DEVELOPMENT OF REAL-TIME, SOLAR POWERED MICROCONTROLLER-BASED WEATHER STATION FOR THE ACQUISITION OF ATMOSPHERIC PARAMETERS

BY

UGHANZE, Ifeanyi James
M.Tech/SPS/2017/6871

DEPARTMENT OF PHYSICS

FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA,

OCTOBER, 2021

DEVELOPMENT OF REAL-TIME, SOLAR POWERED MICROCONTROLLER-BASED WEATHER STATION FOR THE ACQUISITION OF ATMOSPHERIC PARAMETERS

BY

UGHANZE, Ifeanyi James
M.Tech/SPS/2017/6871

A THESIS SUBMITTED TO THE POSTGRADUATE SCHOOL FEDERAL UNIVERSITY OF TECHNOLOGY, MINNA, NIGERIA IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR AWARD OF THE DEGREE OF MASTER OF TECHNOLOGY (M.Tech) IN APPLIED ATMOSPHERIC PHYSICS

OCTOBER, 2021

ABSTRACT

Atmospheric researchers, meteorologists, and farmers continue to face difficulties in assessing real-time weather data. This project created a solar-powered Arduino microcontroller-based weather station that uses meteorological sensors to measure atmospheric temperature, relative humidity, atmospheric pressure, wind speed, and wind direction. The temperature, relative humidity, pressure, wind direction, and wind speed monitoring elements are all part of the weather station. The power supply, electronic control, display, and data transfer sectors are among the others. The output of the sensors is processed by an Arduino Atmega 2560 microcontroller, which displays it on an LCD for on-site users and also transmits it to the Wi-Fi module, which uploads the data online so that out-of-site users can access the measured weather information. For the storing and analysis of measured meteorological parameters, the weather station uses Thinkspeak Technology's Internet of Things (IoT) initiative. For successful real-time atmospheric data logging, the results of the performance evaluation demonstrated appropriate synchronization between electronic sensors, microcontroller, and the internet. Utilisation of low-cost sensors and locally available components made it possible to spend as little as ₦216, 400 (Two Hundred and Sixteen Thousand Four Hundred Naira) to develop to this weather station compared to an imported one which is about $8,000 (Eight Thousand Dollar)

TABLE OF CONTENTS

CHAPTER THREE

CHAPTER FOUR

CHAPTER FIVE

LIST OF TABLES

LIST OF FIGURES

LIST OF PLATES

ABBREVIATTION

ANN	Artificial Neural Network
RF	Radio Frequency
LCD	Liquid Crystal Display
SMS	Short Message Service
BASCOM	Besic Compiler
LM35	Linear Monolithic 35
DHT	Digital Humidity and Temperature
GSM	Global System for Mobile Communication
DHT22	Digital Humidity and Temperature 22
USB	Universal serial Bus
URM	Advance Risc Machine
ULN	Ultra-low Noise
Wi-Fi	Wireless Fidelity
GH2	Gigahertz 2

DC	Direct Current
SD	
EEPROM	Electrically Erasable Programmable Read-only Memory
PEM	Pulse Widths Modulation
CSP	Chips- Scale Package
12C	Inter- Integrated Circuit
BMP	Barometric Pressure
GPIO	General Purpose Input and Output
UART	Universal Asynchronous Receiver Transmitter
IOT	Internet of Things
SOC	System- On-Site-Chip
SPI	Serial Peripheral Interface
LED	Light Emmiting Diode

CHARPTER ONE

1.0 INTRODUCTION

1.1 Background to the Study

Weather tracking is of great importance in our society today, in the past weather monitoring was only useful for agricultural purpose. But with recent advancement in technology, weather information is of great importance in other field of life (Jahedul *et al.,* 2018).

Weather monitoring checks the behaviour of some atmospheric parameters such as temperature, relative humidity, solar radiation, wind speed, wind direction, due point, rain fall and cloudiness (Ukhurebor*et al.*, 2017). Global warming may also lead to weakened and failed states, creating yet more poverty, forced migrations and resource scarcity-conditions that foster extremism and terrorism (www.ucsusa.org/ smartclimatchoice 2009).

The importance of weather tracking cannot be over emphasised, this is due to it great importance in agriculture, communications, air transportation, industries, meteorologist and to atmospheric physicist (Anthony *et al*., 2017).In agriculture, weather is one factor which determines how much success that can be recorded, because crop yield depend on weather in providing energy and water for it sustenance (Abubakar and Sulaiman, 2018). Research has proved that climate change will have great effect on agricultural yield and human activities in the next century(Walthall *et al*., 2018).Hence there is need to have an effective means of monitoring atmospheric parameter at all times. Such a weather station should have an independent power source such as solar energy, for all time weather monitoring. A microcontroller weather station reads and records atmospheric parameters using sensors without any external intervention. The recorded atmospheric parameters can be processed as wired information, in which case, it can be downloaded in a computer or through a server as in wireless communication (Ughanze *et al.,* 2019).

1.2 Statement of the Research Problem

The importance of weather station cannot be over emphasised. Hitherto, the source of atmospheric information is majorly from conventional weather stations, though available in limited numbers, some difficulties are associated with their operations. Some of the challenges associated in them are that:

Real time assessment of atmospheric parameters from conventional weather station for out-of-site users is not achievable, this is because they are not internet driven. Conventional weather station is prone to human error because it requires human presence due to its manual operation. Cost of acquiring conventional weather station is often on the high side. Conventional weather stations are scares to see around due to its size. Malfunctioning of components in existing conventional weather station due to it obsolete nature, leaves one with the option of combining readings from more than one weather station to get reasonable weather parameters of interest. Conventional weather stations are not portable and as such, cannot be moved to different locations for the purpose of gathering atmospheric parameters. To address these challenges, a solar powered microcontroller weather station was conceived. It utilise electronic sensors which will read and record atmospheric parameters, display such data on an LCD for on-site users and also transmit to an online server which will display real time atmospheric data to off-sites users. Such weather station will be portable and solar powered for round the clock acquisition of weather information of interest at any location.

1.3 Aim and Objectives of the Study

The aim of this work is to design and construct a portable solar-powered microcontroller-based weather station that will transmit real time atmospheric parameter to an online server.

This aim will be achieved through the following objectives, which are to:

i. design an electronic circuit using a microcontroller as it central processing unit which will retrieve and record atmospheric temperature, relative humidity, atmospheric pressure, wind direction and wind speed from respective electronic sensors and display the values for on-site users.

ii. interface a Wi-Fi module into the design to serve as a transmission link between acquired atmospheric data and a webpage for use for-off-site users.

iii. utilize miniature, low-cost and locally available components to construct the design, using solar energy as its source of power supply.

iv. evaluate theperformance of the constructed weather station if it meets design objectives.

1.4 Significance/Justification of the study

The study has the following significance:

i. the microcontroller-based weather station is of great benefit to atmospheric physicists and meteorologist under taking research studies, since the weather station provides the needed weather data for atmospheric studies.

ii. weather information that the microcontroller-based weather station provides is of great benefit to the agricultural sector, aviation sector, military operation, to mention just a few for planning and safety.
iii. the wireless and internet driven nature of the microcontroller-based weather station makes it possible for measured atmospheric parameter to be available online to a wider user.
iv. the use of solar energy as a source of power supply for the weather station accounts for the resourceful nature of the study in that it utilise the abundant solar energy available in the study area

1.5 Scope and Limitation of the Study

This work develops a solar powered microcontroller-based weather station that will measure atmospheric temperature, pressure, relative humidity, wind speed and wind direction with the aid of electronic sensors at sixty seconds interval. The weather station is solar powered in order to harness the abundant solar energy of study area. Also the atmospheric data measured will be displayed on a web site for easy accessibility by out-of-site users.

The limitation of the study is that, the measured parameters are limited to those of atmospheric temperature, relative humidity, atmospheric pressure, wind speed and wind direction only. The weather station can be upgraded in future to incorporate more atmospheric parameters

1.6 Justification of the Study

International professional bodies such as The International Oceanic and Atmospheric Administration, International Center for Environmental Information and World Metrological Organisation advocates for local collection of data. This requires the deployment of weather stations to locations. This work makes available low cost and portable weather station that could be easily installed in various locations.

Atmospheric data acquisition is crucial to the study of atmospheric physics, metrology and aviation. The weather station to be designed and constructed in this work is capable of real time digital data acquisition on-spot and online for purposes of record keeping and further analysis by professionals and researchers.

CHARPTER TWO

2.0 LITERATURE REVIEW

1.1 Elements of Weather

Weather is the state of the atmosphere, to the degree that it is hot or cold, wet or dry, calm or stormy, clear or cloudy (Karthik*et al*.,2015). Elementsof weather accounts for these regular changes in atmospheric condition. Some elements of weather are here presented.

2.1.1 Temperature

Temperature is an objective comparative measure of hotness or coldness of a place or a body (Dipak and Ajij, 2016). Celsius is the unit of measuring temperature. Temperature can also be measured in Fahrenheit (^oF) and Kelvin (^oK). The conversion of these units to Celsius is expressed mathematically as shown in equation (2.1) and (2.2)

$$1^0K = 273 + {}^oC \quad (2.1)$$

$$\text{Or } 1^oF = 32 + \frac{9}{2}{}^oC \quad (2.2)$$

Temperature is measured using an instrument calledthermometer.Most thermometers are closed glass tubes. Figure 2.1 shows a thermometer.

Figure 2.1: Conventional Thermometer

2.1.2 Relative humidity

Humidity is the measure of the amount of water vapour in the atmosphere. This amount varies from 0 to 4 %. There are mainly two ways of expressing humidity, that is absolute and relative humidity.

Absolute Humidity is the total amount of water vapour present in the atmosphere per volume of air at a definite temperature. While Relative Humidity is the ratio of the amount of water vapour actually in the air to the maximum amount of water vapour that the air can actually hold. Relative humidity is more useful when expressed as a percentage (Ibrahim, 2018).

$$R.H = \frac{\text{amount of water vapour actually in the air}}{\text{maximum amount of water vapour that the air can actually hold}} \times \frac{100}{1} \qquad (2.3)$$

Hygrometer is the instrument used to measure the water vapor content in the air or humidity. Figure 2.2 shows a conventional Hygrometer

2.2 A Conventional Hygrometer

2.1.3 Atmospheric pressure

Atmospheric pressure is the force per unit area exerted on a surface by the weight of air above that surface in the atmosphere of earth.

The force over the one centimeter is a pressure of 10.1N/m^2

1 Millibar= 1 hector Pascal. At sea level = 101325 Pascal.

The effect of atmospheric pressure is a vital factor in determining relative air pressure along different altitude in a surrounding (Hamid and Alaa, 2015).

A barometer is an instrument used for the measurement of atmospheric pressure. Figure 2.3 shows a conventional Barometer.

Figure 2.3: A Conventional Barometer

2.1.4 Wind direction

Wind is the perceptible natural movement of air in the form of current in a particular direction (Devaraju *et al.,* 2015). While wind direction is an angle measured in clock wise direction between true north direction and direction of the wind movement

(Devaraju *et al.,* 2015).A wind vane is an instrument used in determining the direction of the wind. Figure 2.4 shows a conventional wind vane.

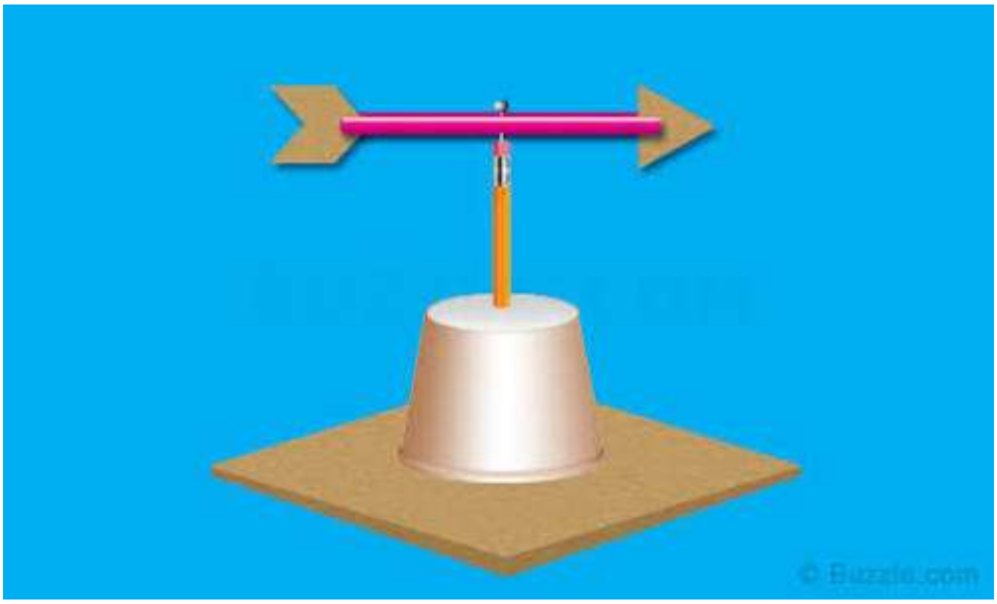


Figure 2.4 A conventional wind vane.

It has a pointer such that, for a sufficiently strong wind, the arrow-head of the pointer points in the direction of the wind.

2.1.5 Wind speed

Wind speed is the speed at which air particles move in the atmosphere measured in kilometer per hour (Kmph) (Devaraju *et al.,* 2015). Wind speed is a fundamental atmospheric quantity caused by air moving from high to low pressure due to changes in temperature (Ibrahim, 2018). Wind speed affect weather forecasting, aviation and maritime operation. A prominent factor affecting wind speed is the pressure gradient.

The pressure gradient is the measure of how much the air pressure changes with distance. It is expressed mathematically as;

$$f = \frac{(P_2 - P_1)}{l} \quad (2.4)$$

Where $(P_2 - P_1)$ is the pressure difference between two points, l is the distance between both points and f is the force that results from the pressure gradient. This force starts the movement of the air and the higher the pressure gradient, the stronger or faster the wind will blow. (Ibrahim, 2018). An Anemometer is the tool used to measure wind speed. It consists of a vertical stand and either three or four concave cups which captures the moving air particles. Figure 2.5 shows a conventional anemometer.

Figure 2.5: An Anemometer

2.1.6 Rainfall

Rain is the liquid water inform of droplet that have condense from atmospheric water vapor and fall under gravity (Dipak and Ajij, 2016). While the amount of water falling in rain, snow, mist and the likes within a given time and area is called rainfall.

The instrument used to measure the amount of rain that has fallen over a specific time is known as rain gauge. Figure 2.6 shows a conventional rain gauge.

Figure 2.6: A conventional Rain Guage

2.1.7 Cloudiness

Cloudiness is an occurrence visible with complex physical processes in the atmosphere (https.//ams.confex.com.)

Ceilometer is the instrument used to measure cloudiness. Figure 2.7 shows a ceilometer

Figure 2.7 A Ceilometer

2.1.8 Solar radiation

Solar radiation is an instantaneous energy emitted by the sun in the form of electromagnetic radiation that reaches the earth surface. It is measured in watt per meter square (Wm^{-2}), (Devaraju *et al.*, 2014).Pyrometer is used to measure global solar radiation falling on the earth. The instrument is shown in Figure 2.8.

Figure 2.8 A Pyrometer

2.2 Weather Monitoring

Weather monitoring and tracking is as old as the existence of mankind, this is because over the year's man has used different theories to predict cloud formation, rainfall, lighting, thunder, hotness and coldness of place and other weather parameters. But all prediction could not stand because they were full of errors. Therefore, the need to have accurate information about weather necessitated the invention of simple weather measuring instruments and weather stations. Two method of approach are used in weather monitoring. These are conventional weather monitoring approach and digital weather monitoring approach.

2.2.1 Conventional Weather Monitoring Approach

This is the use of conventional weather stations having individual simple instruments which reads and records atmospheric parameters, after which human effort is required for data collection and analysis. (Beth and Kong, 2017). Acquiring of atmospheric parameter involves human going to base station to take readings for individual measuring instruments, hence the accuracy of such atmospheric data is highly dependent on the person taking the reading. This is because one might wrongly record a particular value and this will affect the integrity of the weather information (Dipak and Ajij, 2016).

2.2.2 Digital weather monitoring approach

This involves automatic weather stations that has instruments that measures and records atmospheric parameters using sensors without human intervention, the measured atmospheric parameter can be stored in data logger or can be transmitted to remote location via communication link (Weerasinghe *et a.*, 2011). Electronic sensors are used to read and record atmospheric parameters and transmit wirelessly to an online server for out of site users to access using any electronic device that has internet connectivity.

2.3 Review of Related Works

Researchers have worked on microcontroller-weather based station. For example, Iswanto and Helman (2012), worked on a weather monitoring station with remote radio frequency wireless communications. The device has a microcontroller as the central processing unit which receives output from the sensors, process into useful information for analysis. An LCD is connected to the microcontroller to display acquired and processed atmospheric parameter. The device utilise radio frequency module KYL-1020 as a transmission link between the microcontroller and a personal computer for display of atmospheric data at a remote location. The C-computer language was used to programme the microcontroller. Materials used in the device are: LCD for displaying measured atmospheric parameters for on-site users, LM35 sensor for sensing temperature, light dependent resistor sensor for measuring light intensity, anemometer for monitoring wind speed, RF KYL1020v module for atmospheric data transmission and ATmega8535 microcontroller. The device provided real time atmospheric parameter, which were displayed on the LCD and also transmitted to a personal computer at a remote location, for display of atmospheric parameters in graphical format**.** However, the device utilised only analogue sensors, it is not web based and solar energy was not utilised as its source of power supply.

Also, Tajadinn and Abdelrasoul (2015), designed and implemented a real time remote measurement and monitoring weather parameter system. The system utilised a microcontroller to convert analogue output of the sensors into digital format and process into useful format. A GSM module is interfaced with the microcontroller for the purpose of transmitting acquired and processed atmosphere parameter to a cell phone via SMS.BASCOM programming language was used for the microcontroller operations. Components used: LM35 sensor for measuring temperature, anemometer for monitoring atmosphere wind speed, barometric sensor for measuring atmospheric pressure, DHT11 sensor for reading and recording relative humidity, ATmega 32 microcontroller for processing output of the sensor and a real time clock (RTC) which keeps time and date of acquiring and logging of data. The device transmits real time atmospheric parameters to a cell phone via SMS. The system is not web based and it did not utilised renewable energy as it source of power supply.

Furthermore, Goudal *et al.* (2014) designed and implement a microcontroller based real time weather monitoring device with GSM. The device monitored real time atmosphere temperature, relative humidity, pressure and dew point was calculated from the acquired parameter. Microcontroller serves as the central processing unit which receives analogue and digital signals from the sensors, process into useful format for analysis and achieving. An LCD is connected to the input pins of microcontroller for displaying of measured and process atmospheric parameter GSM module was interfaced with the microcontroller for the purpose of remote transmission of processed atmospheric parameter to a personal computer through RS-232 port. Components utilised in the system are: SHT11 sensor for sensing temperature and relative humidity, pressure transducer sensor for monitoring atmospheric pressure, LCD for displaying measured atmospheric parameters, PIC16F877A microcontroller for reception, conversion and processing atmospheric parameters from the sensors. GSM module for transmission of atmospheric parameters to the receivers end. C language was used to programme the microcontroller. The device provided real time atmospheric information and functions well when simulated and provided atmospheric parameters in numeric form. The weather station is not web based and does not utilise renewable energy as it source of power supply.

Karishma *et al.* (2016) presented a weather monitoring system using microcontroller. The system measured atmospheric temperature and relative humidity using analogue and digital sensors. The output of the sensor is analog signal and as such an analogue digital converter (ADC) was utilised to interface it with the input pin of the microcontroller. The microcontroller processes the output of (ADC) into useful format for analysis and archiving. An LCD is interfaced with microcontroller for displaying of measured and processed atmospheric parameters. A GSM module is connected to the microcontroller input pins for the purpose of transmitting processed atmospheric parameters to a cell phone via SMS. The following materials where used to develop the system: LM35 temperature sensor, LCD, GSM module and 89V51RD2 microcontroller. The weather station worked according to design, real time atmospheric parameters were displayed on the LCD for on the site- users and where transmitted to a cell phone via SMS. The device is not web base and did not utilise renewable energy as it source of power supply.

