter(DAC) to determine one of the inputs to the comparator Converts analog signals into binary words.

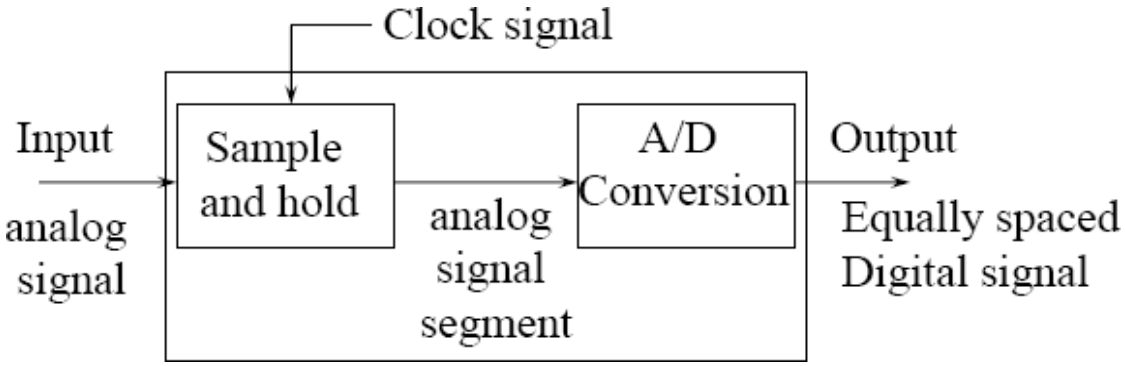

Figure 4.1.9.2: Block diagram of ADC

4.1.9.3 Analog signal:

Analog signals – directly measurable quantities in terms of some other quantity.

Examples:

Thermometer – mercury height rises as temperature rises.

Car Speedometer – Needle moves farther right as you accelerate.

Stereo – Volume increases as you turn the knob.

4.1.9.4 Digital signals:

Digital Signals – have only two states. For digital computers, we refer to binary states, 0 and 1. “1” can be on, “0” can be off.

Examples:

Light switch can be either on or off.

Door to a room is either open or closed.

Figure 4.1.9.4: MCP3008 diagram

4.1.9.5 Analog to Digital Conversion:

- *Quantizing:* It breaking down analog value is a set of finite states.

The number of possible states that the converter can output is:

$$N=2^n$$

Where n is the number of bits in the AD converter

Example: For a 3 bit A/D converter, $N=2^3=8$.

- *Encoding:* It assigning a digital word or number to each state and matching it to the input Signal.

Analog quantization size Q= (Vmax - Vmin)/N = (10V – 0V)/8 = 1.25V

4.1.9.6 Features:

- 200k samples/second • 1, 2, 4 or 8 channels
- Low Power: 5 nA typical standby, 425 µA typical active
- ±1 LSB INL, ±1 LSB DNL
- No missing codes
- Industrial temperature range: -40°C to +85°C
- Single supply operation: 2.7V to 5.5V
- SPI serial interface
- PDIP and SOIC packages

4.1.9.7 Application:

Applications for the MCP300X family include data acquisition, instrumentation and measurement, multichannel data loggers, industrial PCs, motor control, robotics, industrial automation, smart sensors, portable instrumentation and home medical appliances.

4.1.9.8 Types of A to D converters:

- Flash ADC
- Delta-Sigma ADC
- Dual Slope (integrating) ADC
- Successive Approximation ADC

4.1.9.9 Applications:

- *Micro phones:* Take your voice varying pressure waves in the air and convert them into varying electrical signals.
- *Strain Gages:* Determines the amount of Strain When a Stress is applied**.**
- *Thermocouple:* Temperature measuring device converts thermal energy to electrical energy.

4.1.10 ARDUINO:

Arduino is an open-source prototyping platform based on easy-to-use hardware and software.

- Open source and extensible software- The Arduino software is published as open source tools, available for extension by experienced programmers.
- Open source and extensible hardware- The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it.

Figure 4.1.10: Arduino UNO

- The Arduino software is easy-to-use for beginners.
- Inexpensive, Cross-platform, Simple clear programming environment.
- It is flexible enough for advanced users. It rruns on Mac, Windows and Linux.

- Arduino is a key tool to learn new things. Anyone- children, hobbyists, artists, programmers-can start tinkering just following the step by step instructions.

4.1.10.1 Features of Arduino Uno:

- 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 of which 0.5 KB used byBoot loader
- SRAM - 2 KB
- EEPROM - 1 KB
- Clock Speed - 16 MHz

4.1.10.2 Description of Arduino Uno:

The Arduino Uno is microcontroller board based on ATmega328P. It has 14 digital input/output pins of which 6 are PWM, 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller.

An important aspect of the Arduino is its standard connectors, which let users connect CPU board to a variety of inter-changeable add-on modules termed shields. Some shields communicate with Arduino board directly over various pins, but many shields are individually addressable via an I^2C serial bus- so many shields can be stacked and used in parallel.

4.1.10.3 Pin Description:

a. Digital pins: pin no0-13(out of 6 pins with PWM supported)

Digital inputs will come to the Arduino as either on or off (HIGH or LOW,

respectively).

- HIGH is 5VDC,LOW is 0VDC.
- Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode (), digitalWrite () and digitalRead () functions.

Pins with specialized functions:

- *Pin 0 & 1: Serial(RX)&(TX)*

  Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 (portD 0 and 1) USB-to-TTL Serial chip.

- *Pin 2 & 3: External Interrupts*

  These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value connected to portD 2 & 3rd pins.

- *SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).*

  These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.

- *LED: (pin: 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.

- *PMW Pins 3, 5, 6, 9, 10, and 11*

  Arduino provide 8-bit PWM output with the analogWrite() function.

- *Pulse Width Modulation*
  - The on-off pulsing happens so quickly, the connected output device “sees” the result as a reduction in the voltage

b. Analog input pins:

- The Uno has 6 analog inputs, each of which provides 10 bits of resolution.
- Generally in all other microcontrollers there will be only digital pins, in case of sensors we use ADC to convert the analog input to digital input.
- *Some pins have specialized functionality*:
  - I2C: 4 (SDA) and 5 (SCL)

    Support I2C (TWI) communication using the Wire library.
  - AREF

    Reference voltage for the analog inputs used with analog Reference ().
  - Reset

Bring this line LOW to reset the microcontroller

c. Combination of processor pins with Arduino board:

Pin description:

- *Port B(PB7:0) XTAL1/XTAL2/TOSC1/TOSC2*
  - Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit).
  - The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.
- *Port C (PC5:0)*
  - Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). ThePC5...0 output buffers have symmetrical drive characteristics
- *PC6/RESET*
  - If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port.
- *Port D (PD7:0)*
  - Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit).

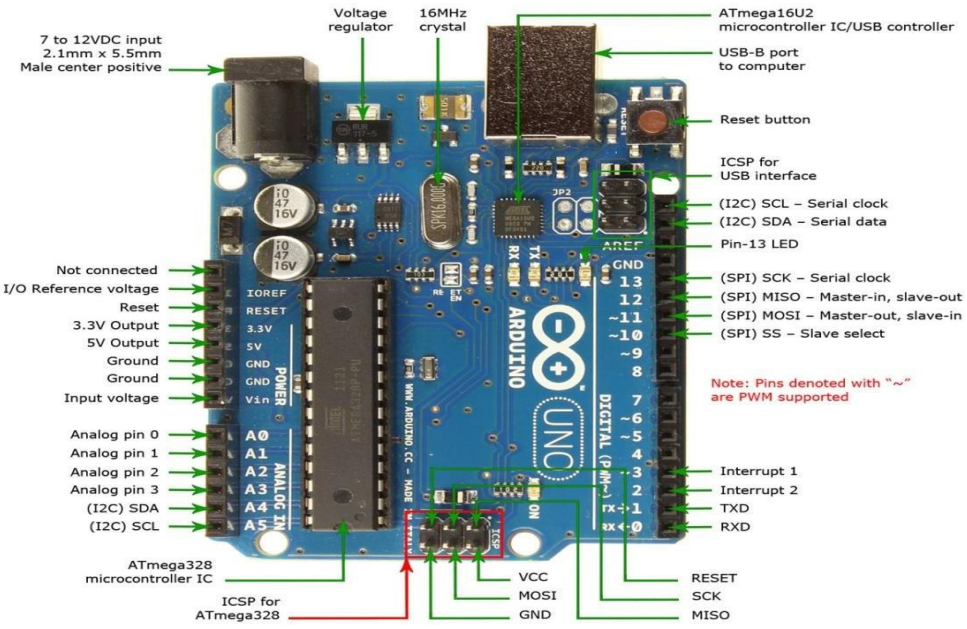

Figure 4.1.10.3(a): Pin description of Arduino Uno

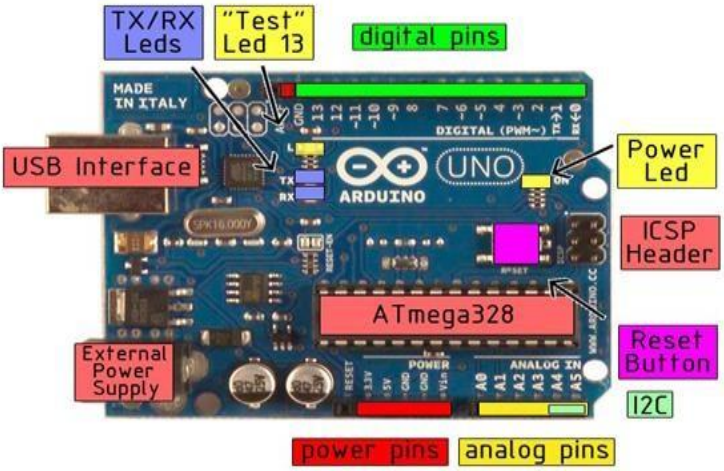

Figure 4.1.10.3(b): Various modules of Arduino Uno

- *AVCC*
  - AVCC is the supply voltage pin for the A/D Converter, PC3:0, and ADC7:6. It should be externally connected to VCC, even if the ADC is not used AREF
- *ADC7:6 (TQFP and QFN/MLF Package Only)*
  - In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter

d. SUPPLY:

The various power supply pins are as follows,

- VIN: The input voltage to the Arduino board when it's using an external power source
- 5V: This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- 3.3V: A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

4.1.10.4 Communication:

- 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).
- 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.

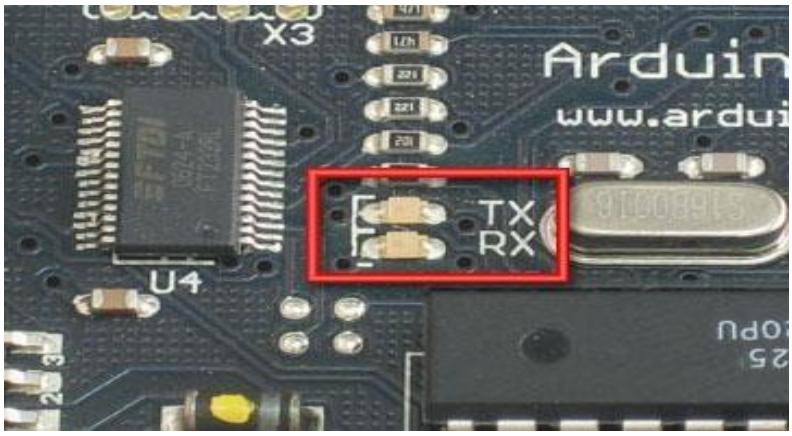

Figure 4.1.10.4: Serial communication pins of Uno

- A Software Serial library allows for serial communication on any of the Uno's digital pins.

The ATmega328 also support I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus

4.1.10.5 Serial Communication:

- Compiling turns your program into binary data (ones and zeros)
- Uploading sends the bits through USB cable to the Arduino
- The two LEDs near the USB connector blink when data is transmitted
- RX blinks when the Arduino is receiving data
- TX blinks when the Arduino is transmitting data
- We can use TTL pins instead serial communication

4.2 SOFTWARE REQUIREMENTS:

4.2.1 PYTHON LANGUAGE:

Guido van Rosum created the Python programming language in the late 1980s. In contrast to other popular languages such as C, C++, Java, and C#, Python strives to provide a simple but powerful syntax. Python is used for software development at companies and organizations such as Google, Yahoo, CERN, Industrial Light and Magic, and NASA. Experienced programmers can accomplish great things with Python, but Python's beauty is that it is accessible to beginning programmers and

allows them to tackle interesting problems more quickly than many other, more complex languages that have a steeper learning curve.

4.2.1.1Writing A Python Program:

Python programs must be written with a particular structure. The syntax must be correct, or the interpreter will generate error messages and not execute the program. This section introduces Python by providing a simple example program. Listing 1.1 (simple.py) is one of the simplest Python programs that does something

How To Run Listing 1.1 (Simple.Py) Using Idle's Editor: IDLE has a built in editor. From the IDLE menu, select New Window, as shown in Figure 4.2.1.1(a).