Abubakar and Sulaiman (2018**),** implemented a microcontroller based mobile weather monitoring system which reads and records atmospheric temperature, relative humidity, light intensity, wind speed and atmospheric pressure. ATmega38 microcontroller served as the central processing unit, it input pins were connected to the output of the atmospheric sensors. The microcontroller receives analogue and digital signals from the sensors as input, process it into useful format, store and display on the LCD. External memory was interfaced with the microcontroller. The device has GSM module interfaced with the microcontroller for remote transmission of atmospheric parameters via SMS. The acquired and processed atmospheric parameters can also be viewed remotely on a personal computer which is interfaced with the microcontroller. Components used: DHT11 sensor was used for reading and recording atmospheric temperature and relative humidity, light dependent resistor measured light intensity, MPX51000AP sensor sensed atmospheric pressure, anemometer was used to monitor the wind speed, memory card was used for data logging real time clock (RTC) was used for keeping time and data logging and ATmega 38 microcontroller which serves as the central processing unit. The device was tested and it worked according to design. Real-time atmospheric parameters were obtained from the device in numeric format and were compared with standard and existing conventional weather station, low percentage error were recorded. The device did not make provision for transmission of atmospheric data to server which be displayed on a website and did not utilised renewable energy as source of power supply to achieve round the clock weather monitoring.

Also, Karthik *et al*. (2015), worked on Arduino based weather monitoring system. The device reads and records atmospheric temperature, relative humidity, light intensity, heat index and dew point. An Arduino microcontroller uses it input pins to retrieve the output signal from the sensors. The microcontroller processes the acquired atmospheric data into useful format, display on the LCD and also transfer to a personal computer through serial port and stored as text file which can be moved to an excel file using macro. The system utilised the following components: LM35 and DHT11 sensor for temperature measurement, light dependent resistor for measuring intensity of light and BMP180 sensor for sensing atmospheric pressure. All the modules used in the design where tested and functions according to design specification. Real time atmospheric parameters where displayed on the LCD and on a personal computer at a remote location. The atmospheric parameters displayed on the personal computer where in graphical representation which indicate the weather pattern. Atmospheric parameters acquired by the weather station is not transmitted to an online server for wider accessibility of the weather information and device did not utilised renewable energy as its source of power supply for all atmospheric parameters acquired.

Emmanuel *et al***.** (2018) implemented an Arduino Uno microcontroller weather station to sense atmospheric temperature, relative humidity and light intensity. The system utilisedATmega328P microcontroller as the central processing unit which receives analogue and digital signals from the sensors output, process into useful format and display on the LCD for on the- site users. For out of site users, the microcontroller is interfaced with a GSM module to transmit acquired and processed atmospheric parameters to a cell phone via SMS.LM135 sensor is used to sense atmospheric temperature DHT22 sensor is used for relative humidity measurement, light dependent resistor for sensing light intensity, LCD used to display acquired and processed atmospheric parameters, DS3231 real time clock was utilised to keep time and date, LM2596 buck converter was used as step-down switching regulation. Bread board and digital multi meter were employed to validate the functionality of the components and they worked according to design, real time atmospheric parameters were displayed on the LCD for on the site and represented on a graphical format when viewed on a personal computer at a remote location. 0.6^0 C average error was observed in the temperature readings when compared to the conventional weather station temperature readings.1% error was observed for relative humidity reading when compared with conventional weather station relative humidity reading. The weather station is not web based and did not utilise renewable energy as it sources of power supply.

Also, Nhiverkar and Mudhokolker (2011) developed a data logger and remote monitoring system for multiple parameter measurement application. The device utilised LM35 sensor to read and record atmospheric temperature and SY-HS 220 sensor to sense relative humidity. The output of the sensors serves as input to ATmega32 microcontroller. The microcontroller converts analogue output of the sensors into digital signal, process it into useful format, display on the LCD and log it into a multimedia memory card which is interfaced to the microcontroller. The device has real time clock for time and date of the system's operation, RS-232 serial link is used to transfer logged atmospheres parameters to a personal computer in a data file using graphical user interface programme. Real time atmospheres parameters were displayed on LCD for the on-site users, at a remote location, Real time atmospheric parameters were also displayed on a personal computer in a graphical and numerical format. Values of the atmospheres parameters measured by the proposed weather station had some deviation when compared with atmospheres parameter values obtained from conventional weather station, for temperature 0.6^0C error was observed and for relative humidity 2% error was observed. The device is not web based, it utilised analogue sensors for the acquisition of temperature data and relative humidity data, this contributed largely to the error margin. The device did not utilise renewable energy as it source of power supply.

Similarly, Kamarul *et al.* (2006) developed a low cost microcontroller based weather monitoring system. The station has LM35 sensor for temperature measurement HSP15A resistive-based humidity sensor for sensing relative humidity and MPX4115 barometric pressure sensor for measuring atmospheric pressure. PIC16F877A microcontroller was utilized in retrieving analogue signals from the sensors, convert it into digital signal, process into useful format and display on the LCD. The processed atmospheric data is logged into a data logger for analysis and achieving. Real time clock chip is interfaced with the microcontroller for time and date keeping. A USB is used to transfer acquired atmospheres parameters to a personal computer. Real time atmospheres parameters were displayed on the LCD. Acquired atmospheric parameters are downloaded on a personal computer in numerical and graphical format with the aid of a USB. Analogue sensors were mainly utilised. The device is not web based and did not utilise renewable energy as it source of power supply.

Shewale and Gaikwad (2017), developed an internet of things based real time weather monitoring system using Raspbery pi. The system employed the following components: DHT11 sensor to measure temperature and relative humidity, light dependent resistor sensor to sense light intensity, BPM180 sensor for atmospheric pressure measurement, ULN 280, marked scale sensor to measure rain water level, general purpose input-output header board and ARM 11 Raspberry pi which serves as the central processing unit. The device used Raspberry pi to retrieve the output of all the sensors with aid of general purpose input-output header board. Raspberry pi is also utilised as a web server using hypertext transfer protocol communication application for transmission and display of processed atmospheric data on the webpage. Real time atmospheric data were displayed on thingsspeaks.com website using graphical format and one channel. The device did not have built in Wi-Fi and built in real time clock, for networking direct network has to be given to the device and also the source of power supply does not provide all day acquisition of atmospheric parameters

Furthermore, Rafi *et al*. (2018) worked on Arduino based weather monitoring telemetry system. The weather station has two main section namely; transmitter and receiver section. At the transmitter DHT11 sensor will sense atmospheric temperature and relative humidity, BMP180 sensor measures atmospheric pressure and light dependent resistor sensor measures light intensity. The outputs of the sensors are retrieved by ATmega328P Arduino microcontroller, processed into useful format and transmit via NRF24101 2.4 GHz. At the receiver section, the transmitted data is received by Arduino UNO R3 with the aid of NRF24 L01 module. The received atmospheric data is processed by the Arduino UNO R3 logged and displayed on a personal computer with the aid of a graphical interface. The device provided real-time atmospheric data which is transmitted to out of site users using radio-frequency and displayed on a personal computer at a remote location. The weather station can only transmit atmospheric parameters within a range of thirty-two meters and did not utilise solar energy as it source of power supply.

Edgar and Juan (2015), worked on a portable weather system for measuring and monitoring temperature, relative humidity and pressure based on Bluetooth communication. The system utilised these components: SHT11 click sensor for temperature and relative humidity measurement altitude click MPL 3115A2 sensor for sensing atmospheric pressure and PIC32 microcontroller with micro c program for reception and processing analogue and digital sensor signals. The weather station has different sections such as main control section which has the task of initialising and prioritizing of processor task, weather control section which calibrate control and compute relative humidity, temperature and atmospheric pressure, the Bluetooth control which is responsible for transmission of acquired and processed atmospheric parameters by the device to any Bluetooth device within the range of connectivity it established. The device took readings of atmospheric parameters after every one minute and store, an average was calculated after 10 minutes, after two hours calculated average of the atmospheric parameters were transmitted to a personal computer via Bluetooth for display and storage. The device does not have server installation to enable its upload atmospheric parameters to a dedicated website. Bluetooth technology works in peer and as such acquired atmospheric parameter from this device limited to few number of people, the device did not utilise renewable energy as its source of power supply.

Odeyemi and Bakare (2015), designed and implemented an improved wind speed meter (Anemometer), which utilise the following components PVC pipe for housing of the control panel compact disc in which the hard disc motor is mounted on for continuous rotation by the locally assembled three cup anemometer. A 9V battery is utilised for Dc voltage source, plastic table tennis eggs was used to construct the anemometer cups and a galvanometer which is used to read the output generated by the hard disc motor and display as wind speed.

Lm 2917 IC was utilised for frequency to voltage converter. This device was able to achieve linearity in its output with accuracy irrespective of the battery level. The device cannot measure speed in the range of 10km/hr. to 25km/hr. the values of the speed cannot be displayed digitally.

Also Mishu *et al.* (2019) developed a microcontroller based portable anemometer for wind monitoring system. The equipment utilised the following components: a three cup anemometer which reads and records wind speed and wind direction with the aid of opt coupler and potentiometer respectively. The calculated wind speed and direction is transmitted to a PIC16F887microcontroller via infrared ray light emitting diode. The microcontroller process the retrieved signal and display on a 16 x 2 liquid crystal display.

The device provides wind speed and direction values on the LCD and it was also tabulated and compared with standard values. The device could not give real time weather information for out of site user and the device could not achieve round the clock reading and recording of atmospheric parameters.

Subair and Abraham (2014), developed an intelligent pressure measuring system which reads an records pressure in the space at different altitude. The device utilise Bmp 180 sensor, PIC16F877A microcontroller basically. The barometric sensor monitors atmospheric pressure and transmits to the PIC16F877A which then converts the analog signal into a useful format and stored in flash chip. The stored data is transferred to a computer with the aid of Rs 232 interface and displayed on a lab view. The equipment is designed to read and record only atmospheric pressure, it is not internet driven and does not utilise renewable energy as it source of energy for round the clock monitoring of pressure data.

Naiz *et al.* (2014) designed and constructed a digital anemometer. The weather station made use of reed switch anemometer sensor for monitoring wind speed andATmega 323 microcontroller for retrieval of the wind speed signal and processing into useful format which is displayed on the LCD for on-site users. The device read and recorded only wind speed, wind speed data is made available for on-site users only. The weather station made use of battery as it source of power supply. The device is not internet driven and this limit the number of persons that can access it data on real time basis.

Similarly, Yahya and Khraisat (2012) developed a wireless meteorological station which utilised LM35 sensors to need and record atmospheric temperature. Three cup anemometer was used to monitor wind speed, while wind vane monitor wind direction. All the output signals from this sensor is retrieved by PIC-based 16F877A microcontroller and processed into useful format and display on an LCD for on-site user. An antenna is used to transmit this atmospheric data for out of site user using graphical user interface. The device can only transmit atmospheric data within the range where the antenna has it coverage and as such the only few persons can access data from the device. The meteorological station did not utilise renewable energy as it source of power supply for round the clock monitoring of weather parameter.

Wearasinghe *et al.,* (2011) worked on automatic weather station. The weather station measured temperature with aid of a semi-conductor temperature sensor and a non-inverting amplifier, honey well HIH 4000 sensor measured relative humidity and an anemometer with optical sensors and a rotary encoder measured wind speed, a wind vane containing rotary potentiometer sensor measured wind direction, barometric pressure sensor monitored pressure and a tipping bucket with 200mm diameter collecting funnel measured the amount of rainfall, the output of the sensor is retrieved by PIC18F4 microcontroller. The microcontroller processed the output of the sensors stored in a micro SD card and transmit to a central station through SMS. The weather station provided atmospheric parameters which was downloaded from an SD on a computer for the purpose of further analysis. The device did not made provision for LCD to enable on site users get weather information and the data transmitted was within a range of 100meter. The weather station did not utilise renewable energy as it source of power supply and retrieved atmospheric data is not uploaded to flat form that different out of site users can access.

Also Anuj *et al.* (2010) designed and developed a multiple channel data logger build up environment. The system measured and recorded atmospheric temperature, relative humidity, carbon (ii) oxide and carbon (iv). PIC18F4458 microcontroller was utilised to process the output of the sensors into useful format and display on the LCD. The processed atmospheric data is interface with real time clock and EEPROM using 1^2C communication protocol for out of site users, a max 232 is used to connect PIC18F4458 microcontroller to a computer for display of atmospheric data. Sensors used are: LM35 sensor which measured atmospheric temperature, HIH-400 sensor which sensed relative humidity and TGS4161 concentration sensor which Co and Co_2 gases measured and recorded atmospheric data were displayed on the LCD and on a personal computer for one user at a time. The device made use of analogue sensors and it is not web based which give access to multiple users to access atmospheric data from their comfort zone. The weather station did not utilise renewable energy as it source of power supply.

Also lay *et al.* (2014) developed a microcontroller based sensing unit in transmitters for wireless weather station. The following components where utilised: MPX 41 15A sensor was used for sensing atmospheric pressure, DHT11 sensor to measure atmospheric temperature and relative humidity, a potentiometer sensor for sensing wind direction. PIC18F452 microcontroller which was programmed with C-language was utilised as the central processing unit of the device. The sensors are connected to the microcontroller through it's input pins to retrieve the output of the sensors and process into useful format for analysis and archiving. An LCD was connected to the microcontroller to display processed atmospheric parameters for on-site users. The microcontroller transmits processed atmospheric data to a based station using radio frequency module at 433MHz within a range of 150m. The device displayed atmospheric parameters monitored for on-site users on the LCD and on a personal computer at the base station. The weather station could not transmit atmospheric data beyond 150 meters for out of site users for the purpose of multiple from these comfort zone, therefore atmospheric data could have been uploaded on a website. The sensor recorded some errors while reading atmospheric pressure, the weather did not utilise renewable energy as it source of power supply.

Also Sarmad and Forat (2017) designed a weather monitoring system using Arduino based data to measure atmospheric temperature, relative humidity, light intensity and wind speed. Arduino UNO microcontroller receives and process the outputs from sensors such as: DHT11 which measures atmospheric temperature and relative humidity, photocell sensor reads and records light intensity and an anemometer utilise DC motor as it sensor to measure wind speed. The Arduino retrieve process and display all the outputs of the atmospheric sensors on an LCD and create an excel data base file using R-language. The weather station does not have provision for uploading atmospheric data to the cloud for multiple users remotely. Also the device did not utilise renewable source of energy for it source of power supply.

2.4 Review of Major Components Used

2.4.1 Solar panel

A solar cell is the integral part that makes up a solar panel. The series and parallel connection of several solar cell forms a photovoltaic module. A single solar cell is shown in figure 2.9, two resistors, a diode and a current source can be utilised to represent a single solar cell.

This model is known as a single diode model of solar cell.

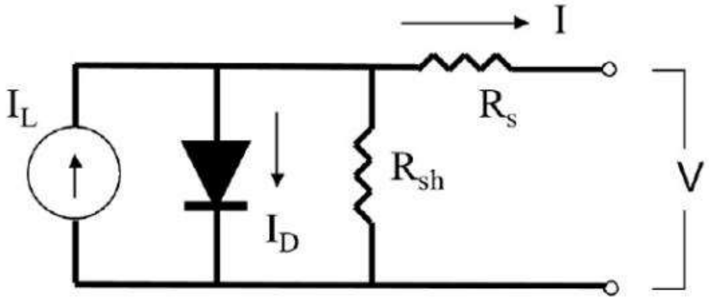


Figure 2.9: Single Diode Model of a Solar Cell (Ami, *et al.,* 2015)

The I-V characteristics equation of solar cell is represented below:

$$I = I_L - I_O \left\{exp\left[\frac{q(V+IRs)}{Ak\tau c}\right] - 1\right\} - \frac{V+IRs}{Rsh} \quad (2.5)$$

I_L is a light generated current or photo current (denoting the current source), I_O is the saturation current (representing the diode), Rs series resistance, A is diode ideality factor, k (= 1.38x 10^{-23} W/m^2K) is Boltzmanns constant, q (=1.6x10^{-19}C) is the magnitude of charge on an electron and T_C is working cell temperature and R_{sh} is the shunt resistor.

Photo current or light generated current, basically depends on the solar insulation and cell working temperature, which is represented below:

$$I_L = G[I_{SC} + K_i (T_C - T_{ref})] \quad (2.6)$$

Where I_{sc} is the short circuit current at 25°C and $1kW/m^2$, K_i is the short circuit current temperature coefficient, T_{ref} is the reference temperature and G is the solar insulation kW/m2.

While, Energy efficiency of a PV system is defined as the ratio of the output energy of the system (that is, electrical energy) to the input energy (that is, solar energy) received on photovoltaic surface. The energy efficiency of a PV system is given by

$$\text{Energy efficiency } \eta en = \frac{V_{oc} I_{sc}}{S} \quad (2.7)$$

Where S is the solar absorbed flux

Where V_{oc} = Open circuit voltage, I_{sc} = short circuit current

The energy efficiency of a PV system at maximum power is defined as the ratio of actual electrical output to input solar energy incident on PV surface area and it is given by

$$\eta el = \frac{V_{mp} I_{mp}}{S} \quad (2.8)$$

(Ami, *et al.*,2015)

The following factors are taken into consideration before the real sizing of the PV array begins:

The dc voltage of the system (V_{dc})

The average sun hours of the installation site per day (T_{sh})

The daily average energy demand in watt-hours (E_d)

Estimating the required daily average energy demand(E_{rd}) is the first step in sizing the array, this derived by dividing the daily average energy demand by the product of the efficiencies of all system components as given in equation (2.9).

$$E_{rd} = \frac{E_d}{\eta b \eta i \eta c} \quad (2.9)$$

Where ηb = battery efficiency

ηi = inverter efficiency

ηc = charge controller efficiency

The average peak power ($P_{ave,peak}$) is calculated by dividing the required daily average energy demand by the average sun hours of the location per day (T_{sh}) is represented in equation (2.10)

$$P_{ave,peak} = \frac{E_{rd}}{T_{sh}} \tag{2.10}$$

The total dc current of the system (I_{dc}) is calculated by dividing the average peak power by the dc voltage of the system is shown in equation (2.11).

$$I_{dc} = \frac{P_{ave.peak}}{V_{dc}} \tag{2.11}$$

The number of modules in series (Nsm) is computed by dividing the system dc voltage by the rated voltage of each module (Vrm) as represented in equation (2.12).

$$N_{sm} = \frac{V_{dc}}{V_{pm}} \tag{2.12}$$

To calculate the number of moduleinparallel (N_{pm}) we divide the total dc current of the system by the rated current of one module (I_{rm}) as in equation (2.13).

$$N_{pm} = \frac{I_{dc}}{I_{rm}} \tag{2.13}$$

The total number of modules (N_{tm}) that form the array is determined by the product of the number of modules in series to the number of modules in parallel as shown in equation (2.14).Therefore the needed array size is shown below(Guda and Aliyu, 2015):

$$N_{tm} = N_{sm} \, x \, N_{pm} \tag{2.14}$$

2.4.2 Battery

Two or more chemical cells connected in series makes up a battery. The combination of materials within a battery is used for the purpose of converting chemical energy into electrical energy (Jacob, *et al*., 2018). Deep cycle battery is utilised in the design for this device. This is because if it discharges to it minimum energy level it can be recharged faster within a short duration. Thus the battery will have large storage capacity to power all the load in the device even when there is known sun light (Saleh *et al.*, 2015). Estimated energy storage (E_{est}) of the battery is the first factor to be considered before sizing the battery required, this is derived by the product of the display average energy demand to the number of autonomy days (D_{aut}), this is mathematically expressed in equation (2.15).

$$E_{est} = E_d x\, D_{aut} \tag{2.15}$$

Safe energy storage (E_{safe}) is calculated by the ratio of obtained estimated energy storage to the maximum allowable depth of the discharge (D_{dish}). This is expressed in equation (2.16).

$$E_{safe} = E_{est}/D_{disch} \tag{2.16}$$

Total capacity of the battery bank in ampere-hours (C_{tb}) is calculated through the ratio of safe energy storage to the rated dc voltage of one battery (V_b). This shown in equation (2.17).

$$C_{tb} = \frac{E_{safe}}{V_b} \tag{2.17}$$

Therefore the total number of batteries (N_b) is computed by dividing the total capacity of the battery bank in ampere-hours by the capacity of one of the selectedbatteries in ampere-hours (C_b) as represented in equation (2.18).

$$N_{tb} = \frac{C_{tb}}{C_b} \quad (2.18)$$

The number of batteries in series (N_{sb}) is derived by the ratio of the system dc voltage to the rated dc voltage of one battery. The is shown in equation (2.19).(Guda and Aliyu, 2015).

$$N_{sb} = \frac{V_{dc}}{V_b} \quad (2.19)$$

2.4.3 Charge controller

Charge controller is a very vital device utilised in the connection between a solar panel and a battery for the purpose of regulating output voltage from the solar panel to the required charging voltage of the battery in use.

A charge controller acts as a switching device between the solar panel and the battery, the charge controller trips of the terminals of the battery when it is fully charge to prevent over charging and also the battery from draining when in use(Ughanze *et al.*, 2019). This enhances the battery performance, life span and reduces safety risk.

The total current capacity of a charge controller is the first factor to consider before making a choice of a suitable charge controller to be used. Therefore, if the current capacity is determined, then the total number of charge controllers will be computed and the cost charge controllers to be used will be estimated. This will lead to calculating the total cost of the charge controllers to be used.

Required Charge Controller Current (I_{rcc})

$$I_{rcc} = I^{M}_{sc} x N_{pm} \text{ x } F_{safe} \quad (2.20)$$

Where I^{M}_{sc} = the short circuit of the selected module

F_{safe}=safe factor, the safe factor is necessary in order to allow for a reasonable system expansion.

N_{pm} = number of modules in parallel

Number of Charge Controllers (N_{cc})

$$N_{cc} = I_{rcc}/I_{cc} \tag{2.21}$$

Where I_{cc} = charge controller current

Cost of Charge Controllers in Naira (C_{tcost})

$$C_{tcost} = N_{cc} \times C_{cost} \tag{2.22}$$

(Saleh *et al.*, 2015).

2.4.4 Voltage regulator

High voltage supply to electronic equipment or appliances poses serious risk which could lead to damage to some components in an electronic equipment. In order to protect this electronic device there is a need for a voltage regulator that will peg 5v output from the battery which is tolerated by constituent components of the device a voltage regulator is utilised.

A voltage regulator produces a fixed voltage output of a preset capacity irrespective of variations in it input voltage or load conditions (Mohammed and Sumit, 2015).

Voltage regulators come in different forms with different outputs, most available regulators are 5 volts, 9 volts and 12 volts. For a device with a maximum operational voltage of 5 volt + 5v voltage regulator is utilised. As indicated in Table 2.1, the last two digits in the name indicate the output voltage (Diarah *et al.*, 2014).

Table 2.1: Names and Output Voltage of Different Voltage Regulators

Name	Voltage
LM7805	+5
LM7809	+9
LM7812	+12
LM7905	-5
LM7909	-9
LM7912	-12

2.4.5 Microcontroller

An Arduino mega 2560 is a microcontroller board based on ATmega 2560. It has 54 digital input/output pins, 15 of the 54 pins is used as PWM output pins and 5V operated.ATmega 2560 has 4 UARTs hard ware serial ports and 16 MHz crystal oscillator, a USB connection, a power jack ICSP header and a reset button (Gauri*et al.*, 2017)

ATmega 2560 is supported with a full suit of program and system development tools such as C-compilers, micro assemblers, program debugger/simulators, in circuit emulators and evaluation kits. (Anusha*et al*., 2016), Figure 2.10 shows an Arduino Mega 2560

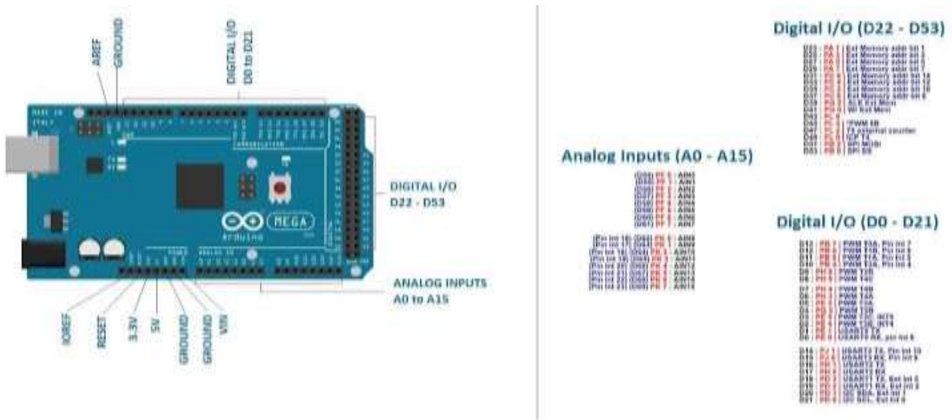


Figure 2.10: Arduino Mega 2560

2.4.6 DHT22 sensor

DHT22 sensor is utilised in the monitoring of temperature and relative humidity through data logging. DHT22 sensor essentially works with a thermistor and a capacitive sensor which gives digital output signal (Maulik and Irshad, 2018). DHT22 sensor is a calibrated digital sensor which senses atmospheric temperature and relative humidity and produce digital signal output. Stable and reliable. It has a wet components resistive sense and an NTC temperature measuring device, it also has low power consumption, 5V rated, quick response, transmit data over a long distance and utilisea single bus data communication protocol to interface with a microcontroller (Rafi *et al.*, 2018). Figure 2.11 shows a DHT22 sensor

.

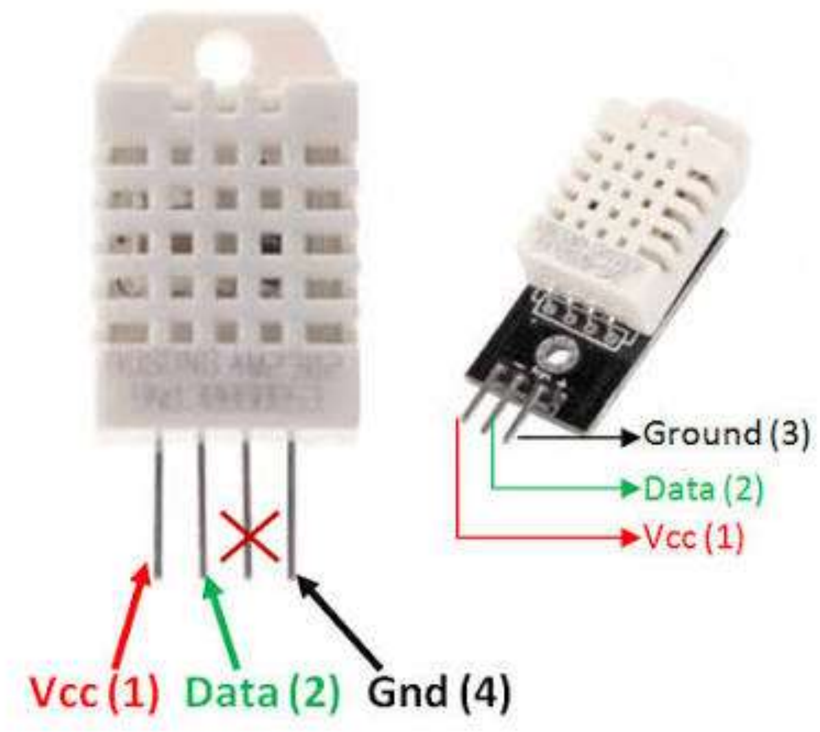


Figure 2.11 DHT22 sensor

2.4.7 BMP 180Sensor

BMP180 sensorhas a digital barometric pressure sensor of Bosch Sensotec, Analogy to Digital Converter,Inter Integrated Circuit (I^2c) communication, low voltage (3.3V) consumption and low attitude noise of about 0.2m. The sensor pressure data is compensated by a calibrated data of the EEPROM of the BMP180 (Poonam *et al*., 2015).

BMP180 can take samples of data for as fast as 128 samples per second. It is low cost and has a miniature size(Kirit*et al*., 2016). BMP180 is based on piezo-resistive technology for EMC robustness, high accuracy and linearity as well as long term stability and this makes it connectivity with a microcontroller of mobile device via the I^2c bus. BMP 180 has an operating pressure of 300hpa to 1100hpa and operating voltage of 3.3v (Subair and Abraham, 2014) Figure 2.12 shows a BMP180 sensor

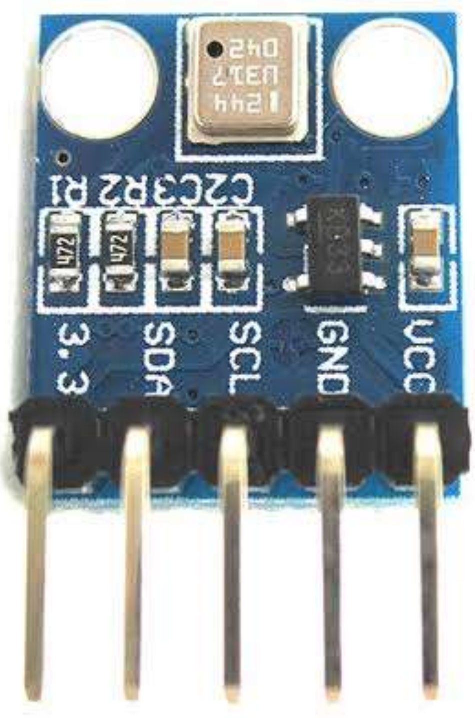


Figure 2.12 BMP180 sensor

2.4.8 Wind vane

A wind vane is a mechanically assembled device which measures wind direction with the aid of a resistive sensor embedded in it. Eight switched exist in the wind vane and each is connected to different resistor sensors. Weather rack calculates the resistance value that each resistor produces when calculating the voltage on the resistors. Wind vane often times reads and records values for 8 directions but many read up to 16 directions if when two contacts are closed at the same time (Beth and Kong, 2017). Figure 2.13 shows a constructed wind vane.

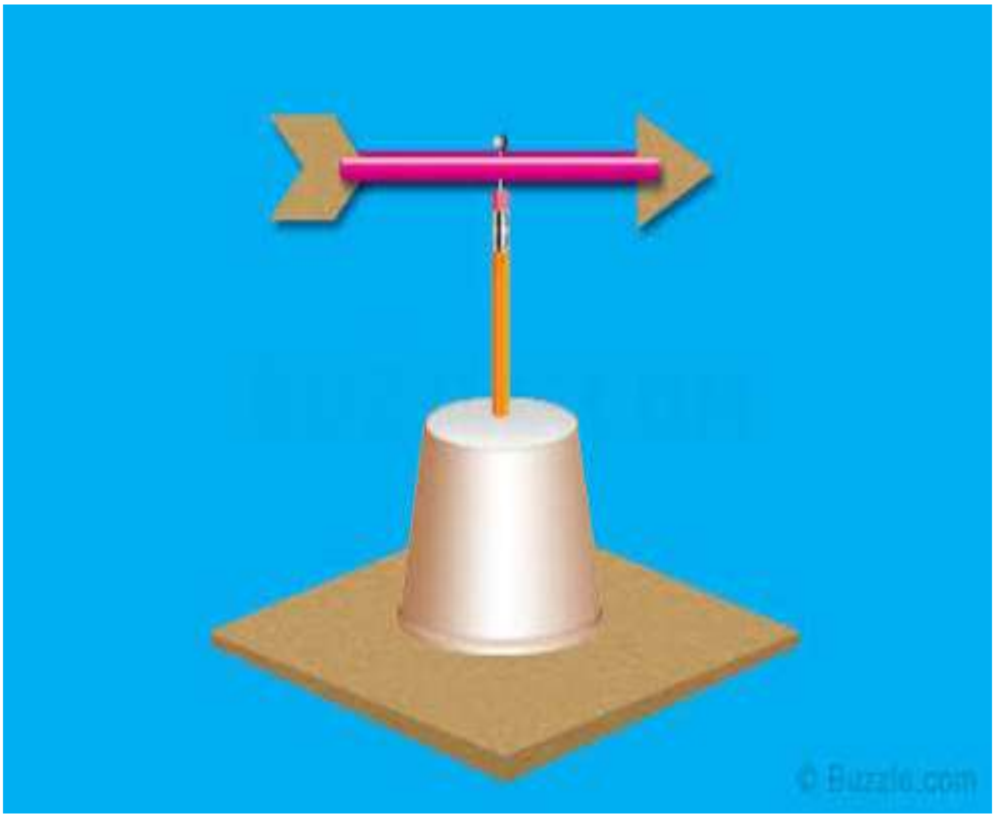


Figure 2.13 A Wind Vane

2.4.9 Anemometer

An Anemometer a mechanically assembled device which has three cup to calculate wind speed by calculating the rotation of the anemometer cups at a set interval. If usually have a speed sensor switch which is interfaced with a magnet, so that for each rotation the sensor switch will close and open. The magnet is usually placed on the shaft of the rotating cup symmetry and this enables the rotation of the three cups if there is effect of wind on it (Dipak and Ajij, 2016). Figure 2.14 shows a mechanically construct and anemometer

Figure 2.14An Anemometer

2.4.10 Liquid crystal display (LCD)

The display section has 84 by 48pixel LCD display with 8 input pins which are connected to the microcontroller through which measured atmospheric data are displayed as output. The LCD uses serial peripheral interface (SPI) communication protocol and has two registers namely: command and data (Emmanuel *et al.*,2018). Figure 2.15 shows the LCD

Figure 2.15: An LCD

2.4.11 ESP8266 Wi-Fi module

ESP8266 Wi-Fi module has TCP/IP protocol stack integrated on chip for the purpose of Wi-Fi network provision. ESP8266 is a programmed SOC and it communicates with a microcontroller through UART interface. It works with a supply of 3.3v. The module is configured with AT commands and the microcontroller is programmed to send at commands in a required sequence to configure the module in client mode. The module can be used as client and server modes. Once it gets connected in a Wi-Fi network, an IP address is established which can be accessible in its local networks. The module is has 2 GPIO pins alongside UART pins. It also has inbuilt SPI protocol by using the two pins of UART as data lines and by configuring the two GPIO pins as control lines and clock signal. It also 1MB on-chip flash memory, internal power management unit with all regulators and PLLs. The on-chip processor is a 32 bit CPU. (Kondamudi and Gupta, 2016). Figure 2.16 shows the picture of ESP8266 Wi-Fi module

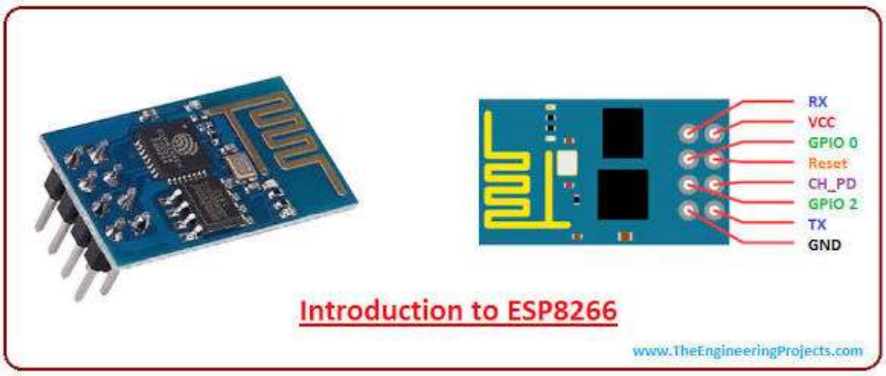


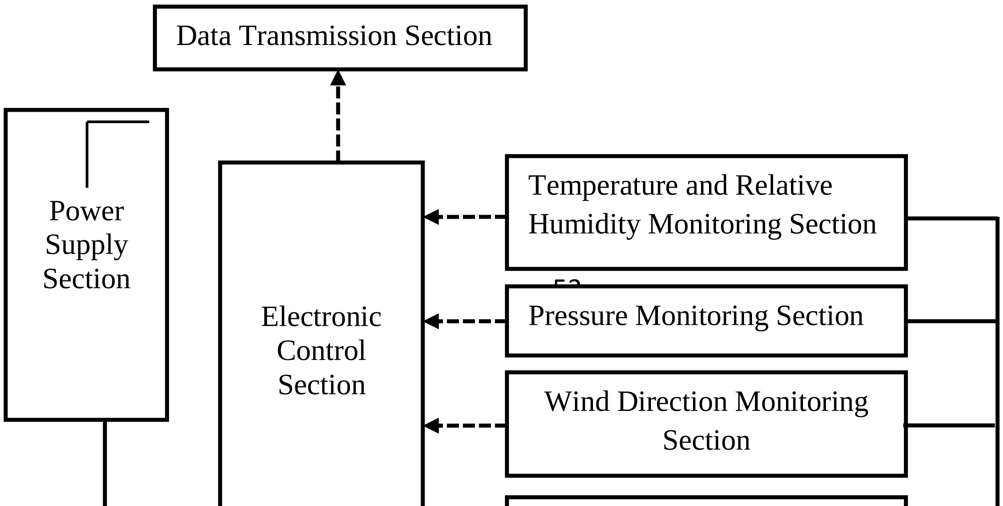
Figure 2.16 An ESP8266 Wi-Fi module

CHAPTER THREE

3.0 METHODOLOGY

3.1 System Design

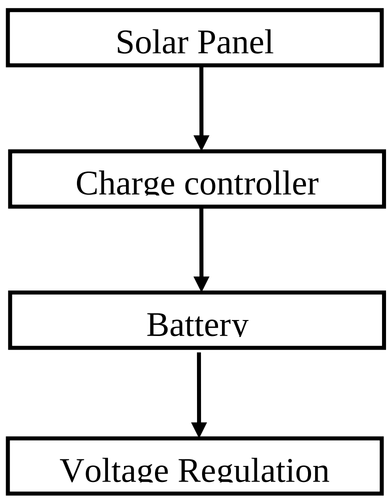
The block diagram formulated for the conceived solar powered microcontroller-based weather station is shown in Figure 3.1.It shows the signal flow through sections of the device.



The device consists of the following sections: power supply, electronic control and the temperature, relative humidity, pressure, wind speed and wind direction monitoring sections. Other sections include the data transmission and the display sections. Solid lines were used to show the power line while the broken lines indicate the direction of signal flow. The design description for each section is presented in the following subsections.

3.1.1. Power supply section

The power supply section provides the electrical energy needed by the weather station for round the clock monitoring of atmospheric parameters. Consequently, the weather station is powered using the abundant energy of the tropical African sun, that is, solar energy. Figure 3.2 shows the block diagram of the power supply section.



The power supply section comprises of the following subsections: solar panel, charge controller, battery and voltage regulation sub sections. Before diving into the design description of the section, it is necessary to have an idea of the overall power rating of the device. This will be instrumental in subsequent design regarding the subsections.

The overall power rating of the device is obtained from the summation of the individual power ratings of the major components used in the design. The weather station was designed to operate on 5V as its maximum voltage. The power rating of the device is determined as follows:

The microcontroller (ATmega 2566) used in this work is rated 50mA/5V, thus the power P, consumed by the component is calculated as:

$$\mathrm{P} = \mathrm{IV} \tag{3.1}$$

Where I and V are the current and voltage passing through the microcontroller. Substituting into equation (3.1) yield; 50mA x 5V = 0.025W

The microcontroller requires 0.025W to operate.

DHT22 sensor used is rated 40mA/5V, thus the DHT22 sensor power is calculated using (3.1) as;40mA x 5V = 0.02W.Hence, the DHT22 Sensor will require 0.02W power to operate.

The BMP180 Sensor used is rated 100μA/3.3V and as such its power is calculated using (3.1) as0.00033W.

Reed switch sensor for the anemometer is rated 1.2A/5V. Thus, the operating power is calculated using (3.1) as 6W.

Continuous variable resistor that will be utilised by the wind vane is rated 10Ω/5V. Thus, the operating power is calculated using the modified version of (3.1). that is;

$$P = V^2/R \tag{3.2}$$

(3.2) yield 2.5W.

Nokia 5110 LCD has a current and voltage rating of 80mA/3.3V. Thus, the power needed for its operation was calculated using (3.1) as 0.264W.

ESP 8266 Wi-Fi module has a current voltage rating of 80mA/3.3V, thus the required power for its operation was also calculated using (3.1) as 0.264W.

The total power rating is summarised in Table 3.1.

Table 3.1: Total load of the weather station

Name of component	Component Value	Formula	Computation	Power (W)
DHT 22 sensor:	40mA/5v	P=IV	40mA x 5V	0.02W
Bmp 180 sensor	$10\mu A/3.3V$	P=IV	$100\mu A\ x\ 3.3V$	0.00033W
Nokia 5110 LCD	80mA/3.3V	P=IV	80mA x 3.3V	0.264W
ESP8266 Wi-Fi Module	80mA/3.3V	P=IV	80mA x 3.3V	0.264W

Reed Switch Anemometer Sensor	1.25A/5V	P=IV	1.25A x 5V	6W
Continuous variable resistor wind vane sensor	10R/5V	$P=\frac{V^2}{R}$	5^2V/10R	2.5W
ATmega 2566 microcontroller	50mA/5V	P=IV	50mA x 5V	0.025W
Total				**9.0733W**

The total load of the solar powered microcontroller-based weather station is the summation of the power required by individual components considered. Therefore, 9.07W worth of power will be needed to run the Microcontroller-based weather station.