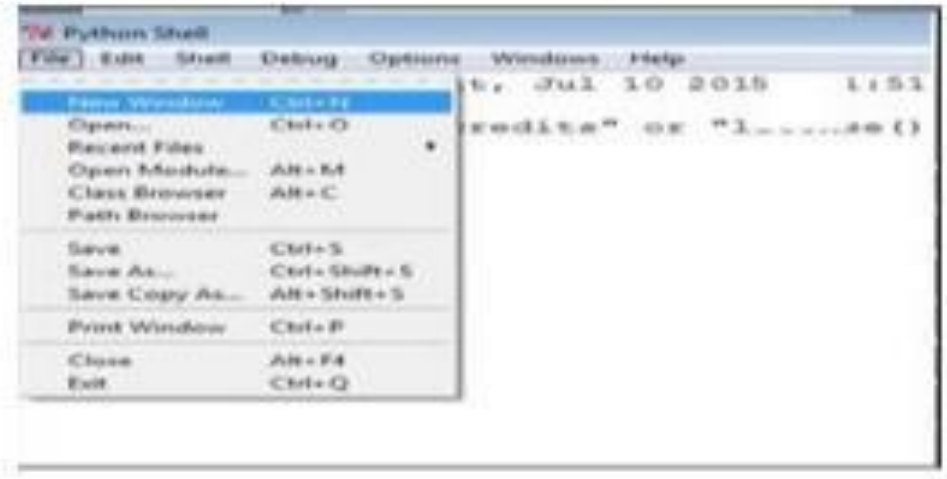

Figure 4.2.1.1(a): Launching the idle editor

Figure 4.2.1.1(b): The simple python program typed into the idle editor

Figure 4.2.1.1(b) shows the resulting editor window with the text of the simple Python program. Save the program using Save option in the File menu as shown in figure 4.2.1.1(C).

Spare the code to a record named simple.py. The genuine name of the record is immaterial, yet the name "basic" precisely depicts the idea of this program. The expansion .py is the augmentation utilized for Python source code.

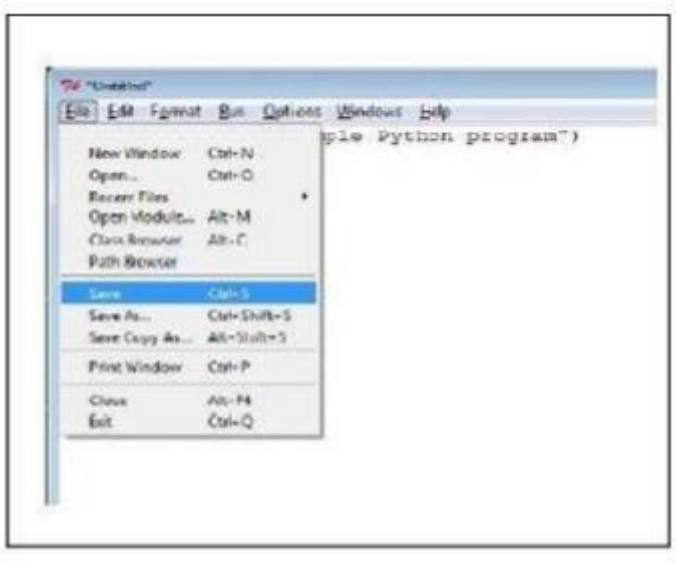

Figure 4.2.1.1(c): Saving a file created with the idle editor

We can run the program from inside the IDLE proofreader by squeezing the F5 work key or from the editorial manager's Run menu: Run! Run Module. The yield shows up in the IDLE intuitive shell window.

4.2.2 Thing Speak:

The Internet of Things gives access to an expansive scope of inserted gadgets and web administrations. Thing Speak is an open information stage and API for the Internet of Things that empowers you to gather, store, investigate, imagine, and follow up on information from sensors or actuators, for example, Adriano, Raspberry Pi, Beagle Bone Black, and other equipment. For instance, with Thing Speak you can make sensor-logging applications, area following applications, and an interpersonal organization of things with announcements, so you could have your home indoor regulator control itself in light of your present area. The essential component of Thing Speak movement is the channel, which contains information fields, area fields, and a status field. After you make a Thing Speak channel, you can compose information to the channel, process and view the information with MATLAB code, and respond to the information with tweets and different cautions. The regular Thing Speak work process lets you: Make a Channel and gather

information Analyze and envision the information Follow up on the information utilizing any of a few Apps Thing Speak API is accessible on Git Hub and incorporates the entire Thing Speak API for handling HTTP asks for, putting away numeric and alphanumeric information, numeric information preparing, area following, and announcements.

4.2.2.1 Collect information in new channel:

This case demonstrates to make another channel to gather broke down information. You read information from people in general Thing Speak channel 103024 – Precision farming, and compose it into your new channel.

A. Create a channel: Sign In into Thing Speak utilizing either your Math Works Account or Thing Speak account, or make another Thing Speak.

B. Click Channels > My Channels. On the Channels page, click New Channel, shown in figure 4.2.2(a).

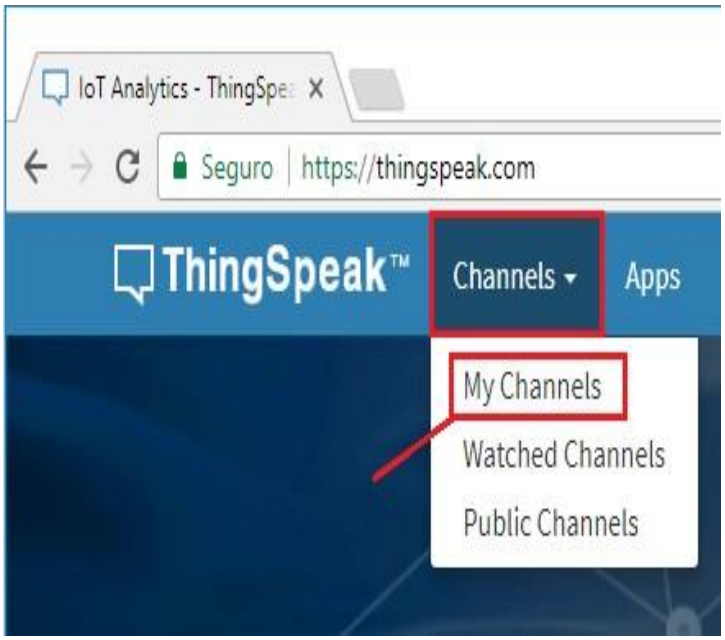

Figure 4.2.2(a): channel selection

Check the boxes next to Fields 1–3 as shown in figure 4.2.2(b). Enter these channel setting values:

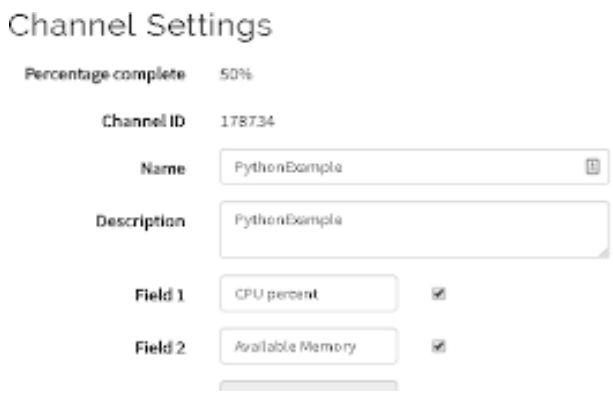

Figure 4.2.2(b): Channel settings

Name: LIGHTNING DETECTION

- Temperature
- Pressure

A. Click Save Channel at the bottom of the settings. You now see these tabs:

1. Private View: This tab shows data about your channel that no one but you can see.
2. Public View: If you make your channel openly accessible, utilize this tab to show chose fields and channel perceptions.
3. Channel Settings: This tab demonstrates all the channel alternatives you set at creation. You can alter, clear, or erase the channel from this tab.
4. API Keys: This tab shows your channel API keys. Utilize the keys to peruse from and keep in touch with your channel.
5. Data Import/Export: This tab empowers you to import and fare channel information.