3.1.1.1Solar panel

The solar panel will convert the radiant energy of the sun into electrical energy. The choice of solar panel to use in terms of their number and electrical specifications is dependent on a series of calculations and estimation processes. To do this, location of deployment of the device is crucial. Nigeria usually experiences wet and dry weather; thus, average radiation shall be taken into account in this work. To compute the average peak power ($P_{ave,peak}$) extractible by a solar panel, recall equation (2.10), the required daily energy demand E_{rd} of $217.68\ Wh/day$ and the average solar radiation data in a day (that is the sum hour, T_{sh}) of 9 hours (Okonkwo and Nwokoye, 2014). The average peak power ($P_{ave,peak}$) is expressed as;

$$P_{ave\ peak} = \frac{E_{rd}}{T_{sh}} = \frac{217.68}{9} = 24.19W$$

This implies that a solar panel can sufficiently supply above the earlier computed 9.07W power demand of the device.

Total DC current $I_{dc}$is calculated using equation (2.10)

$$I_{dc} = \frac{P_{ave\ peak}}{V_{dc}} = \frac{24.19}{5} = 4.84A$$

This value is well above the current requirement for each component used in the design as shown in Table 3.1.

The number of modules in series (Nsm) is then determined by dividing the system dc voltage (battery voltage) by the rated maximum voltage of each module Vpm, (module to be used is rated 18V) as presented in equation (2.12).

Using equation (2.12) we calculate the number of modules in series*;*

$$N_{sm} = \frac{V_{dc}}{V_{pm}} = \frac{12}{18} = 0.0667 \text{ Approximately 1}$$

Next, we obtain the number of module in parallel (N_{pm}) by dividing the total dc current of the system (4.84 A, as earlier computed) by the rated current of one module I_{rm}, (maximum current of module used) as in equation (2.13).

$$N_{pm} = I_{dc}/I_{pm}$$

$$=\frac{4.84}{7.23} = 0.67 \text{ approximately 1}$$

The condition for both connections requires more than one module. Therefore, none of them can be implemented. Thus, a single module was used to intercept the radiant energy of the sun.

Table 3.2 shows the PV array sizing

Table 3.2: Required PV array

Parameters	**Formula**	**calculation**	**Computed value**
Required daily energy demand (E_{rd})	$E_{rd} = E_d$	217.68	217.68Wh/day

Average peak power	$P_{ave.peak} = \frac{E_{rd}}{T_{sh}}$	$\frac{217.68}{9}$	24.19W
Total Dc current (I_{dc})	$I_{dc} = \frac{P_{ave.peak}}{V_{dc}}$	$\frac{24.19}{5}$	4.84A
Number of series modules (N_{sm})	$N_{sm} = \frac{V_{dc}}{V_{pm}}$	$\frac{12}{18}$	0.67 approximately 1
Number of parallel modules	$N_{pm} = \frac{I_{dc}}{I_{pm}}$	$\frac{4.84}{7.23}$	0.67 approximately 1

From the result arrived at, a choice was made for a mono crystalline solar module by Rubitec solar HU130/12V to provide the voltage source to the weather station for continuous atmospheric data acquisition. The specification is shown in Table 3.3.

Table 3.3 Specification of rubitec solar panel type Hu130/12V

Specification	**Value**
Typical Power	(5%) 130 Watts
Current at typical power	

(Maximum power current) V_{pm}	7.23amps
Voltage of typical power	
(Maximum power voltage) I_{pm}	18.0 volt
Open circuit voltage (V_{oc})	21.5 volts
Short circuit current (I_{sc})	7.74amps
Irradiance (E)	$1000W/m^2$
Module Temperature (T_c)	25^0C

Source: Rubitec Solar Co., LTD

3.1.1.2 Charge controller

The charge controller acts as a switching device between the solar panel and the battery, the charge controller regulates the output voltage from the solar panel to the required charging voltage of the battery. The specification of the charge controller to use depends on the rating calculation of its current and voltage.

The charge controller current is obtained using equation (2.20). $I_{rec} = I_{sc} \times N_{pm} \times F_{sfe}$

$I_{sc}$is the short circuit current of the panel = 7.74A

$N_{pm}$is the number of parallel module = 1

T_{safe} is the safety factor of the charge controller = 1.25

$$I_{rcc} = 7.74 \times 1 \times 1.25 = 9.675A$$

9.675Acharge controller is not commercially available. Therefore, a 10A rated charge controller will be utilised. Thus, the specification will be $V_{dc} = 5V, I_{cc} = 10A$.

Number of charge controller is calculated using equation (2.21).

$N_{cc} = \frac{I_{rcc}}{I_{cc}} = \frac{9.675}{10} = 0.968,$ Approximately 1

Table 3.4 shows the rating of the charge controller

Table 3.4: Charge controller rating

Parameters	Formula	Calculation	Computed Value
Required charge controller current (I_{rcc})	$I_{rcc} = I_{sc} \times N_{pm} \times F_{safe}$	$7.74 \times 1 \times 1.25$	9.675A
Number of charge controller (N_{cc})	$N_{cc} = \frac{I_{rcc}}{I_{cc}}$	$\frac{9.675}{10}$	0.968 approximately 1

3.1.1.3 Battery

The battery is the power bank of the entire system. The electrical energy generated by the solar module is reserved in the battery. A 12V rechargeable deep cycle battery wasutilised as the DC voltage source to the device. Some of the battery characteristics are computed below:

The daily average energy demand in Watt-hour (Ed) is the product of the energy needed to operate the weather station (Ws) earlier determined as 9.07 W and the number of hours per day (hr).

Ed = 9.07W x 24hours = 217.68WL/day

To determine the estimated energy storage (E_{est}) in the battery, we recall equation (2.15);

$$E_{est} = E_d x\, D_{aut}$$

Daily average energy demand E_d was earlier calculated as 217.68WL/day and the number of autonomy days (D_{aut}) of one day. E_{est} is calculated as 217.68Wh

From equation (2.16) safe energy store is given as $E_{safe} = \frac{E_{est}}{D_{dis}}$

Where D_{dis} is the maximum depth of discharge of the battery chosen as 75%.

$$E_{safe} = \frac{217.68Wh}{0.75} = 290.24W$$

The total capacity of a battery bank is obtained by recalling equation (2.17) given as

$$C_{tb} = \frac{E_{safe}}{V_b}$$

WhereV_b is the voltage of the battery which is 12V.

$$C_{tb} = \frac{290.24}{12} = 24.19A$$

Table 3.5summarises the needed battery capacity.

Table 3.5: Summary of required battery capacity

Value of components	**Formula**	**Calculation**	**Computed value**
Daily average energy	$E_d = W_s \times hr$	9.07Wx24hours	217.68Wh Wh/day
Estimated energy stored (E_{est})	$E_{est} = E_d \times D_{aut}$	217.68x1	217.68 WhWh
Safe energy stored E_{safe}	$E_{safe} = \frac{E_{est}}{D_{dis}}$	$\frac{217.68Wh}{0.75}$	290.24W
Total capacity of the battery the battery bank (C_{tb})	$C_{tb} = \frac{E_{safe}}{V_b}$	$\frac{290.24}{12}$	24.19A

3.1.1.4 Voltage regulation

Voltage regulation is very vital for the device to peg the 12V battery output to a level tolerable to the constituent components of the weather station. A maximum voltage of 5V is sufficient and as such a LM7905 voltage regulator was utilised, C2 capacitor is interfaced with the voltage regulator to filter the 12V input voltage of the voltage regulator, C3 capacitor is also interfaced with voltage regulator to filter 5V output from the voltage regulator which then goes into the microcontroller and other components of the device which utilises 5V.

Thecircuit diagram of the power supply section is shown in Figure 3.3.

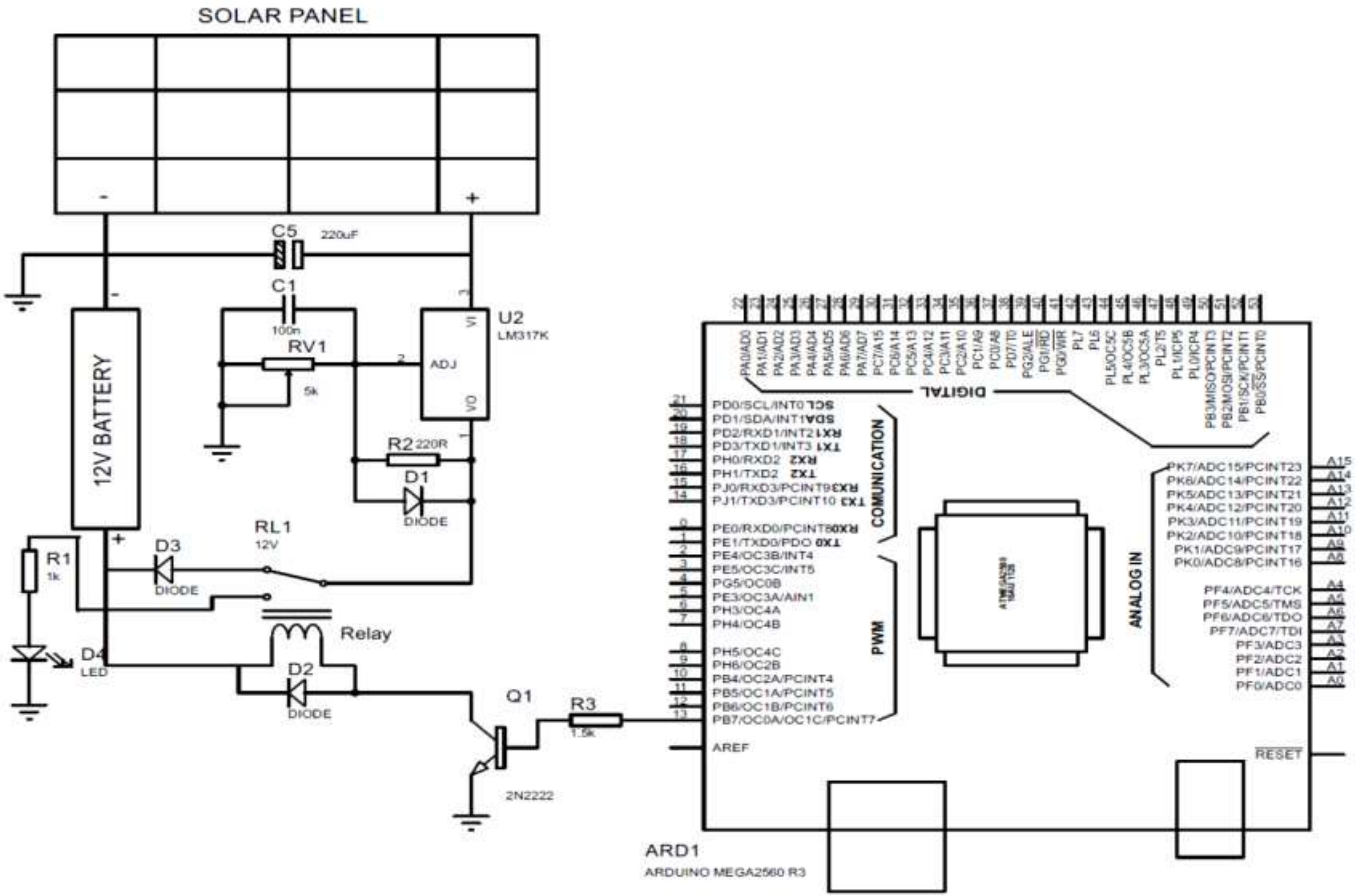


Figure 3.3: Circuit connections of the solar panel, battery and voltage regulator.

A 20W mono crystalline solar panel was utilised to convert light and heat energy into electrical energy, which result into DC voltage of 12V. From Figure 3.3 capacitor C5 is placed for the purpose of filtration. LM317K variable resistor is used to achieve this purpose, because it regulates the 18V DC voltage produced from the solar panel to the required charging 12V required by the battery. From figure 3.3, the diode D1 protects the LM317K against back supply and diode D3 prevent discharge of voltage from the battery when it is not charging due to absence of sunlight. The charge controller has a variable resistor RV_2 which measures the voltage level of the battery, the RV_2 is connected to pin PF0/ADC0 terminal on the microcontroller for the purpose of digital display of the battery level. To avoid over charging without being noticed. The interface between the charge controller and the microcontroller make it possible for PB7/OC0A/OC1C/PCINT7 at the microcontroller terminal to remain at zero if the battery is not fully charged, thus the transistor Q1 will remain off and the relay contact will be at the common (c) and normally closed (NC). Pin PF0/ADC0 terminal on the microcontroller is interfaced with charge controller in such away that it detects when the battery is fully charged, thus pin PB7/OC0A/OC1C/PCINT7 at the microcontroller terminal will move to 1 state and this trigger the transistor to on and also make the relay contact to be between the normally open (NO) and common (C). (i.e the relay will open). The charge controller relay energies and the battery stops charging, this is noticed by an LED light which comes on because it is connected to the normally open of the relay to show that the battery is fully charged.

3.1.2 Electronic control section

The electronic control section is made up of Arduino ATmega 2560 microcontroller. It serves as the heart of the device. All signals from the atmospheric parameter sensor sections are fed into the microcontroller for processing and further transmission to output sections. The set of instructions used to program the microcontroller was carried out using C programing language as attached in appendix A. Figure 3.4 shows the Pin schematic of the Arduino ATmega 2560 microcontroller utilised in the weather station.

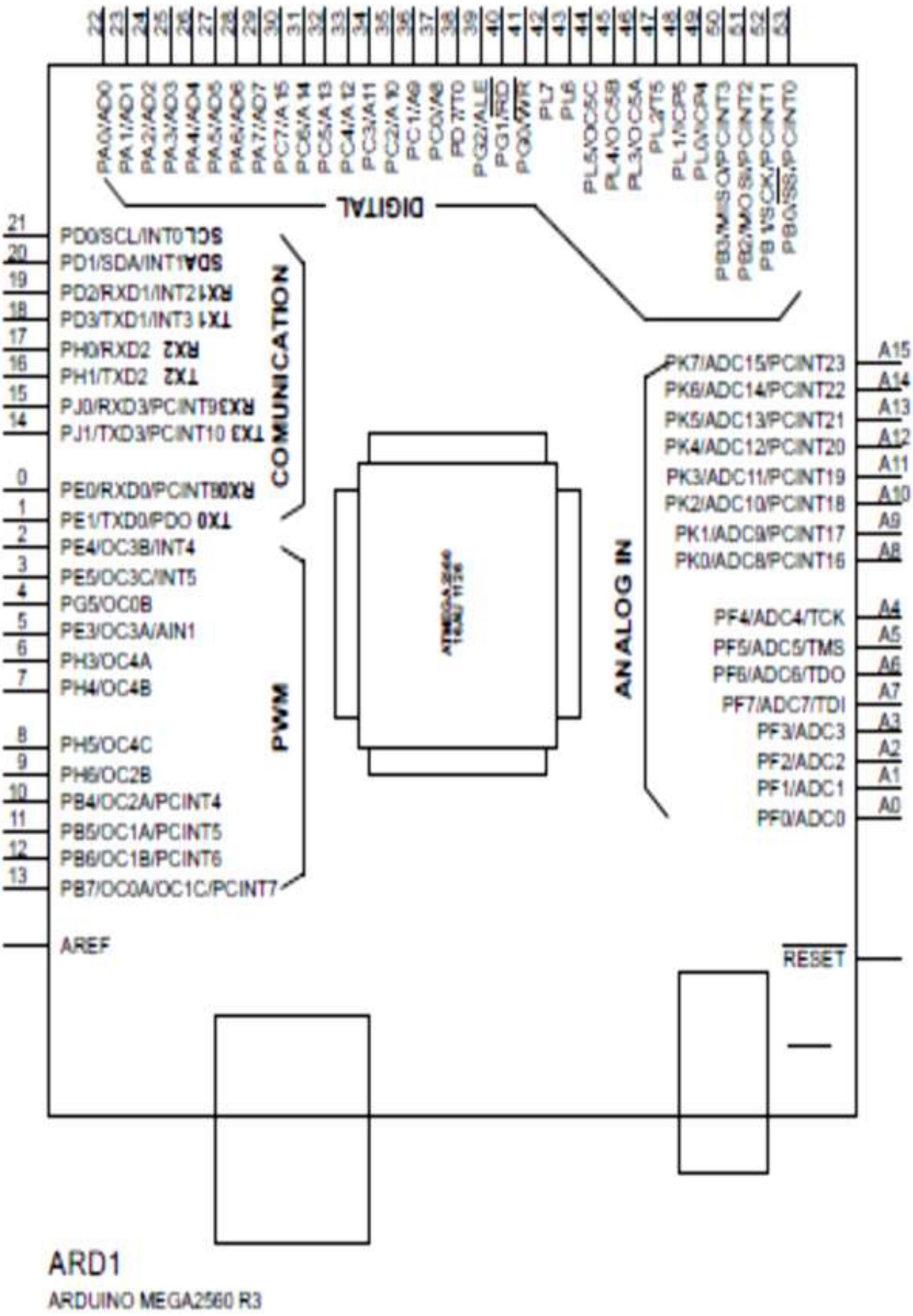


Figure 3.4: Pin schematic of the Arduino ATmega 2560 microcontroller

The microcontroller utilise it analogue and digital pins for receptions and processing of atmospheric signals received from the different electronic sensors.

3.1.3 Temperature and relative humidity monitoring section

DHT22 sensor was used to jointly measure and record atmospheric temperature and relative humidity at same time. It has the advantage of space economy as against using separate sensors for both parameters. The DHT22 sensor used in this work was earlier shown in Figure 2.11 while the circuit diagram showing the interconnection between the sensor and the microcontroller is shown in Figure 3.5.

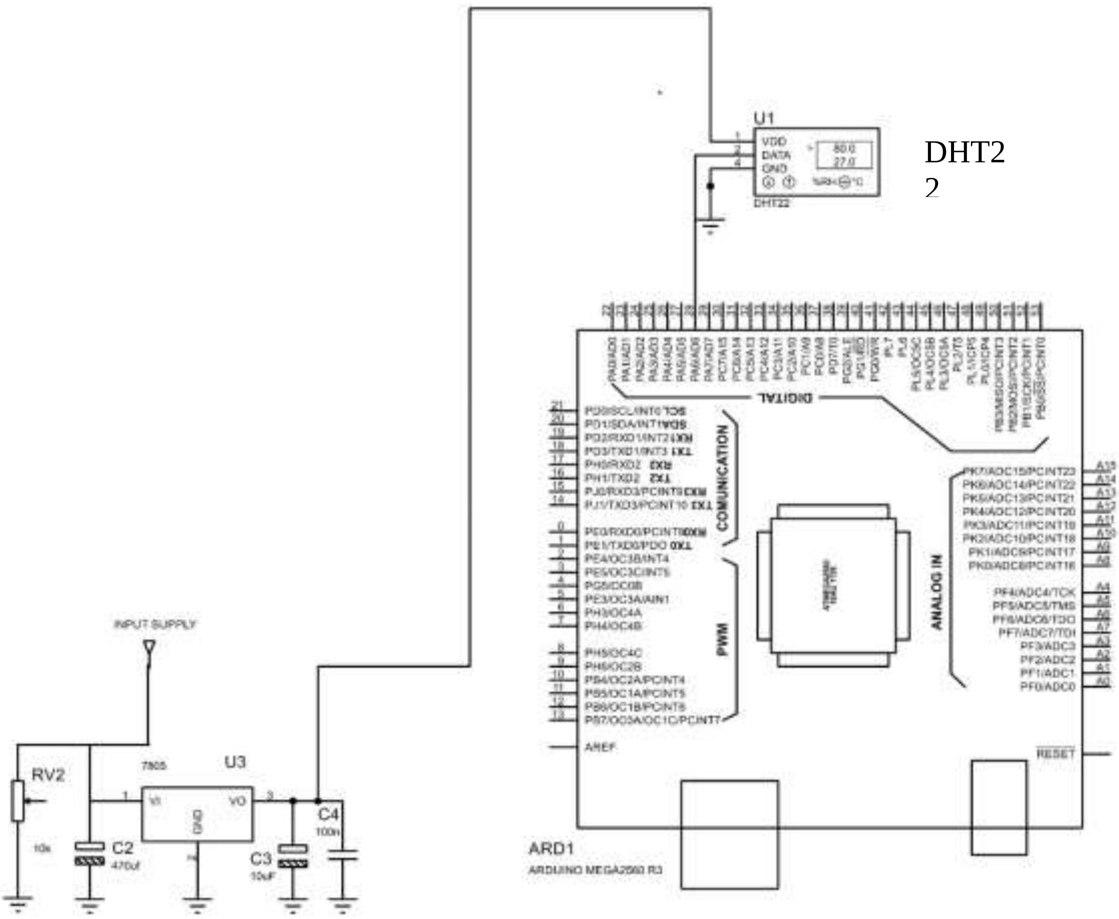


Figure 3.5: Circuit interface between the DHT22 sensor and the Atmega 2560 microcontroller.

From Figure 3.5, DHT22 sensor has three terminals: Vcc terminal which is the supply terminal and is connected to the 5V DC voltage source, through the output of the voltage regulator. VGND terminal is earth terminal which is connected to the zero voltage sources. The data line terminal is the terminal that carries information of the recorded atmospheric temperature and relative humidity readings. This data terminal is connected to PA6/A D6 of the microcontroller for the purpose of processing.

3.1.4 Pressure monitoring section

This section uses the BMP 180 sensor to measures atmospheric pressure and supply it as analogue output to the microcontroller. The sensor was shown in Figure 2.12 while the circuit connection between the microcontroller and the BMP 180 sensor is shown in Figure 3.6.

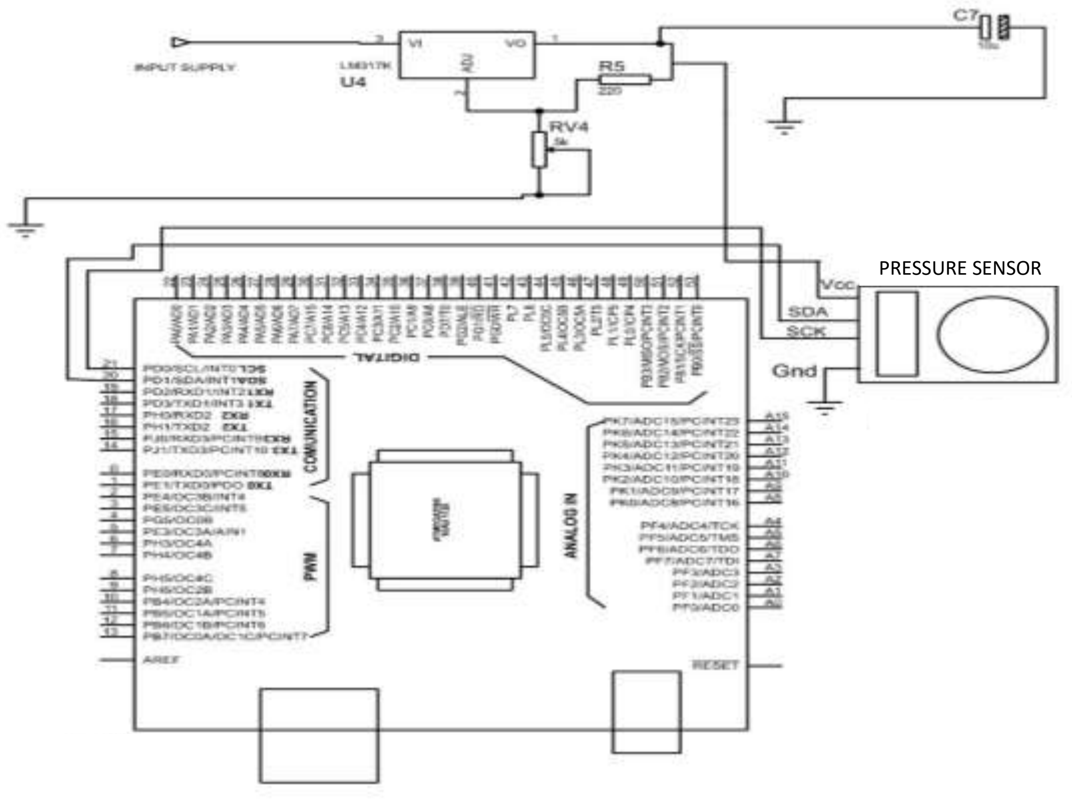


Figure 3.6: Circuit connection of microcontroller and the BMP 180 sensor.

From Figure 3.6, the BMP 180 sensor has four pin terminals: pin Vcc is connected to LM317 voltage regulator, which has 3V DC voltage source as it output, this is because the BMP 180 requires a maximum of 3V.

Terminal Gnd pin is connected to a zero-voltage source, for the purpose of earthling. Terminals SDA pin and SCK pin communicate with the microcontroller using I^2C

communication protocol. It is the serial clock pin terminal which the microcontroller sends a clock pulse to the BMP 180 sensor to give it output has received data for processing into useful format, and it is interfaced with the microcontroller through pin PDO/SCL/INT SCLO. SDA is the serial data pin terminal through which the microcontroller receives data from the sensor via pin PD1/SDA/INTSDA1 of the microcontroller.

3.1.5 Wind speed monitoring section

A three-cup anemometer device was assembled locally to measure the wind speed. The three-cup anemometer was mounted on a shaft, a magnet is attached at the base of the shaft. A reed switch sensor is positioned perpendicularly facing the magnet. When the three-cup anemometer rotates due to the effect of wind, the magnet passes the red switch, for every rotation, this triggers the reed switch to send a logic zero to the microcontroller. This process continues and the microcontroller counts the triggers. For every rotation made, the microcontroller calculates the wind speed using Davis equation.

$$\nu = P[\frac{2.25}{T}] \quad (3.3)$$

where ν is the speed in miles per hour (m/lh), P is the number of pulse during sampling period and T is the period per second.

Figure 3.7 shows the circuit connection between the microcontroller and the anemometer.

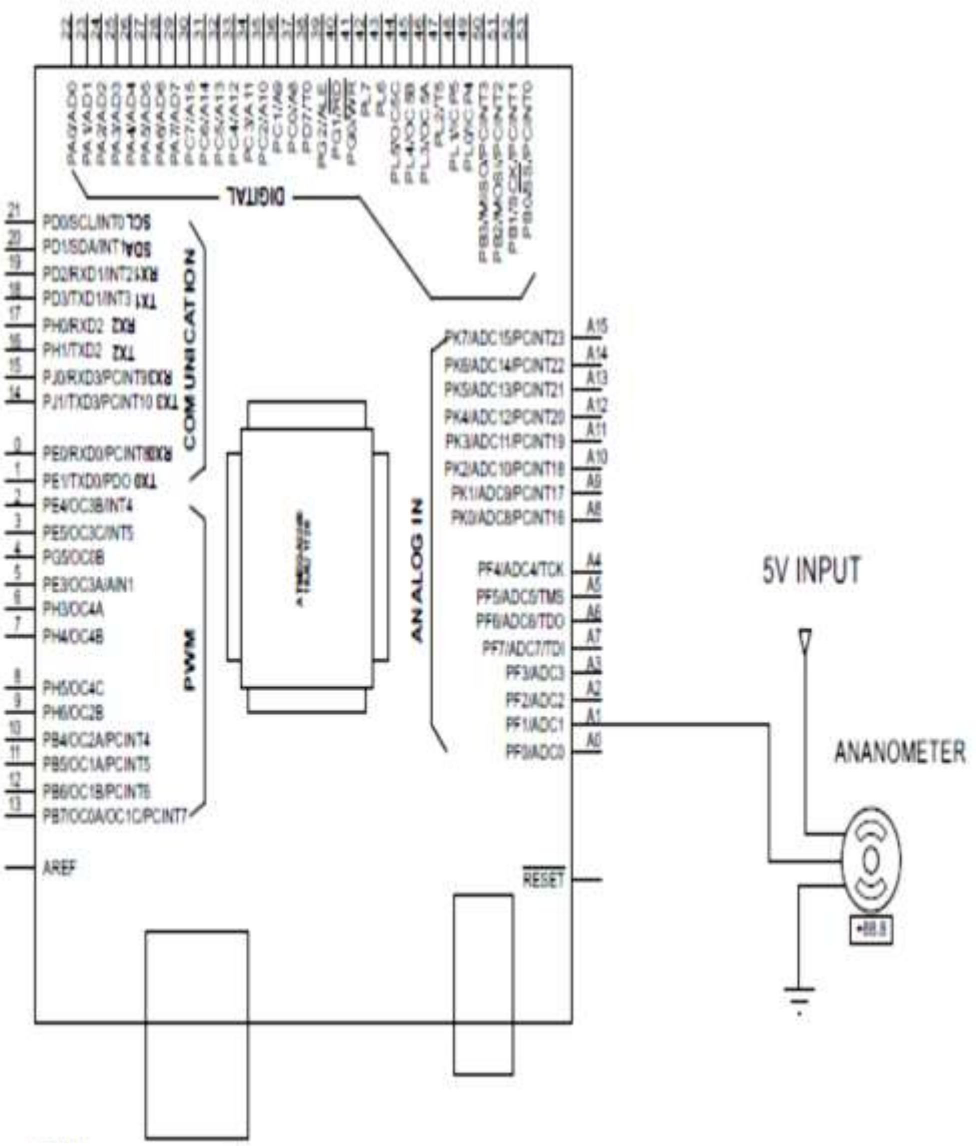


Figure 3.7: Circuit connection between the microcontroller and the anemometer

From the circuit diagram connection, it can be seen that the microcontroller receives wind speed signal from the anemometer through the output terminal of the anemometer connected to pin PFI/ADC1 of the microcontroller.

3.1.6 Wind direction monitoring section

A wind vane was locally constructed such that it shaft is coupled directly to a continuous moving variable resistor that gives an output of 0V to 5V which is mapped to $0^{o}$to 360^{o}. North direction serves as reference degree which is0^{o}, while east, south and west are 90^{o}, 180^{o} and 270^{o} respectively.

Figure 3.8 shows the circuit connection between the wind vane and the microcontroller.

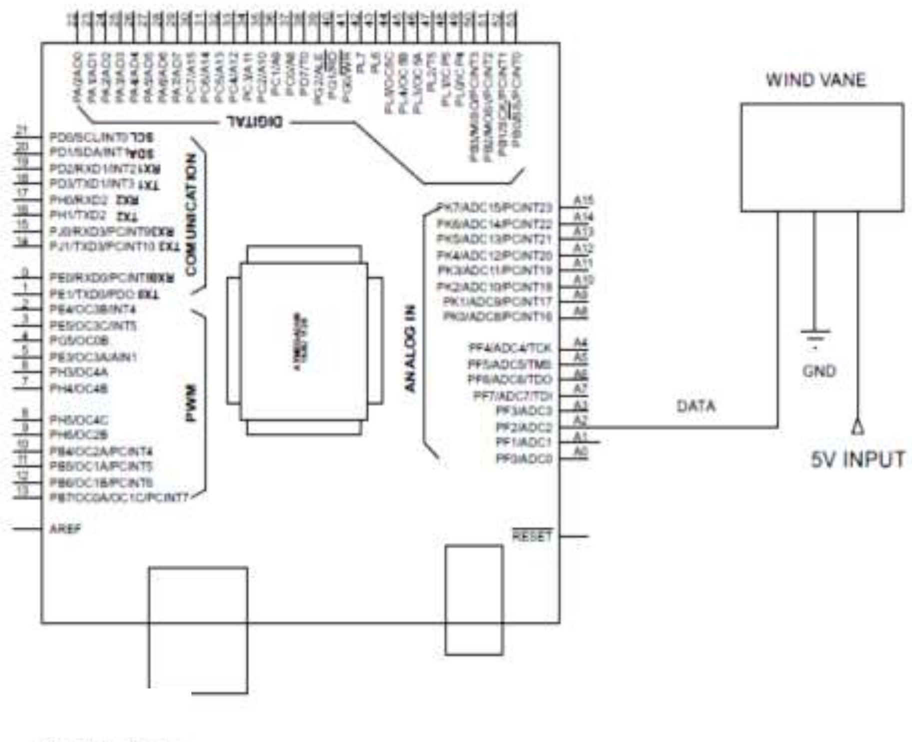


Figure 3.8: Circuit connections between the wind vane and the microcontroller

From Figure 3.8, the wind vane has three terminals, terminal Vcc is connected to the 5V DC voltage source from the LM 7805 voltage regulator output. G_{nd} terminal is connected to the zero voltage for the purpose of earthling. The output signal is connected to the microcontroller through pin PF2/ADC2.

3.1.7 Display section

This section utilise the screen of Nokia 3310 which has 84 by 48 pixels for the display of real time weather information for on-site users. The pixel LCD has 8 pins which serves as input pins to the microcontroller through which measured and processes atmospheric data are displayed as output on the LCD. The Nokia 3310 LCD has SPI as its communication protocol. Figure 3.9 shows the circuit connection between the LCD and the microcontroller.

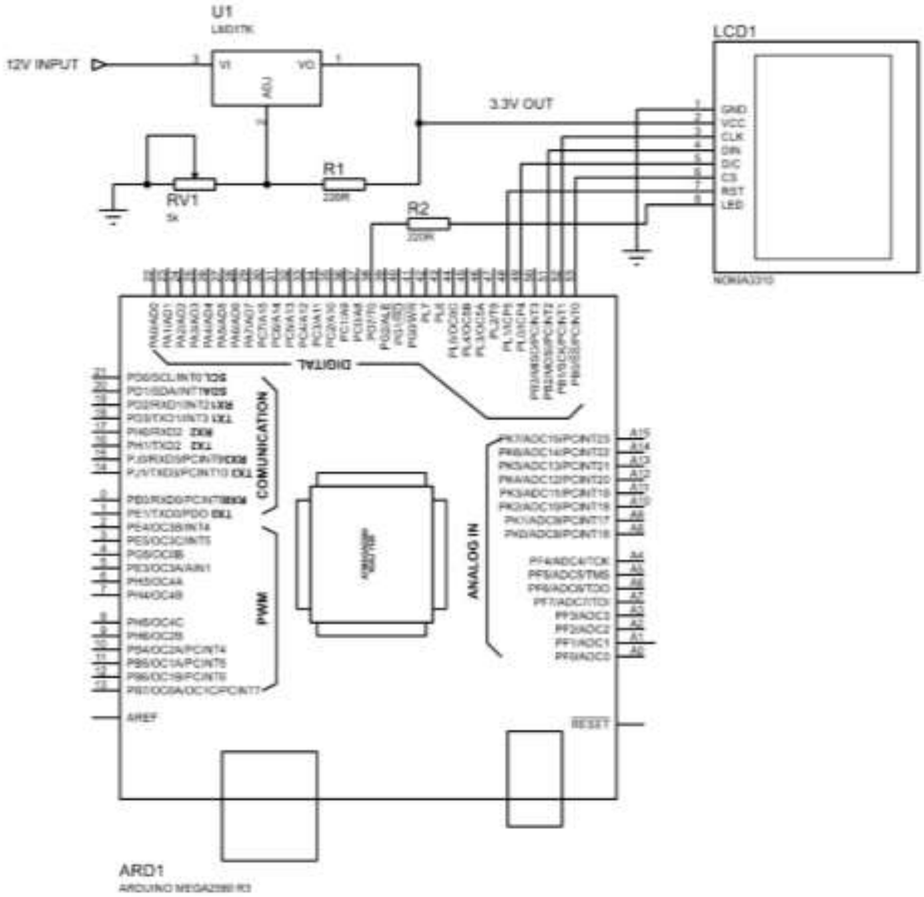


Figure 3.9: Circuit connections between the LCD and the microcontroller

From the circuit diagram, the LCD has the following terminal pins: Vcc terminal is connected to the regulated 5V DC source from the LM7805 voltage regulator. Terminal Gnd is connected to the zero voltage for the earthling. While CLK terminal is the clock pin terminal which the microcontroller sends clock pulse to the LCD and it is connected to pin PB1/SCK/PCINT1 of the microcontroller. Pin DIN is the data bus terminal through which the microcontroller sends processed data to the LCD and it is connected to pin PB2/MOSI/PCINT2 of the microcontroller.

Pin D/C is the data/command pin terminal which enables the microcontroller to communicate to the LCD to sense incoming data as a command to be displayed on the LCD, it is connected to pin PLO/ICP4 of the microcontroller. CS is the clip select pin terminal that selects a slave on the data and this achieved by the programme that the microcontroller is running on, CS terminal is interfaced with microcontroller through pin PBO/S5/PCINTO of the microcontroller.

RS pin terminal resets the display on the LCD and it is connected to pin PL1/ICP5 of the microcontroller. LED pin terminal is the back light that enables the screen to display processed data by the microcontroller and it is interfaced with microcontroller through pin PD7/TO.

3.1.8 Data transmission section

A Wi-Fi module serves as the data transmission link using Universal Asynchronous Receiver Transmitter (UART). The Wi-Fi module sends signal to the microcontroller during initialisation process, this triggers the microcontroller to send signals of processed atmospheric data to the Wi-Fi module to upload to an online server. Through the online server, real-time weather information can be accessed for out-of-site users. Figure 3.10 shows the circuit diagram connection of the Wi-Fi module and the microcontroller.

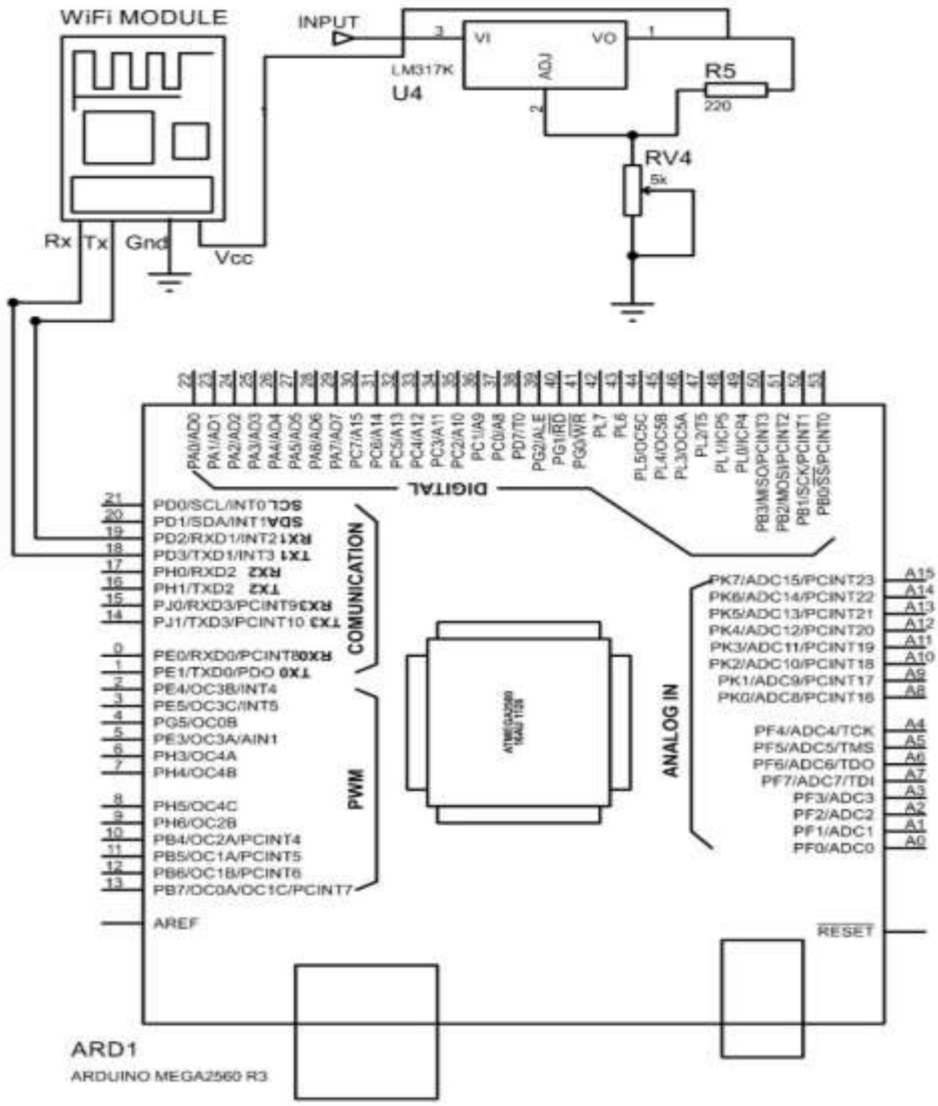


Figure 3.10: Circuit connection of the Wi-Fi module and the microcontroller

The Wi-Fi module has 4 pins terminals; Vcc is the pin terminal and it is connected to 3V DC voltage source from the LM317 voltage regulator. While Gnd pin terminal is connected to zero volt for earthling. Rx pin terminal is the receiver communication channel of the Wi-Fi module which is interfaced with microcontroller through pin PD3/TXD1/INT3TX1 and TX pin terminal is the transmitter communication channel of that interface between the microcontroller and Wi-Fi module. Tx pin terminal is connected to pin PD2/RXD1/INT2.

The overall circuit diagram showing the interconnection between sections of the designed weather station is shown in Figure 3.11.