B. Analyze your information: This case demonstrates to peruse temperature and stickiness information from Thing Speak channel 103024-Precision Agriculture, which gathers soil related information from an Arduino® gadget. You compose the temperature and stickiness information into your exactness agribusiness channel, alongside the ascertained soil dampness information. The channel at that point enables to picture the outcomes.

- Sign In either to your Math Works Account or Thing Speak account, or make another Thing Speak account.
- Create a Channel as your Precision agribusiness channel.
- Write information to your channel:
- This method peruses dampness, temperature.
- Use a MATLAB Analysis application to peruse, figure, and compose your information.
- Go to the Apps tab and snap MATLAB Analysis. At that point click New. Select the Custom layout, and snap Create.
- In the Name field, enter Dew Point Calculation.
- In the MATLAB Code field, enter the accompanying lines of code.
- Save people in general exactness agribusiness channel ID and your accuracy horticulture channel ID to factors.
- reached = 103024;
- write hid = 103024;
- Save your Write API Key to a variable.
- write Key = 'L1Z55NZ10NZGNHKL'; To discover your Channel ID and Write API Key, allude to Channel Info on the My Channels tab

4.2.3 Website for IOT based lightning prediction system (viganataranga):

In this project we designed a website for showing the location were lightning is detected ,so that we can take immediate action to those people leaving in that particular location.For designing website we used different web languages they are namely three:

- HTML
- CSS
- PHP

The website is interlinked with the google maps to show the location.

4.2.3.1 HTML

Hyper Text Markup Language (HTML) is the primary building block of creating a website. HTML is a very basic markup language and requires

memorization of a few dozen HTML commands that structure the look and layout of a web page. Before writing any HTML code or designing your first web page, you must decide on an HTML editor or text editor, such as Notepad or WordPad.

Once you have obtained an HTML editor and are ready to begin setting up your website, think about how you want the site to look and be set up. Consider even drawing out your ideas, to help visualize the site and pages in the site. Below are some considerations to think about when designing your web page.

4.2.3.1.1 Writing basic HTML:

After installing an HTML editor and setting up a folder, you are ready to begin creating your page. Begin by creating a file named index.htm or index.html as your start page. All servers on the Internet look for an index file if no file is specified. Once you have created the index.htm or index.html file and it is open in your HTML editor, we recommend inserting the below source code into your page. If your HTML editor automatically places HTML code into your page or you have a WYSIWYG editor, you can skip this step.

The above code is a very basic example of the code that helps make up every web page. As you can see, the code starts with <html>, which is defining that everything within <html> is HTML code. Next, you have <head>, which is defining the heading of your HTML document. Third, we have the <title> section within <head>, which defines the web page title that is displayed at the top of the Internet browser window. Finally, the <body> section contains what is shown on the web page.Below is additional code that can be placed in the <body> section of the code to help familiarize you with some of the most commonly used HTML commands.

```
<html>

<head>

<title>My first web page</title>

</head>

<body>
```

```
Your web page content goes here

</body>

</html>
```

Figure 4.2.3.1.1(a): Basic HTML Program

As you can see from looking at the above code, you will realize that the basic HTML commands are fairly simple to use. First, we start off with , which is telling the browser to center the information within these tags. Next, the <h1> or heading one statement tells the browser to display the text in the largest heading style. Next, the <hr> tag tells the browser to display a line straight across the screen. The third line contains
 that creates a line break on the page. Next, the <p> is short for "paragraph" and helps separate the text on the page. Next, the <b> tag is short for bold and will bold the text contained within the tag. Next, the <ul> starts a bullet list and each bullet is represented by the <li> tag. Finally, the "<a href" tag is a method of creating a link to another location. In this example, we are creating a bulleted list of links to Computer Hope and Google.

```
<center><h1>Welcome to my web page</h1></center>

<hr>

<br>

<p>Hello and welcome to my first website.<br><br>

<b>These are my favorite links:</b><br>

<ul>

<li><a href="https://www.computerhope.com">Computer Hope</a></li>
```

```
<li><a href="http://www.google.com">Google</a></li>


</ul>


</p>
```

Figure 4.2.3.1.1(b): HTML basic code for designing a website

4.2.3.2 CSS

Like HTML, CSS is not really a programming language. It is not a *markup language* either — it is a *style sheet language*. This means that it lets you apply styles selectively to elements in HTML documents. For example, to select all the paragraph elements on an HTML page and turn the text within them red, you'd write this CSS:

```
p {
color: red;
}
```

Let's try it out: paste those three lines of CSS into a new file in your text editor, and then save the file as `style.css` in your `styles` directory.

But we still need to apply the CSS to your HTML document. Otherwise, the CSS styling won't affect how your browser displays the HTML document. (If you haven't been following on with our project, read Dealing with files and HTML basics to find out what you need to do first.)

1. Open your `index.html` file and paste the following line somewhere in the head (that is, between the `<head>` and `</head>` tags):

```
<link href="styles/style.css"rel="stylesheet"type="text/css">
```

Save `index.html` and load it in your browser. You

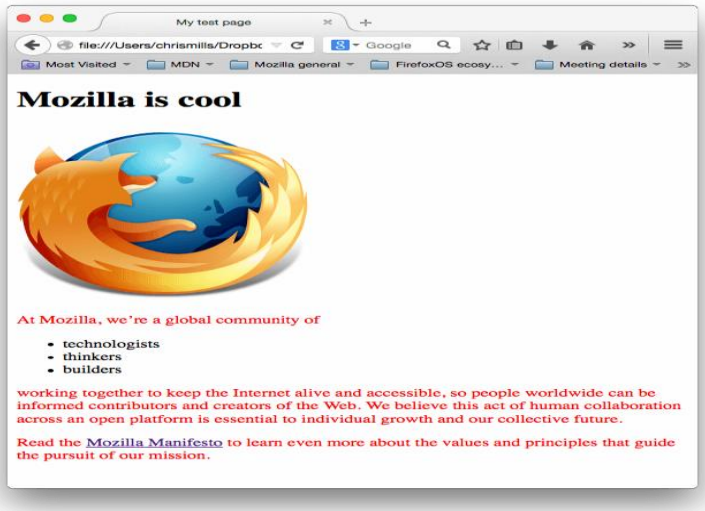

Figure 4.2.3.2(a): Output of the code

Anatomy of a CSS ruleset

Let's look at the above CSS in a bit more detail:

Figure 4.2.3.2(b): Explanation of the code

The whole structure is called a rule set (but often "rule" for short). Note also the names of the individual parts:

Selector

The HTML element name at the start of the rule set. It selects the element(s) to be styled (in this case, `p` elements). To style a different element, just change the selector.