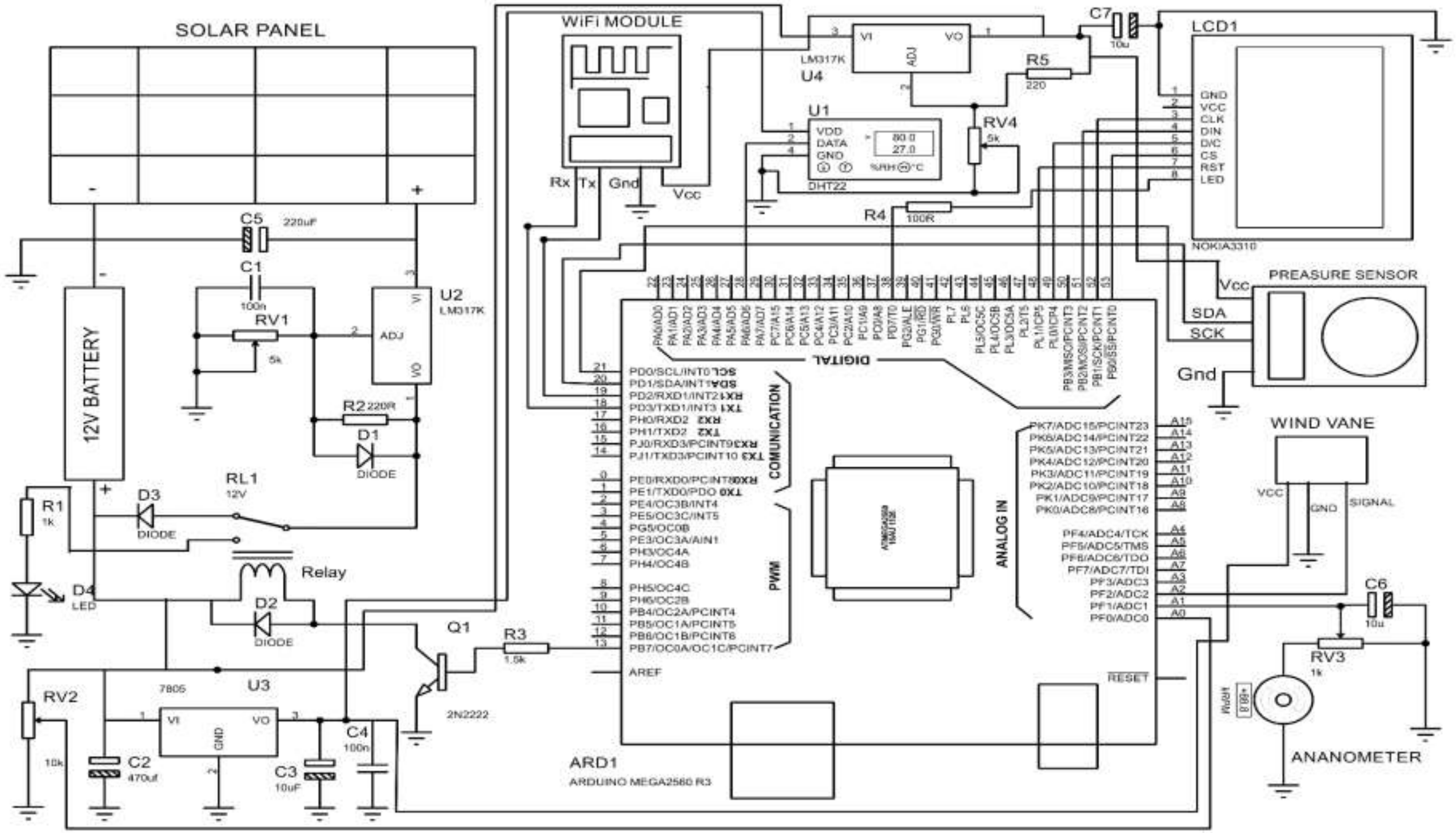


Figure 3.11: The overall circuit diagram showing the interconnection between sections of the designed weather station

3.2 Construction

This deals with the physical implementation of the weather station in accordance with the design so described. Attention was paid to local content, as such some components and sections were improvised using locally available substitutes. The following types of construction was done.

3.2.1 Electronic based construction

All components were sourced from electronic markets within Nigeria and assembled on breadboard to ascertain their workability before being hard-wired on Veroboard through the process of soldering. This was done section by section, followed by pre-testing and final linkage. The electronic panel was carefully placed in a plastics casing for protection of the circuitry and safety of handlers. PlateI, II and III shows the components mounted on the Vero Board, locally constructed anemometer and wind vane and the casing interior of the designed and constructed microcontroller-based Weather Station respectively.

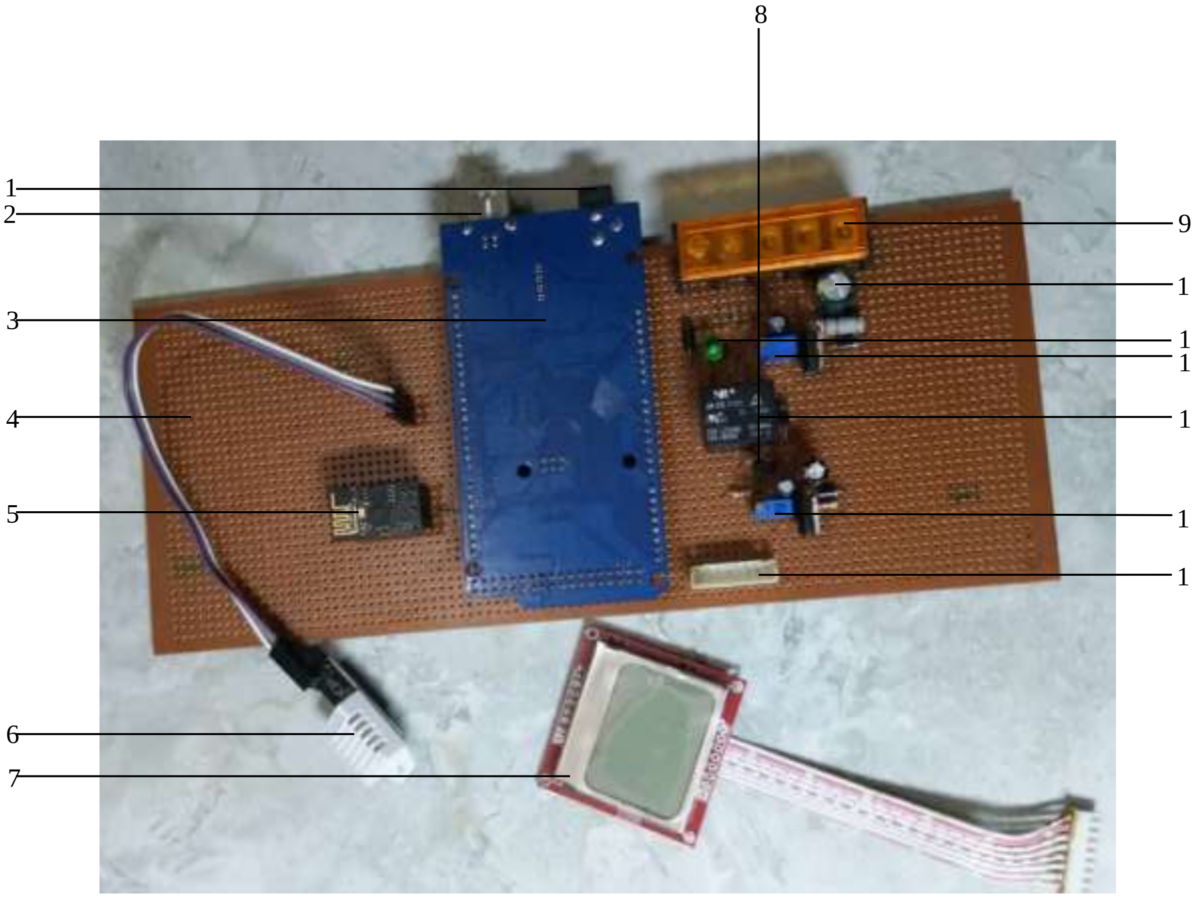


Plate I: Components mounted on vero board

Names of Major components mounted on the PCB

1. Battery plug
2. USB plug
3. Microcontroller on the PCB board
4. Veroboard
5. Wi-Fi module
6. DHT 22 sensor
7. LCD
8. Diode
9. Power port
10. Capacitor
11. LED indicator
12. Variable resistor
13. Relay
14. Resistor
15. Port for connecting the LCD

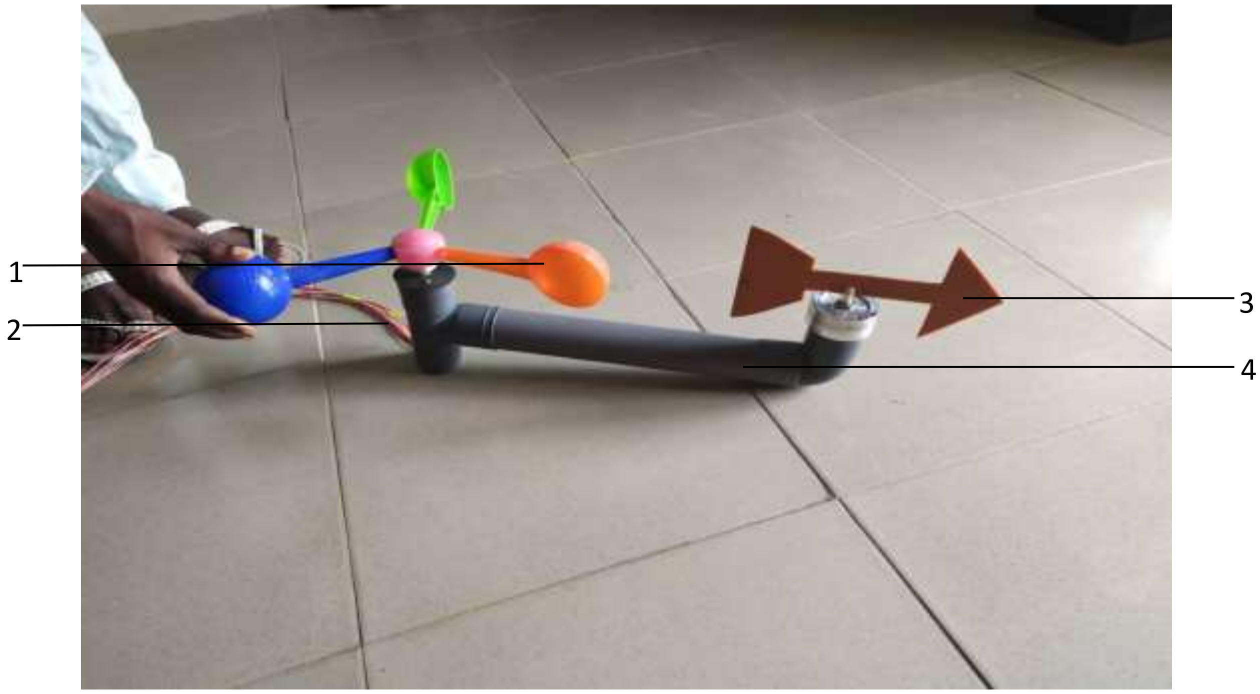


Plate II: Locally constructed anemometer and wind vane

Names of components labelled

1. Locally constructed three cup anemometer
2. Signal/electrical cables to the microcontroller
3. Locally constructed wind vane
4. Pipe housing the three cup anemometer, wind vane and signal/electrical cables to the microcontroller

Plate III: Casing interior of the designed and constructed microcontroller-based weather station

Names of components in plate III

1. Battery
2. Casing

3.2.2 Mechanical construction

A metal stand was carefully constructed using steel pipes and angle iron on which the device is mounted. Provisions were made to adequately sits the solar panel and the designed and constructed device. The stand is adjustable to whatever height a user intends by slotting an appropriate extension rod into the vertical supporting stand provided. It has the advantage of holding the solar panel and wind instruments in suitable positions and at heights of interests for experimental purposes. The stand was painted white for aesthetics and protection from damage that might result due to prolong exposure to the forces of weather. Plate IV shows the adjustable metal stand.

Plate IV: Adjustable Metal Stand

Names of the labels on plate IV

1. Position for mounting of the casing housing the electronic sensors and electronic panel
2. Vertical stand
3. Position for mounting wind instrument
4. Position for mounting solar panel

3.2.3 Software implementation

The designed and constructed weather station is microcontroller-controlled and web-based. Therefore, it is software dependent. The implementation of the software is subdivided as follows:

3.2.3.1 Flowchart

A flowchart shows the software implementation of the processes of an operation.

The flowchart for the operation, of the microcontroller-based weather station developed in this work is shown in figure 3.12

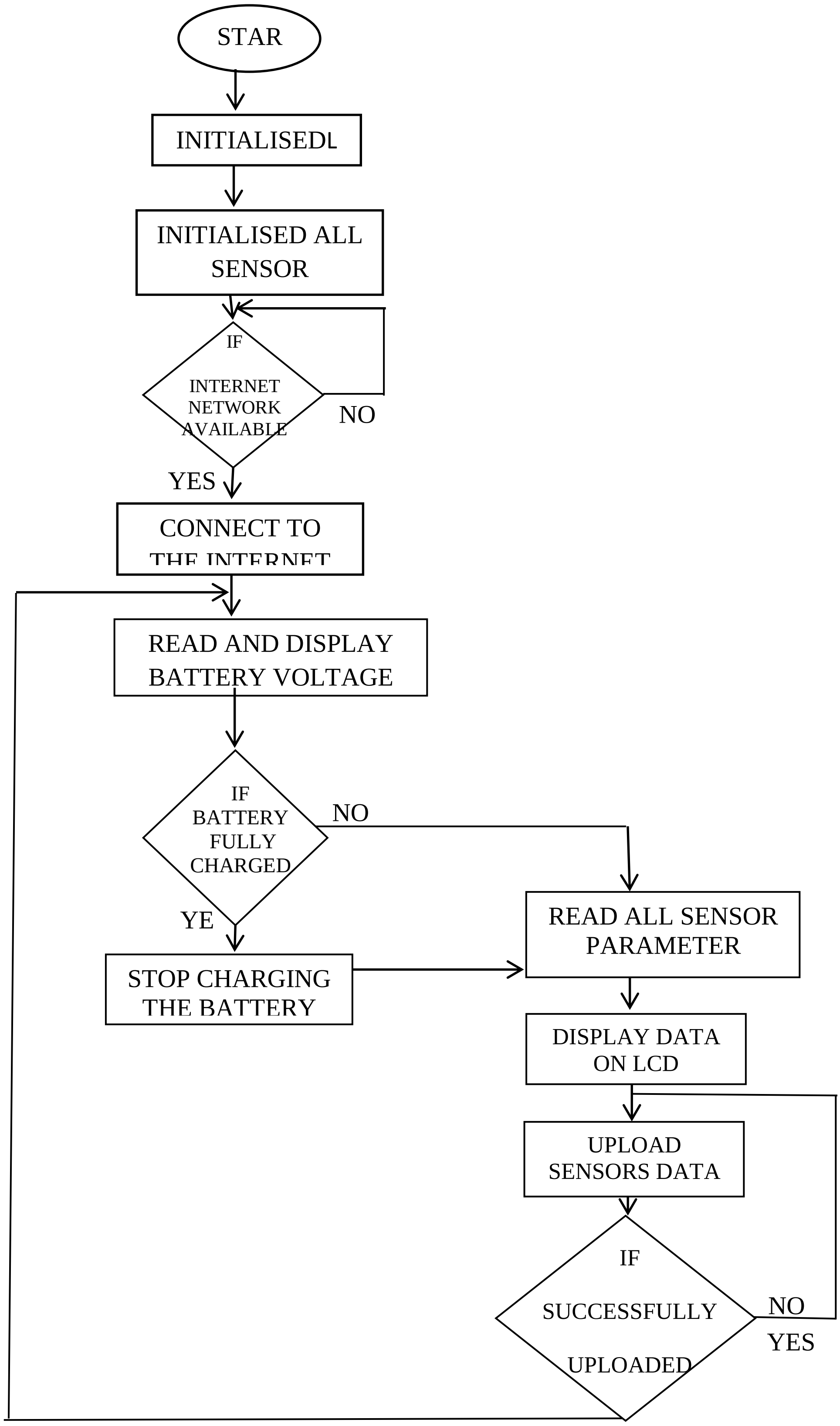


Figure 3.12 Software operations of the weather station

The system begins with initialisation. This initialisation process configure all the peripherals (Modules) connected to the microcontroller such as LCD, pressure sensor, temperature and humidity sensor, Wi-Fi Module, anemometer sensor and wind vane sensor. After all the initialisation, the system checks for network (Internet) availability in order to set up data uploading channel with the cloud. The microcontroller remains here until a successfully connection is achieve.

After connected to the internet, the system read battery voltage through its internal analog to digital converter (ADC) and display it on the LCD for user visualisation. Using the data obtain from the battery, the system check to see if battery is fully charge or not. When fully charge, the battery will be disconnected from the charging source else the battery keep charging.

After all the above processes, the microcontroller read all parameters which are pressure, temperature, humidity, wind speed and direction. The microcontroller uploads the data read from all the sensors to the cloud and check to see if the process was successful or not. If the data was successfully uploaded, the microcontroller go back to the sensors reading stage and loop around that stage continuously. If data fails to be uploaded, the microcontroller will try it again for few seconds before going back to data reading stage.

3.2.3.2 Website implementation, the Thingspeak technology

In this work, logged atmospheric data are hosted on the web using Internet of Things (IoT). Internet of things (IOT) is a software technology which provides internet working for physical devices, buildings, vehicles and other components like sensors and actuators, by giving network connectivity to systems embedded with electronics software, these objects are able to collect and exchange data. By using internet of things objects to be sensed or controlled remotely through existing network. It therefore gives opportunity to connect physical world with computer based system.

In this work, a platform of IoT, Thingspeak was employed as an interface between the device and out-of-site users. Thingspeak was used to manage network connection, real-time data collection from sensors, storage of data collected and analysis or visualisation of collected data. This innovation has saved a lot of cost and reduction in the size of the device.

Figure 3.13 shows the interface of thingspeak.com website

Figure 3.13: Interface of thingspeak.com website

Other details extractable from the Thingspeak platform are data, time and unique ID number of each data entry. To lunch the Thingspeak environment, the following stops are followed:

The process begins by creating a valid e-mail, which will be used to register an account with www. Thingspeak.com. A password and a user name is then made available to the webpage for subsequent visit to the site. There after a link will be sent to the new valid e-mail. This link give access to confirmation and activation of the new webpage created for logging data from the weather station. Thingspeak site is accessed by logging in with my user name and password, this lead to provision of an ICON write key which is alpha numerical.

The write key serves as a unique ID on the Thingspeak platform. The write key will be inserted in the microcontroller program, in order to enable the site recognise parameters coming from the solar powered weather station, for the purpose of storing it data and display graphically on it webpage. Channels are now created to see each incoming atmospheric temperature, relative humidity, atmospheric pressure, wind speed and wind direction as field 1, field2, field 3, field 4 and field 5 respectively. Each field stores in excel file and graphically displays atmospheric parameter uniquely assigned to it. On the Thingspeak webpage.

3.3 Mode of Operation of the Weather Station

The designed and constructed microcontroller based weather station can be mounted for operation as shown in Plate V.

Plate V: The Set-Up of the Real-Time, Solar-Powered Microcontroller-Based Weather Station.

The metal stand is mounted on the ground firmly, there after the casing housing the control panel and other electronic sensors is placed on the metal stand and screwed firmly. The solar panel also placed on the metal stand and screwed firmly to the metal stand, the cables on the solar panel is connected appropriately to the inlet space provided on the control panel casing. Locally constructed anemometer and wind vane along with it cables is mounted on the metal stand and wind direction and wind speed is passed into the casing for the purpose of connecting them into appropriate input pins of the microcontroller to achieve accurate communication between the wind vane, anemometer and the microcontroller. At the point of mounting the locally constructed anemometer and wind vane, a GPS is employed to determine the north pole in such a way that the wind vane arrow will be at a reference position pointing anemometer as the north direction.

Next, the start button for the Wi-Fi module is turned on. This enable the Wi-Fi module to activate it initialisation process and there after create internet network. This is visibly seen by the appearance of three green lights showing initialisation process. After the initialisation process is completed by the Wi-Fi module, a red light will appear indicating that the Wi-Fi module has established internet network, but needs to be connected to source of power.

At this point, the start button on the control panel is turned on, once the device comes on the microcontroller activates configuration process on all the electronic sensors and charging processes between the battery and solar panel. Once this is done, the microcontroller immediately starts sending commands to the electronic sensors in order to retrieve their atmospheric data. The retrieved data is displayed on the LCD for on-site users and also uploaded online via Thingspeak for out of-site users. The uploaded data is stored in an excel file for archiving purpose, the uploaded data is also analysed graphically and digitally at a minute interval.

CHAPTER FOUR

4.0 RESULTS AND DISCUSSION

In this chapter, the performance of the designed and constructed weather station shall be evaluated and the test results presented. The purpose of performance evaluation is to assess the device if it meets design objectives. Two types of test shall be carried out: simulation test and field test.

4.1 Simulation of Circuit Design

In order to evaluate the performance of the circuit designed, some sections of the system were stimulated on proteus simulation software. The proteus simulation package shows the visual procedure of an electronic operation from start to finish. It identifies errors along the implementation pathway and display the expected output for a successful operation, to do this, the circuit design is reproduced on proteus and the relevant test tools and equipment activated. Once the simulation command is issued, a visual display of the electronic operation and result is obtained for the selected section. Figures 4.1, 4.2 and 4.3 gives the proteus simulations results for the Wi-Fi module initialisation in progress, Wi-Fi module initialization completed and the temperature and relative humidity display sections respectively

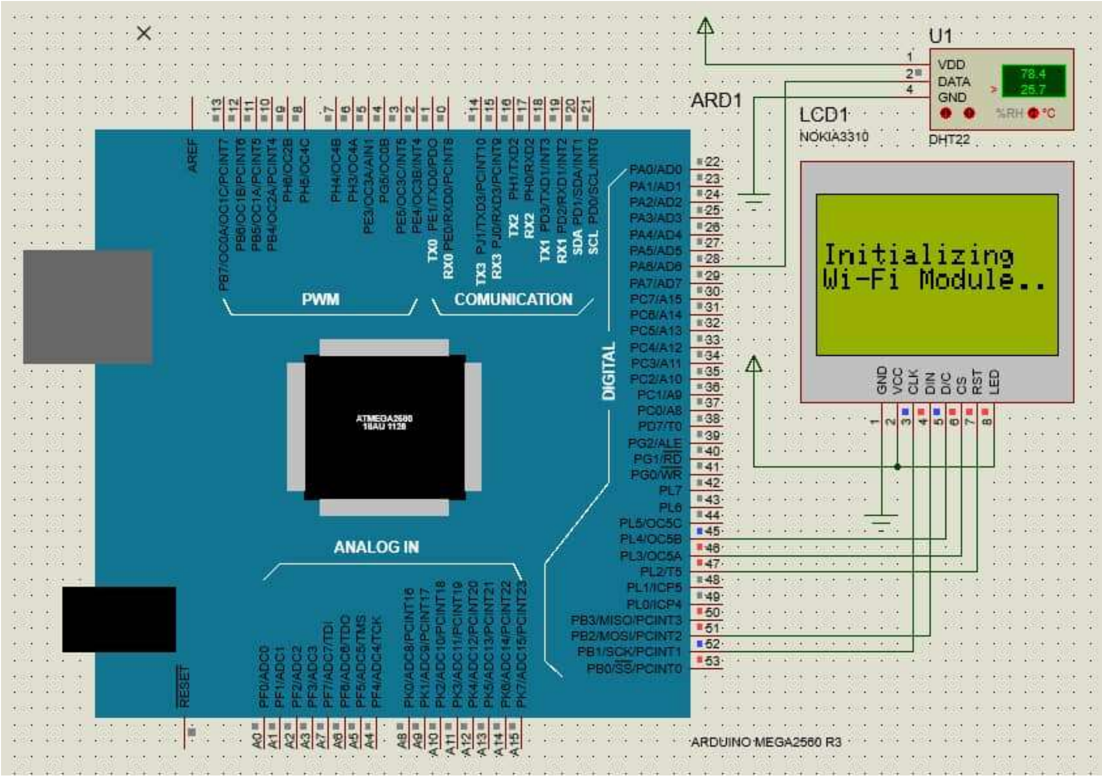


Figure 4.1: Proteus simulation result of Wi-Fi module initialisation process

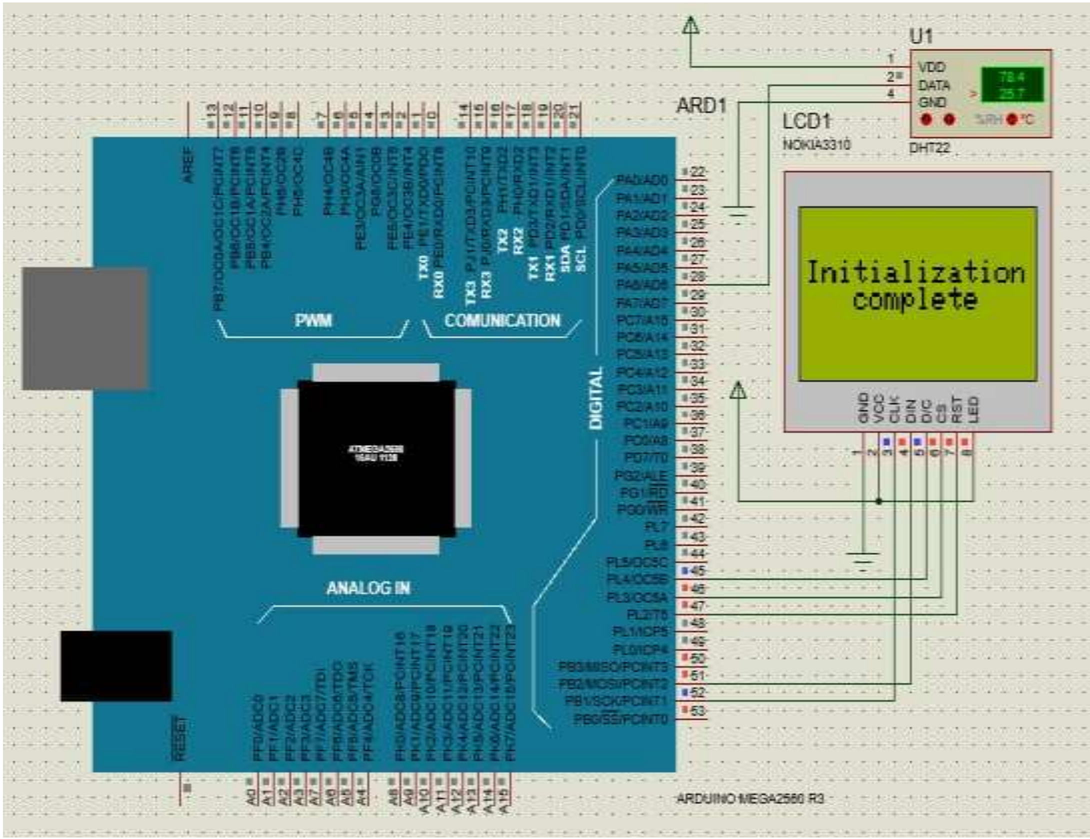


Figure 4.2: Proteus simulation result of Wi-Fi module initialisation completed

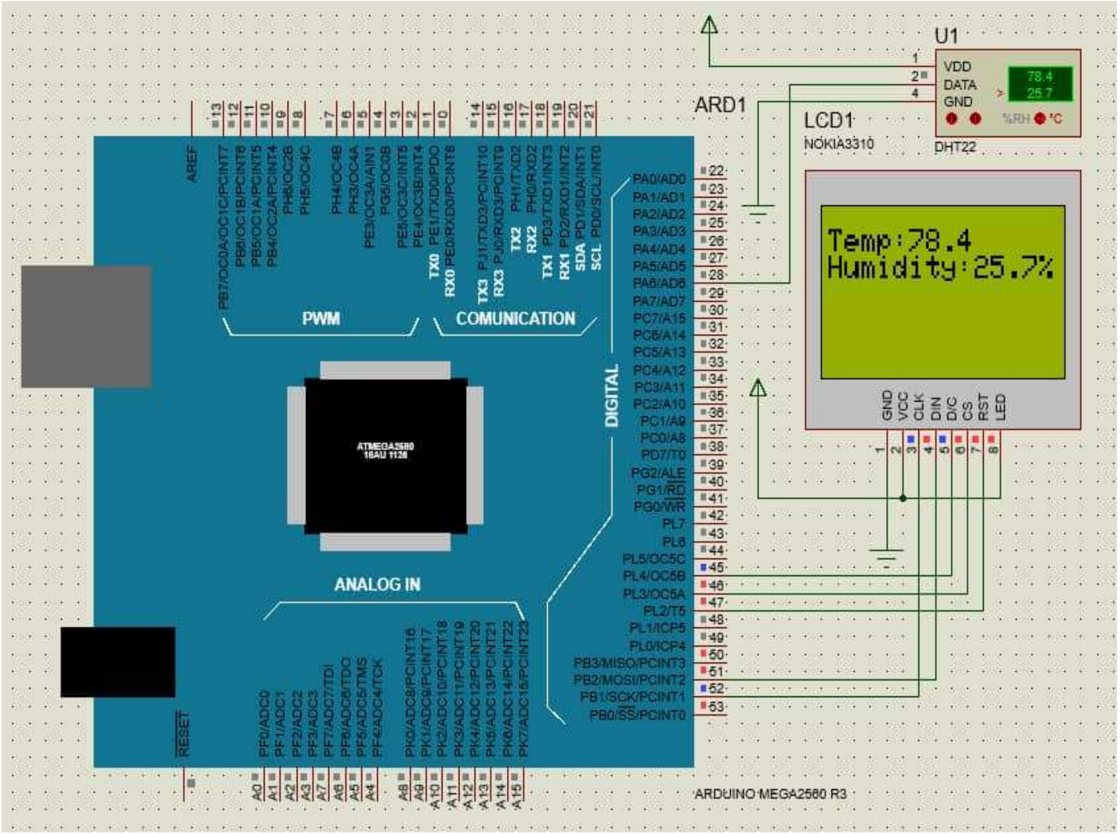


Figure 4.3: Proteus simulation result for temperature and relative humidity section

The result obtained for the selected sections show that the internal circuit operation is smooth. Same positive results are obtained for other sections since they are all controlled from same microcontroller. Positive simulation result is a technical performance comment and an indication that the device will perform effectively when physically operated.

4.2 Field Testing

The designed and constructed device was deployed to a location of interest and set into operation as explained in section 3.3. The result is monitored on-site and off-site. That is, on the LCD and online respectively. The results obtained shall be analysed and interpreted. The performance comment made after field testing is a reflection of the overall effectiveness of the weather station. Plate VI shows the field deployment of the weather station.

Plate VI: Field Testing

After deployment of the weather station on the field, atmospheric parameters were monitored on-the-site and off-site.

4.2.1 On-site data monitoring

On-site monitoring was achieved in real-time using the LCD positioned on the weather station's casing. Some of the LCD read-out are presented in Plates VIIa, VIIb and VIIc.

Plate VIIa: Initialisation process of the Wi-Fi Module

Plate VIIb: Initialisation process of the Wi-Fi Module complete

Plate VIIc: Display of atmospheric parameters on the LCD

4.2.2. Off-site data monitoring

For off-site monitoring the results are uploaded to the Thingspeak platform in graphical and digital formats every minute. These results are presented in Figures 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 4.10, 4.11, 4.12 and 4.13 respectively.

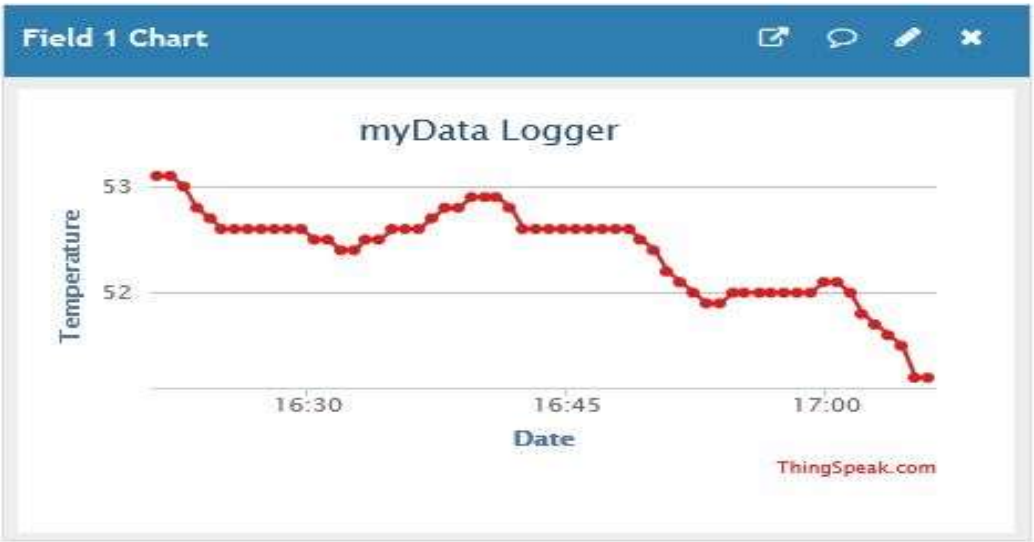


Figure 4.4: Graphical presentation of atmospheric temperature on the webpage

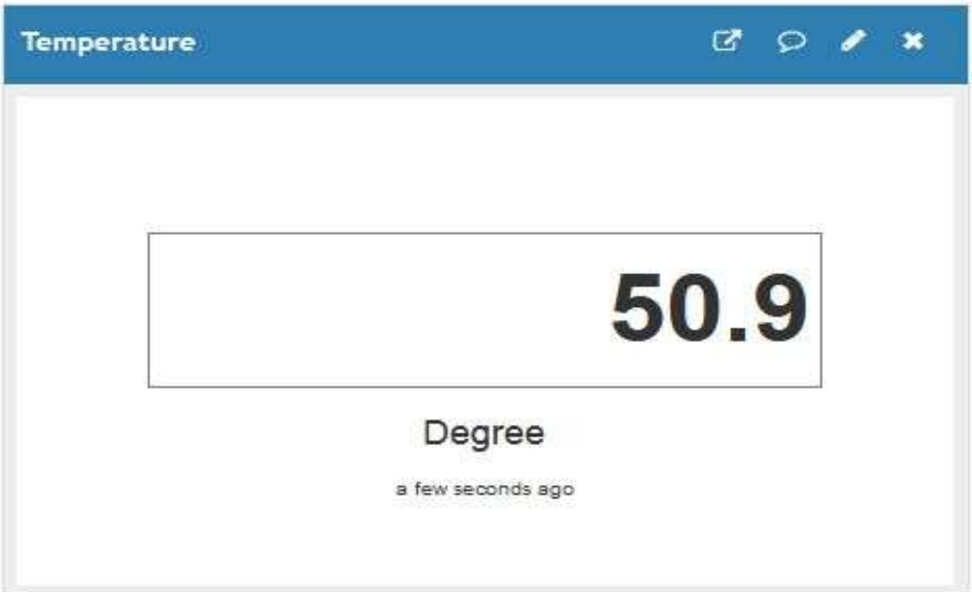


Figure 4.5: Digital presentation of atmospheric temperature as viewed on the webpage.

Figure 4.6: Graphical presentation of relative humidity as viewed on the webpage

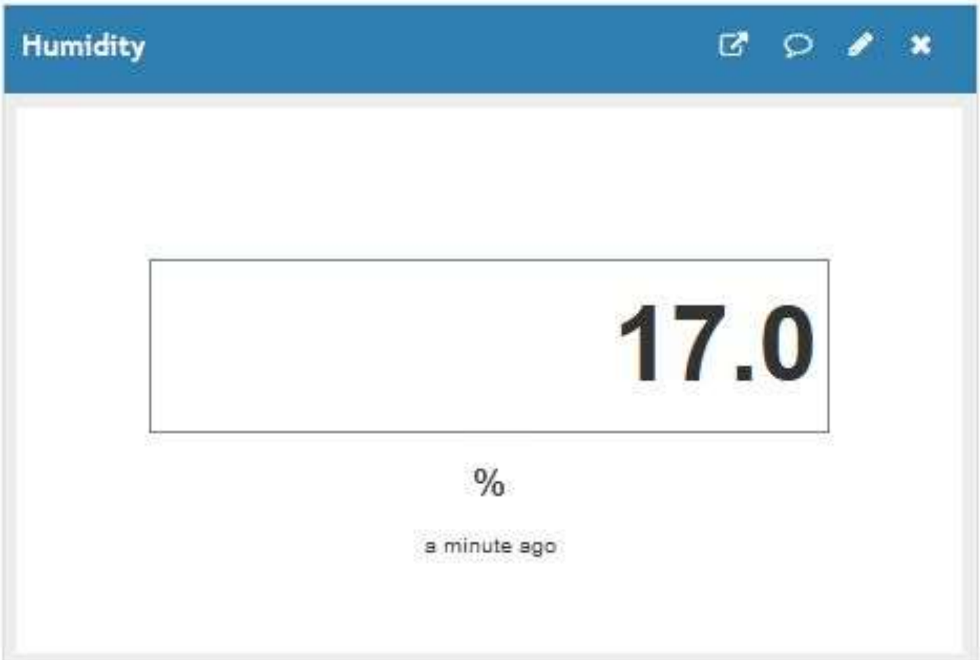


Figure 4.7: Digital presentation of relative humidity as viewed on the webpage.

Figure 4.8: Graphical presentation of atmospheric pressure as viewed on the webpage page

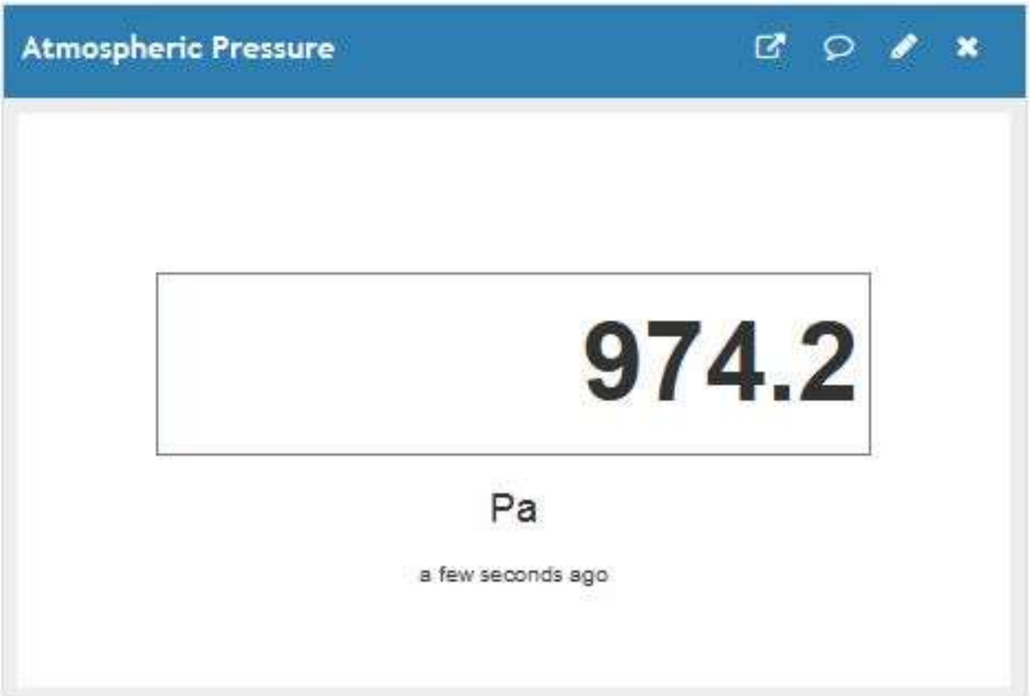


Figure 4.9: Digital presentation of atmospheric pressure as viewed on the webpage.