Declaration

A single rule like `color: red;` specifying which of the element's properties you want to style.

Properties

Ways in which you can style a given HTML element. (In this case, `color` is

a property of the `<p>` elements.) In CSS, you choose which properties you want to affect in your rule.

Property value

To the right of the property after the colon, we have the property value, which chooses one out of many possible appearances for a given property (there are many `color` values besides `red`).

Note the other important parts of the syntax:

- Each rule set (apart from the selector) must be wrapped in curly braces (`{}`).
- Within each declaration, you must use a colon (`:`) to separate the property from its values.
- Within each rule set, you must use a semicolon (`;`) to separate each declaration from the next one.

So to modify multiple property values at once, you just need to write them separated by semicolons, like this:

```
p {
color: red;
width:500px;
border:1px solid black;
}
```

Selecting multiple elements

You can also select multiple types of elements and apply a single rule set to all of them. Include multiple selectors separated by commas. For example:

```
p,li,h1 {
color: red;
}
```

Different types of selectors

There are many different types of selectors. Above, we only looked at element selectors, which select all elements of a given type in the given HTML documents. But we can make more specific selections than that.

4.2.3.3 PHP

PHP started out as a small open source project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

- PHP is a recursive acronym for "PHP: Hypertext Preprocessor".
- PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites.
- It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.
- PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time.
- PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time.
- PHP is forgiving: PHP language tries to be as forgiving as possible.
- PHP Syntax is C-Like.

4.2.3.3.1 Common uses of PHP:

- PHP performs system functions, i.e. from files on a system it can create, open, read, write, and close them.
- PHP can handle forms, i.e. gather data from files, save data to a file, through email you can send data, return data to the user.
- You add, delete, modify elements within your database through PHP.

- Access cookies variables and set cookies.
- Using PHP, you can restrict users to access some pages of your website.
- It can encrypt data.

4.2.3.3.2 Characteristics of PHP:

Five important characteristics make PHP's practical nature possible .

- Simplicity
- Efficiency
- Security
- Flexibility
- Familiarity

4.2.3.3.3 "Hello World" Script in PHP:

To get a feel for PHP, first start with simple PHP scripts. Since "Hello, World!" is an essential example, first we will create a friendly little "Hello, World!" script.

As mentioned earlier, PHP is embedded in HTML. That means that in amongst your normal HTML (or XHTML if you're cutting-edge) you'll have PHP statements like this

```
<html>


<head>
<title>Hello World</title>
</head>


<body>
<?php echo "Hello, World!";?>
</body>
```

```
</html>
```

It will produce following result −

```
Hello, World!
```

If you examine the HTML output of the above example, you'll notice that the PHP code is not present in the file sent from the server to your Web browser. All of the PHP present in the Web page is processed and stripped from the page; the only thing returned to the client from the Web server is pure HTML output.

All PHP code must be included inside one of the three special markup tags ATE are recognised by the PHP Parser.

```
<?php PHP code goes here ?>


<?    PHP code goes here ?>


<scriptlanguage="php"> PHP code goes here </script>
```

Figure 4.2.3.3(a): Syntax Of Php

A most common tag is the <?php...?> and we will also use the same tag in our tutorial.

From the next chapter we will start with PHP Environment Setup on your machine and then we will dig out almost all concepts related to PHP to make you comfortable with the PHP language.

4.2.4 ARDUINO SOFTWARE:

The Arduino integrated development environment (IDE) is a cross-platform application written in Java, and derives from the IDE for the Processing programming language and the Wiring projects. It is designed to introduce programming to artists and other newcomers unfamiliar with software development. It includes a code editor with features such as syntax highlighting, brace matching, and automatic indentation, and is also capable of compiling and uploading programs to the board with a single

click. A program or code written for Arduino is called a "sketch". Arduino programs are written in C or C++.

- setup(): a function run once at the start of a program that can initialize settings
- loop(): a function called repeatedly until the board powers off

Arduino boards can be controlled using an implementation of Wiring, which is a version of Processing developed specifically for electronic I/O. Arduino looks like Processing, but is actually built in C, so there are a few differences to look out for.

Arduino is a standard window application and starts by clicking on the program icon. A window for writing “program” or “sketch” will be opened as shown below.

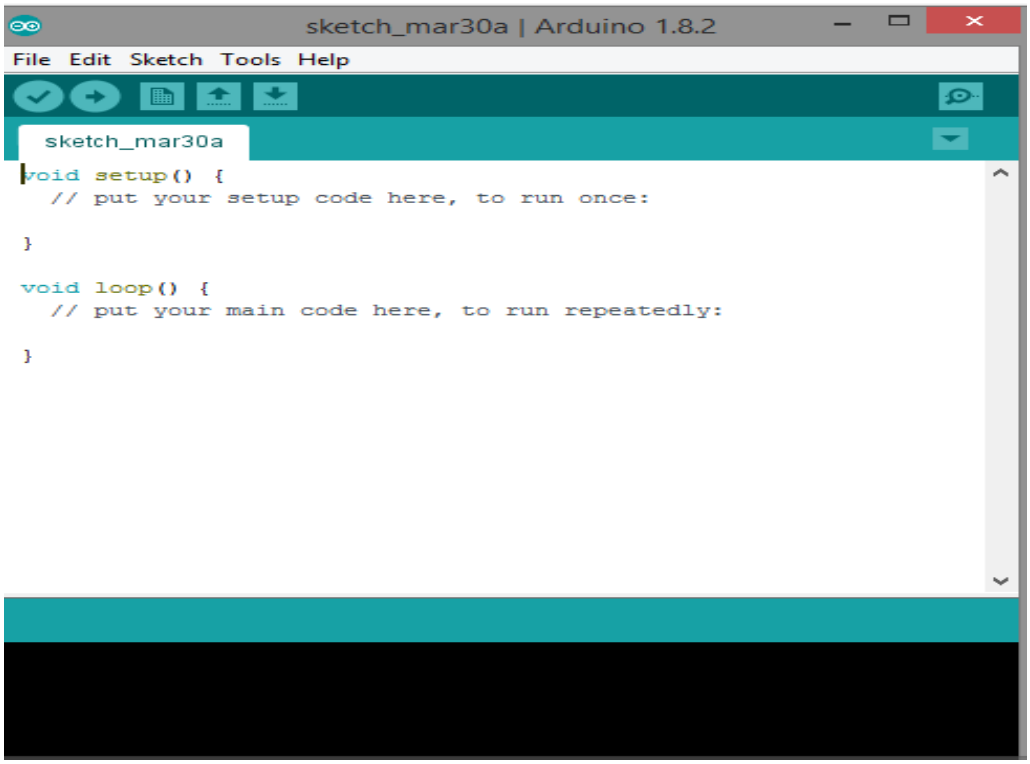

Figure 4.2.4: Arduino IDE

And a code is written and then saved with desired name. Then it is compiled and checked for any errors which are displayed in below space. By interfacing the arduino board using USB, the code is uploaded directly by selecting the uploading option in the arduino software itself.

A typical first program for a microcontroller simply blinks an LED on and off. In the Arduino environment, the user might write a program like this:

```
The integrated pin 13 LED
#define LED_PIN 13
void setup(){
pinMode(LED_PIN, OUTPUT);// Enable pin 13 for digital output
}
void loop(){
digitalWrite(LED_PIN, HIGH);// Turn on the LED
delay(1000);// Wait one second (1000 milliseconds)
digitalWrite(LED_PIN, LOW);// Turn off the LED
delay(1000);// Wait one second
}
```

It is a feature of most Arduino boards that they have an LED and load resistor connected between pin 13 and ground; a convenient feature for many simple tests. The previous code would not be seen by a standard C++ compiler as a valid program, so when the user clicks the "Upload to I/O board" button in the IDE, a copy of the code is written to a temporary file with an extra include header at the top and a very simple main() function at the bottom, to make it a valid C++ program.

CHAPTER 5

BLOCKDIAGRAM

The block diagram of the proposed system as shown in Figure 5.1 consists of sensing Unit such as Air Quality Sensor, temperature and humidity sensor, Barometric and Pressure Sensor, Lightning sensor.

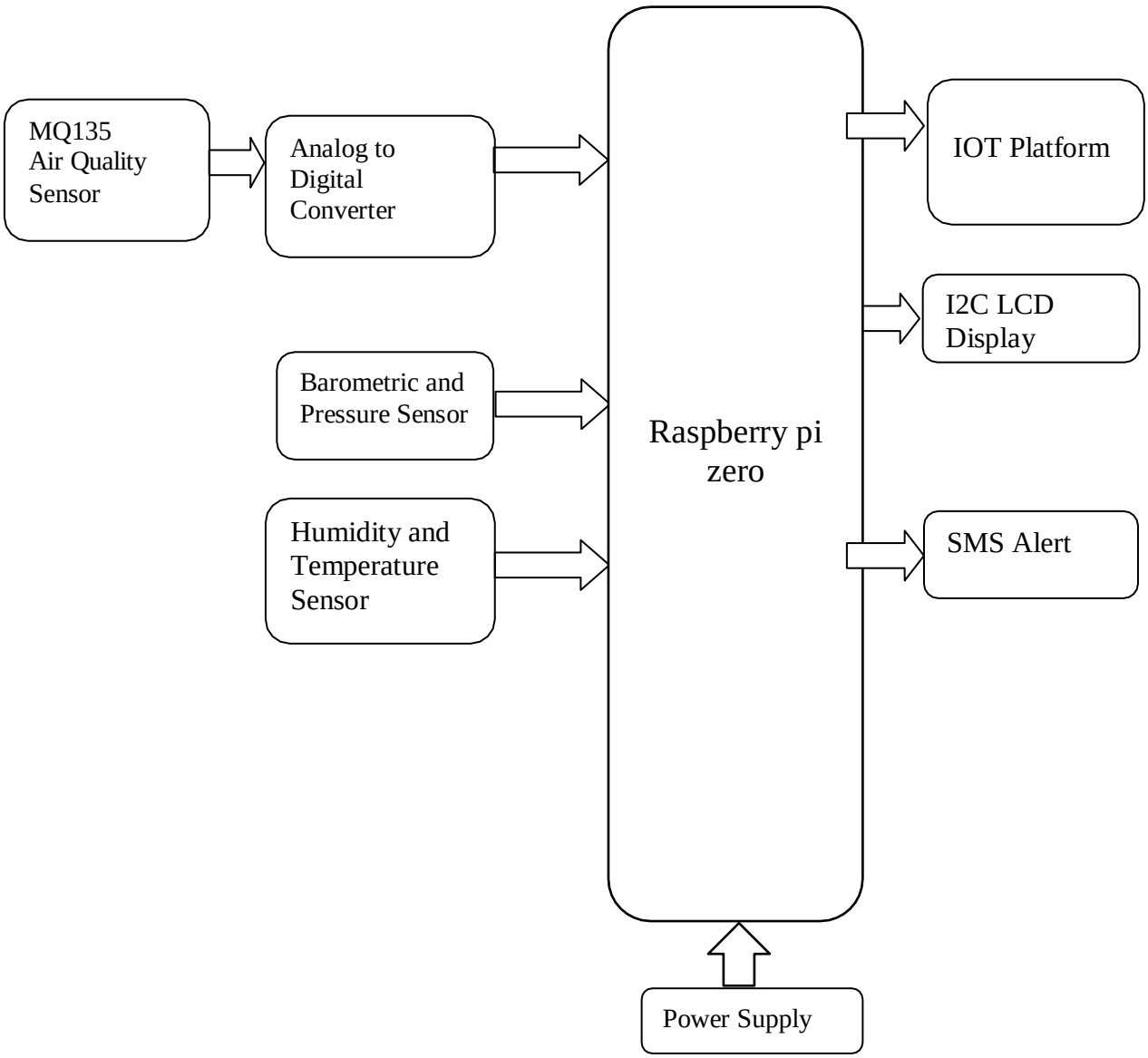

Figure 5.1: Block Diagram

5.2 Working Principle:

From the above outline, plainly the controller gathers all the data from all the sensors and send as a sms if its alert related to lightning as well as through website we can see the location where lightning has been occurred. And weather information will be uploaded to think speak as well as shown in LCD display.

Raspberry pi is interfaced with MQ135, BMP085, DHT22, AS3935 sensors.MQ135 sensor to obtain air quality in an environment (harmful gases like ammonia, nitrogen oxide, smoke, CO2).BMP085 sensor to get pressure condition in pascal. DHT22 sensor to give the data regarding humidity and temperature .AS3935 sensor to detect lightning strike before a couple of minutes. Up to 40 km this sensor can able to detect with more accuracy.

It will be useful if we install this in a villages because of low range of the sensor. All the weather information will be displayed in LCD and uploaded in thing speak. This system will keeps on tracking information regarding lightning strike .If lightning occurs then that it sends an alert sms to all the people in the respective villagers so that they can be alert n move to safe place. Not only sms it also provides location through website. Because of it consist of all these features its an effective system when compared to existing one .In which IoT place a major role.

CHAPTER 6

RESULTS

The present lightning detection by using IOT and control system is more advantageous than the existing system because it can detects the lightning and send an alert and message to the nearer people. It can detect temperature, humidity, pressure of wind and any harmuful gases in air.

To obtain the relevant information, we designed system by using Raspberry pi, Temperature (DHT22) and Humidity sensor, Barometric and pressure sensor(BMP085) ,and LCD. The DHT22 sensor is used to measure the temperature of the surrounding air. The BMP085 is used to measure the Pressure.

The experimental results are documented as follows:

Figure 6.1(a): Displaying the status of Lightning Detection

Figure 6.1(b): Displaying status of Humidity

Figure 6.1(c): Displaying status of Air quality

Figure 6.1(d): Displaying status of Pressure

Figure 6.1(e): Displaying status of Temperature

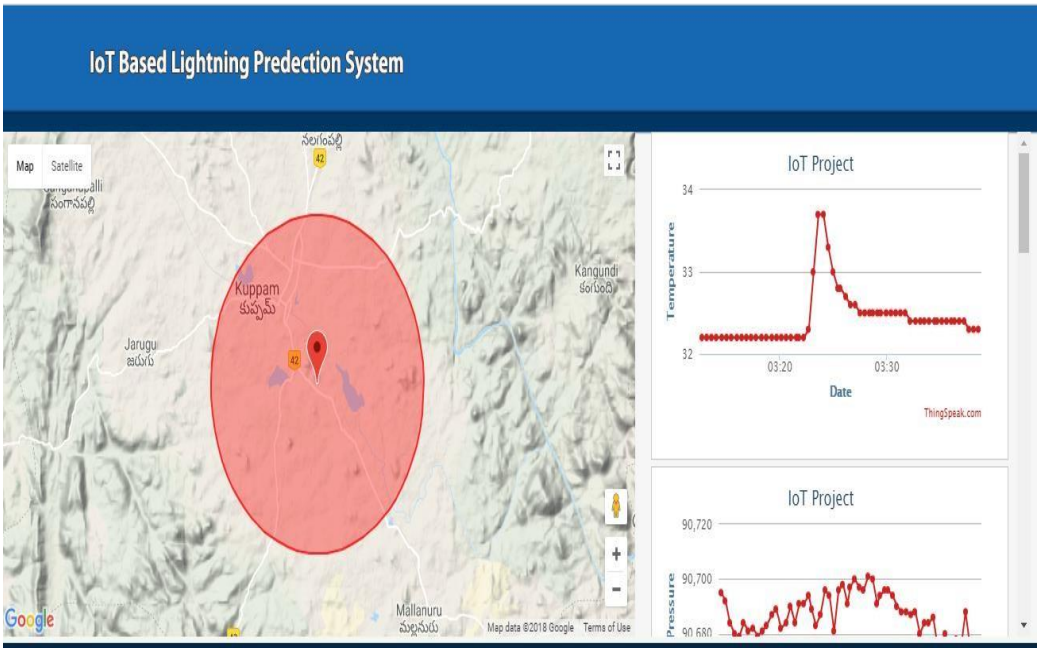

Figure 6.1(f): (LEFT) Showing the lightning detected location in website (www.viganataranga.com) by using the google maps & (RIGHT) showing the graphs of temperature and pressure that uploaded in thingspeak.

CHAPTER 7

CONCLUSION

We know that Lightning Strikes is one of the natural disaster that leads to loss of lots of people as well as nature. To provide solution for this Lighting Detection using IOT system has been designed which provides weather information and lightning detection alert. Weather information has been uploaded to IoT platform as well as in display which is helpful for the people to know weather condition so that they can take further steps. Lightning alert has been send through sms as well as location has been shown through website. So that we can able to find the location where lightning has been occurred. This system provides effective solutions to all the problems which has been present in existing system.

REFERENCES

[1] A European lightning density analysis using 5 yr of ATDnet data
G.Anderson and D. Klugmann. Natural Hazards and Earth System Sciences December 2013

[2] Analysis of warm season thunderstorms using an object-oriented tracking method based on radar and total lightning data
T.Rigo, N. Pineda, and J. Bech Natural Hazards and Earth System Sciences September 2010

[3] Lightning Applications in Weather and
Climate Research Colin G. Price Surv
Geophysics December 2013

[4] Frequency distributions ofsome parameters of negative downward lightning flashes based on accurate-stroke-count studies
Maurício G. Ballarotti,1 Claudia Medeiros,1 Marcelo M. F. Saba,1 Wolfgang Schulz,2 and Osmar Pinto Jr. Journal of Geophysical Reserach Vol 117 March 2012

[5] High-resolution climatology of lightning characteristics within Central Europe, Wapler, K. (2013): Meteol. Atmos. Phys., 122(3-4), 175-184, DOI 10.1007/s00703-013-0285-1
http://link.springer.com/article/10.1007%2Fs00703-013-0285-1

[6] Local Time Variation in Land/Ocean Lightning Flash Density as Measured by the World Wide Lightning Location Network

Erin H. Lay(1), Abram R. Jacobson(1), Robert H. Holzworth(1), Craig J. Rodger(2), Richard L. Dowden Journal of Geophysical Research Vol. 112

[7] On The Relationship Between Terrain Variations and LLS-Derived Lightning Parameters Kenneth L. Cummins International Conference on Lightning Protection (ICLP) Vienna 2012

[8] The importance of total lightning in the future of weather Nowcasting Nicholas W. S. Demetriades, Martin J. Murphy and Ronald L. Holle >AMS January 2004

[9] Poelman, Dieter R., Wolfgang Schulz, Christian Vergeiner, 2013: Performance Characteristics of Distinct Lightning Detection Networks Covering Belgium. *J. Atmos. Oceanic Technol.*, 30, 942–951.

[10] Webinar on Lightning Performance Validation Techniques Ken Cummins, published online by Vaisala (http://www.vaisala.com/en/services/training/seminarsandwebinars/lightningwebinars/Pages/default.aspx)

[11] Rakov, V. A., M. A. Uman, and R. Thottappillil (1994), Review of lightning properties from electric field and TV observations, J. Geophys. Res., 99, 10,745–10,750, doi:10.1029/93JD01205.

[12] Schulz, W., Cummins, K., Diendorfer, G., and Dorninger, M.: Cloud-to-ground lightning in Austria: 5 a 10-year study using data from a lightning location system, J. Geophys. Res., 110, D09101, doi:10.1029/2004JD005332, 2005. 6880, 6891

[13] R. K. Said M. B. Cohen and U. S. Inan Highly intense lightning over the oceans: Estimated peak currents from global GLD360 observations. JOURNAL OF GEOPHYSICAL RESEARCH: ATMOSPHERES, VOL. 118, 1–11, doi:10.1002/jgrd.50508, 2013

[14] S. Pédeboy and W. Schulz. Validation of a ground strike point identification algorithm based on ground truth data. ILDC 2014

[15] Maxime Brissard et all. Spatial and temporal homogeneity of lightning archive. ILDC 2012

D. R. MacGorman, M. W. Maier, W. D. Rust, *Lightning Strike Density for the Contiguous United States from Thunderstorm Duration Records*, May 1984.
2.

S. A. Prentice, "The CIGRE Lightning Flash Counter: Australian Experience", *CIGRE Paper*, 1974.
2.

P. Y. Guillo, *Study of Lightning Parameters Using Advanced Analytical and Statistical Tools*, University of Toronto, September 1985.
2.

V. P. Idone, R. E. Orville, "Lightning Return Stroke Velocities in the Thunderstorm Research International Program (TRIP)", *J. Geophys. Res.*, vol. 87, pp. 4903-4915, 1982.
CrossRef Google Scholar
3.

K. Berger, R. B. Anderson, H. Kroninger, "Parameters of lightning flashes", *Electra*, pp. 23-37, 1975.
3.

Appendix –A: Project Code

```
Python 2.7.14 (v2.7.14:84471935ed, Sep 16 2017, 20:19:30) [MSC v.1500 32 bit
(Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> import I2C_LCD_driver
from time import *
import Adafruit_BMP.BMP085 as BMP085
from RPi_AS3935 import RPi_AS3935
import RPi.GPIO as GPIO
from datetime import datetime
import sys
import Adafruit_DHT
import Adafruit_GPIO.SPI as SPI
import Adafruit_MCP3008
import time
import urllib
import requests


CLK = 18
MISO = 23
MOSI = 24
CS = 25
DHTsensor = Adafruit_DHT.DHT22
mylcd = I2C_LCD_driver.lcd()
sensor = BMP085.BMP085()
GPIO.setmode(GPIO.BCM)
AS3935sensor = RPi_AS3935(address=0x03, bus=1)
AS3935sensor.set_indoors(True)
AS3935sensor.set_noise_floor(0)
AS3935sensor.calibrate(tun_cap=0x0F)
mcp = Adafruit_MCP3008.MCP3008(clk=CLK, cs=CS, miso=MISO, mosi=MOSI)
a=1
```

```
def handle_interrupt(channel):
    a=0
    time.sleep(0.003)
    global AS3935sensor
    reason = AS3935sensor.get_interrupt()
    if reason == 0x01:
        print "Noise level too high - adjusting"
        AS3935sensor.raise_noise_floor()
    elif reason == 0x04:
        print "Disturber detected - masking"
        AS3935sensor.set_mask_disturber(True)
    elif reason == 0x08:
        now = datetime.now().strftime('%H:%M:%S - %Y/%m/%d')
        distance = AS3935sensor.get_distance()
        print "We sensed lightning!"
        print "It was " + str(distance) + "km away. (%s)" % now
        print "Energy was" + str(AS3935sensor.get_energy())
        print ""
        mylcd.lcd_display_string('                    ', 1)
        mylcd.lcd_display_string('                    ', 2)
        str_pad = " " * 16
        my_long_string = "Alert lightning Detected!"
        my_long_string = str_pad + my_long_string
        mylcd.lcd_display_string("Est Dist:" + str(distance) +"Km " , 2)
        for i in range (0, len(my_long_string)):
            lcd_text = my_long_string[i:(i+16)]
            mylcd.lcd_display_string(lcd_text,1)
            sleep(0.4)
            mylcd.lcd_display_string(str_pad,1)
        authkey = '150506Afze9Ckx590c2551' # Your authentication key.
        mobiles = "7829285032,8008252580,8050915026"
```

```
        message = "Hai, A Lightning strike is imenent in your area. Estimated distance is:" + distance + "Kms. Strike is predicted in 30-45 Minutes.Please stay indoors till the storm clears out."
        sender = "THUNDC"
        route = "9"
        values = {'authkey' : authkey,'mobiles' : mobiles,'message' : message,'sender' : sender,'route' : route            }
        url = "https://control.msg91.com/api/sendhttp.php" # API URL
        postdata = urllib.urlencode(values) # URL encoding the data here.
        try:
            req = requests.get(url, params=postdata)
        except:
            pass
    a=1
authkey = '150506Afze9Ckx590c2551' # Your authentication key.
mobiles = "7829285032,8008252580,8050915026,7330739122,8639842038"
message = "Hai, A Lightning strike is imenent in your area. Estimated distance is:10Kms. Strike is predicted in 30-45 Minutes.Please stay indoors till the storm clears out."
sender = "THUNDC"
route = "9"
values = {'authkey' : authkey,'mobiles' : mobiles,'message' : message,'sender' : sender,'route' : route }
url = "https://control.msg91.com/api/sendhttp.php" # API URL
postdata = urllib.urlencode(values) # URL encoding the data here.
try:
            req = requests.get(url, params=postdata)
            print req
except:
            pass

pin = 17
GPIO.setup(pin, GPIO.IN)
GPIO.add_event_detect(pin, GPIO.RISING, callback=handle_interrupt)
```

```
mylcd.lcd_display_string("Welcome!!!",1)
mylcd.lcd_display_string("Lightning Det:ON",2)
time.sleep(1.5)
while True:
 while a:
  mylcd.lcd_clear()
  mylcd.lcd_display_string('Lightning Det:ON',2)
  temperature, humidity = Adafruit_DHT.read_retry(DHTsensor, '4')
  values = mcp.read_adc(7)
  mylcd.lcd_display_string('Temp={0:0.2f}*C'.format(sensor.read_temperature()), 1)
  time.sleep(a)
  mylcd.lcd_display_string('                    ', 1)
  mylcd.lcd_display_string('Pres={0:0.2f}Pa'.format(sensor.read_pressure()), 1)
  time.sleep(a)
  mylcd.lcd_display_string('                    ', 1)
  mylcd.lcd_display_string('Humd={0:0.1f}%'.format(humidity), 1)
  time.sleep(a)
  mylcd.lcd_display_string('                    ', 1)
  mylcd.lcd_display_string('AirQuaInd=' + str(values), 1)
  time.sleep(a)
  mylcd.lcd_display_string('                    ', 1)
  mylcd.lcd_display_string("Energy=" + str(AS3935sensor.get_energy()), 1)
  time.sleep(a)
  mylcd.lcd_display_string('                    ', 1)
```