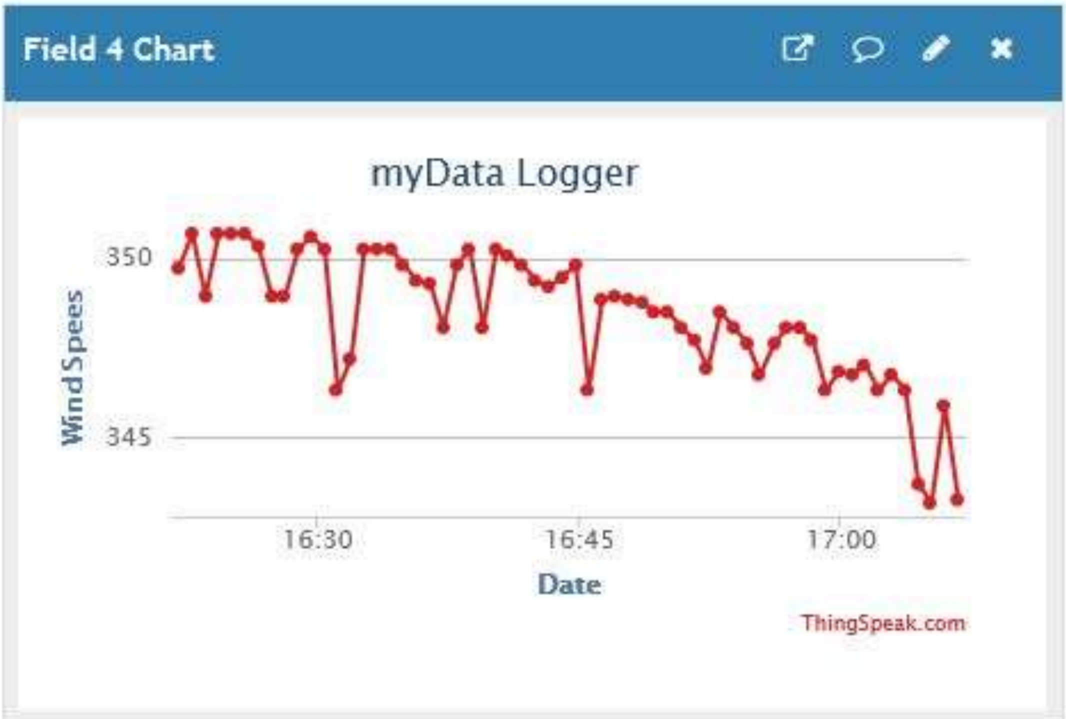


Figure 4.10: Graphical presentation of wind speed as viewed on the webpage

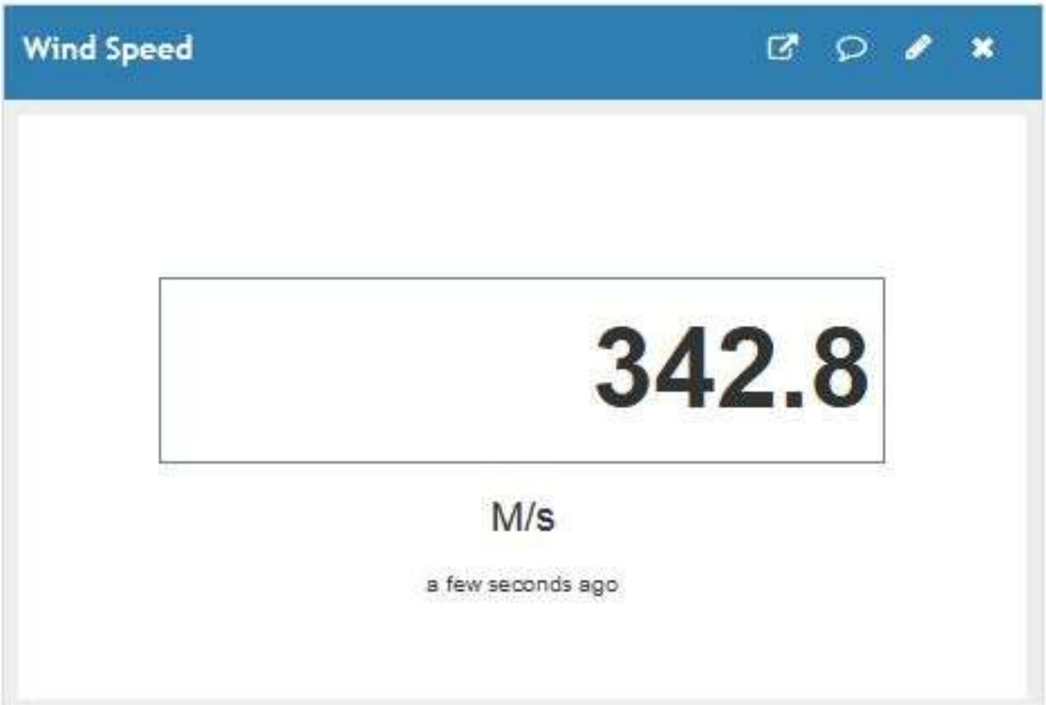


Figure 4.11: Digital presentation of wind speed on the viewed webpage.

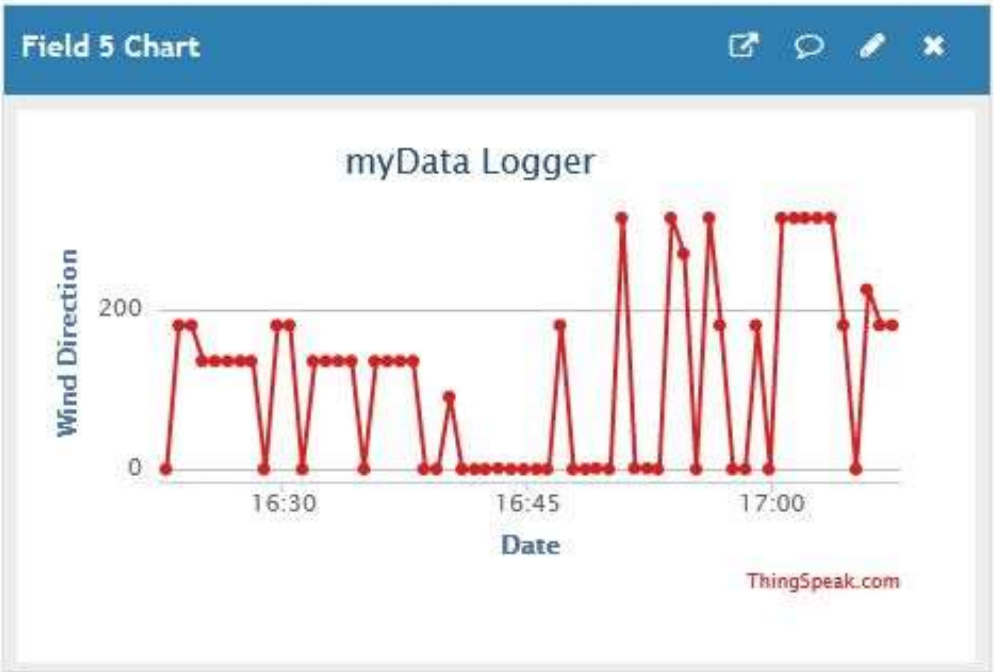


Figure 4.12: Graphical presentation of wind direction as viewed on the webpage.

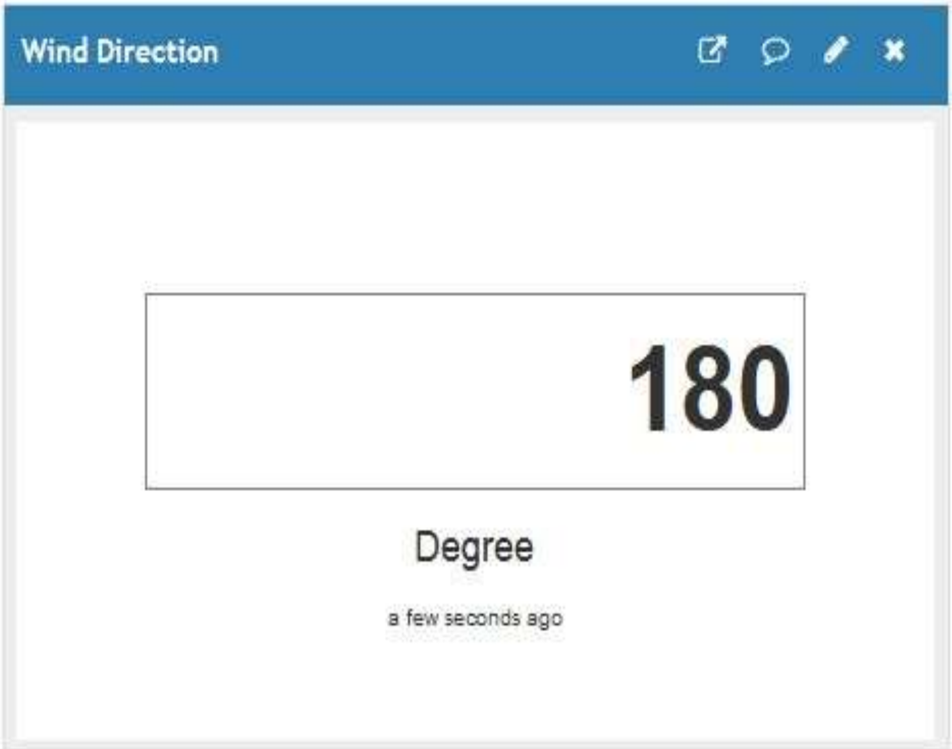


Figure 4.13: Digital presentation of wind direction as viewed on the webpage

The Thingspeak environment is capable of down loading the acquired data in an excel format which can now be saved as file and named accordingly using the date and time for which the atmospheric information was acquired. Figure 4.14 and 4.15 is a screen shot of the downloadable excel file format from the Thingspeak environment.

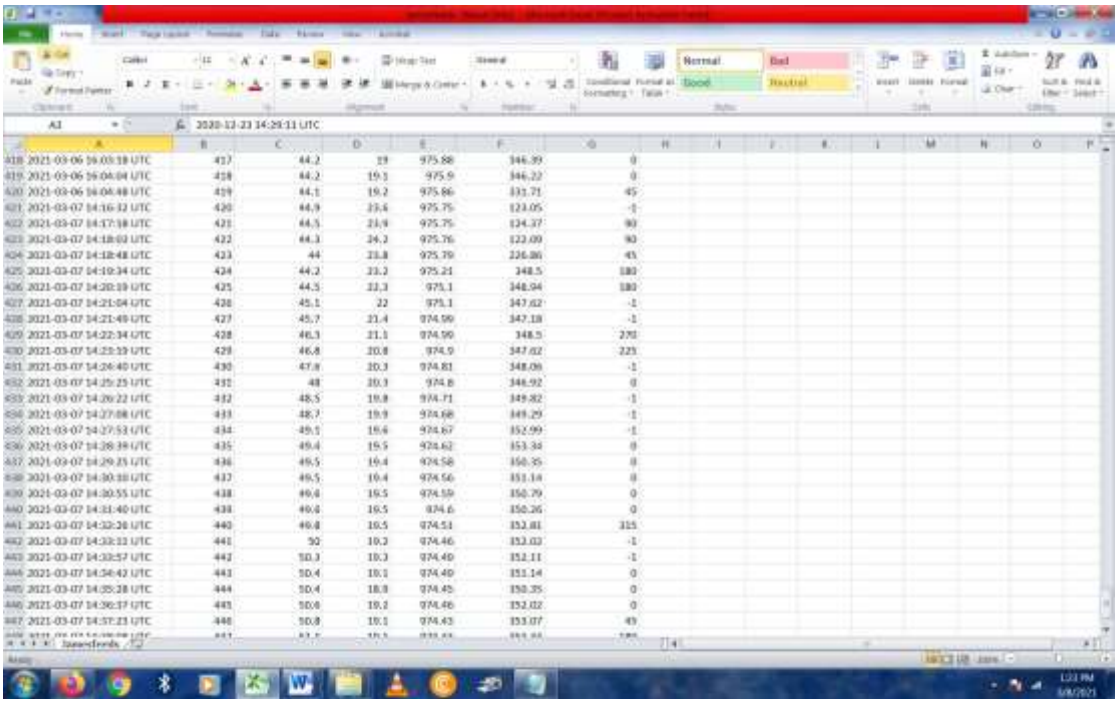


Figure 4.14: Screenshot of download atmospheric information from the Thingspeak.

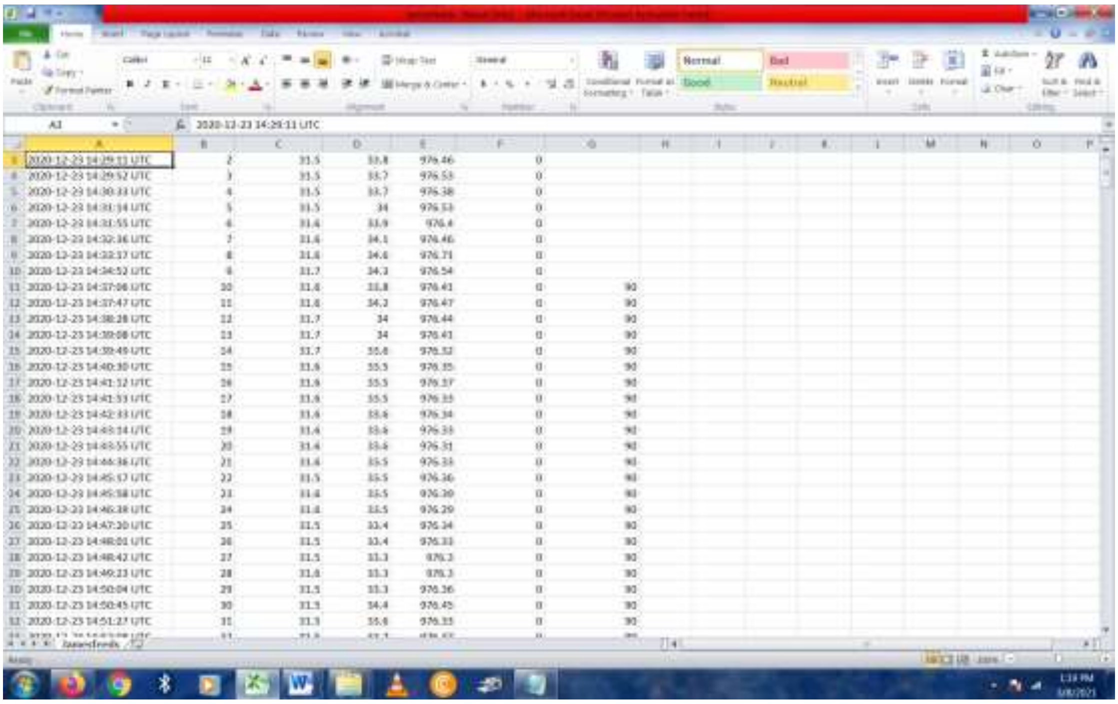


Figure 4.15: Screenshot of download atmospheric information from the Thingspeak.

From the Thingspeak excel environment. It is possible to obtain table of individual atmospheric parameters and plots of such parameters for the purpose of analysis and discussion. This is done by copying and pasting a parameter of interest from Figures 4.14 and 4.15 into Microsoft excel environment and obtaining their plots. These were done respectively for each of the atmospheric parameters as follows:

4.2.2.1. Temperature readings and analysis

Temperature reading obtained by the developed weather station as monitored via the Thingspeak platform were extracted into Microsoft Excel as tabulated in Table 4.1

Table 4.1: Temperature readings from the developed weather station

Date and Time (minute (^{0}C)	ID Number	Temperature
2021-03-07 14:16:32 UTC	420	44.9
2021-03-07 14:17:18 UTC	421	44.5
2021-03-07 14:18:03 UTC	422	44.3
2021-03-07 14:18:48 UTC	423	44
2021-03-07 14:19:34 UTC	424	44.2
2021-03-07 14:20:19 UTC	425	44.5
2021-03-07 14:21:04 UTC	426	45.1
2021-03-07 14:21:49 UTC	427	45.7
2021-03-07 14:22:34 UTC	428	46.3
2021-03-07 14:23:19 UTC	429	46.8
2021-03-07 14:24:40 UTC	430	47.6
2021-03-07 14:25:25 UTC	431	48
2021-03-07 14:26:22 UTC	432	48.5
2021-03-07 14:27:08 UTC	433	48.7
2021-03-07 14:27:53 UTC	434	49.1
2021-03-07 14:28:39 UTC	435	49.4
2021-03-07 14:29:25 UTC	436	49.5
2021-03-07 14:30:10 UTC	437	49.5
2021-03-07 14:30:55 UTC	438	49.6
2021-03-07 14:31:40 UTC	439	49.6
2021-03-07 14:32:26 UTC	440	49.8
2021-03-07 14:33:11 UTC	441	50
2021-03-07 14:33:57 UTC	442	50.3

To analyse the temperature variation, a graph of temperature against time was plotted as shown in figure 4.16

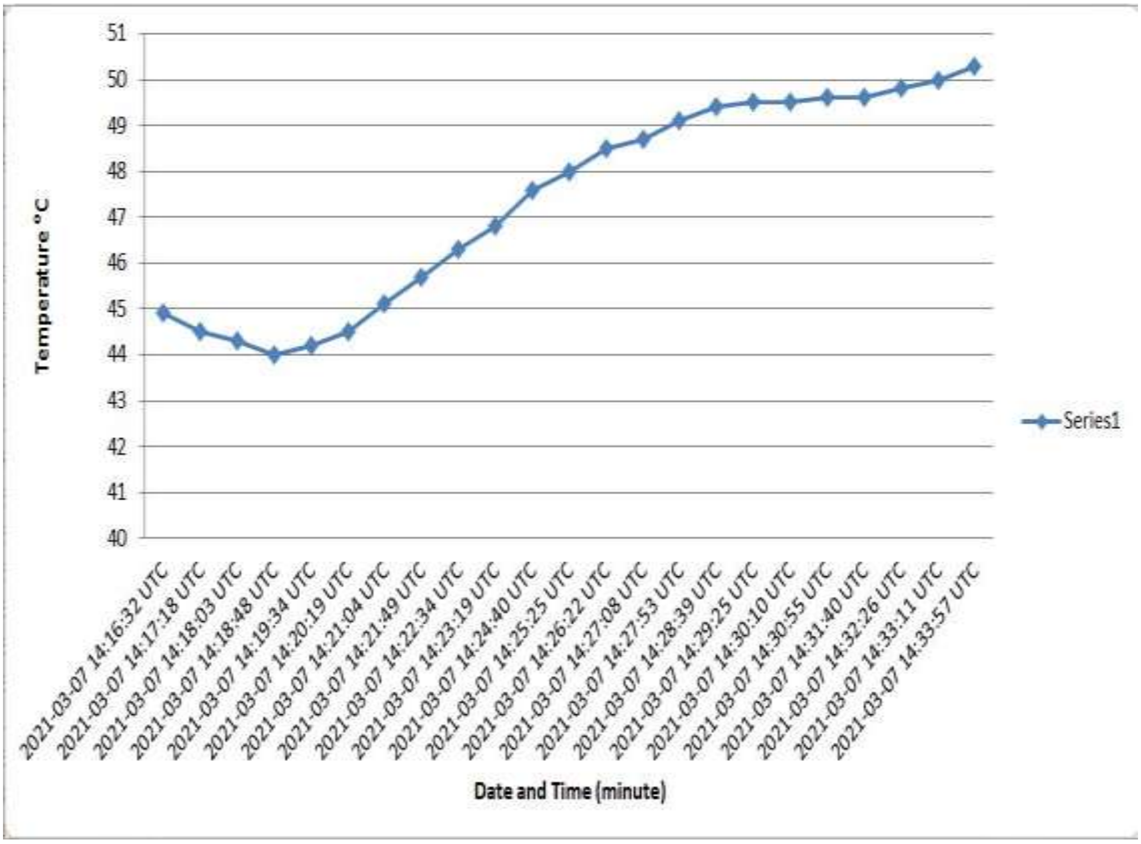


Figure 4.16: Temperature graph

The graph has temperature on the vertical axis, while time and date is on the horizontal axis. From Figure 4.16, the temperature was minimum at 44.0^{0}C and maximum at 50.3^{0}C corresponding to 2:18:48pm and 2:33:57pm of 7th March, 2021 respectively. The curve indicates that temperature dropped from 45^{0}C to 40^{0}C and rose to 50^{0}C within the period monitored.

4.2.2.2. Relative humidity readings and analysis

Relative humidity reading retrieved from the developed weather station as monitored via the Thingspeak platform were extracted into Microsoft Excel as tabulated in table 4.2 .

Table 4.2: Relative humidity readings from the developed weather station

Date and Time (minute	ID Number	Relative Humidity (%)
2021-03-07 14:16:32 UTC	420	23.6
2021-03-07 14:17:18 UTC	421	23.9
2021-03-07 14:18:03 UTC	422	24.2
2021-03-07 14:18:48 UTC	423	23.8
2021-03-07 14:19:34 UTC	424	23.2
2021-03-07 14:20:19 UTC	425	22.3
2021-03-07 14:21:04 UTC	426	22
2021-03-07 14:21:49 UTC	427	21.4
2021-03-07 14:22:34 UTC	428	21.1
2021-03-07 14:23:19 UTC	429	20.8
2021-03-07 14:24:40 UTC	430	20.3
2021-03-07 14:25:25 UTC	431	20.3
2021-03-07 14:26:22 UTC	432	19.8
2021-03-07 14:27:08 UTC	433	19.9
2021-03-07 14:27:53 UTC	434	19.6
2021-03-07 14:28:39 UTC	435	19.5
2021-03-07 14:29:25 UTC	436	19.4
2021-03-07 14:30:10 UTC	437	19.4
2021-03-07 14:30:55 UTC	438	19.5
2021-03-07 14:31:40 UTC	439	19.5
2021-03-07 14:32:26 UTC	440	19.5
2021-03-07 14:33:11 UTC	441	19.2

For the purpose of analysing relative humidity variation, a graph of relative humidity against time was plotted as presented in figure 4.17.

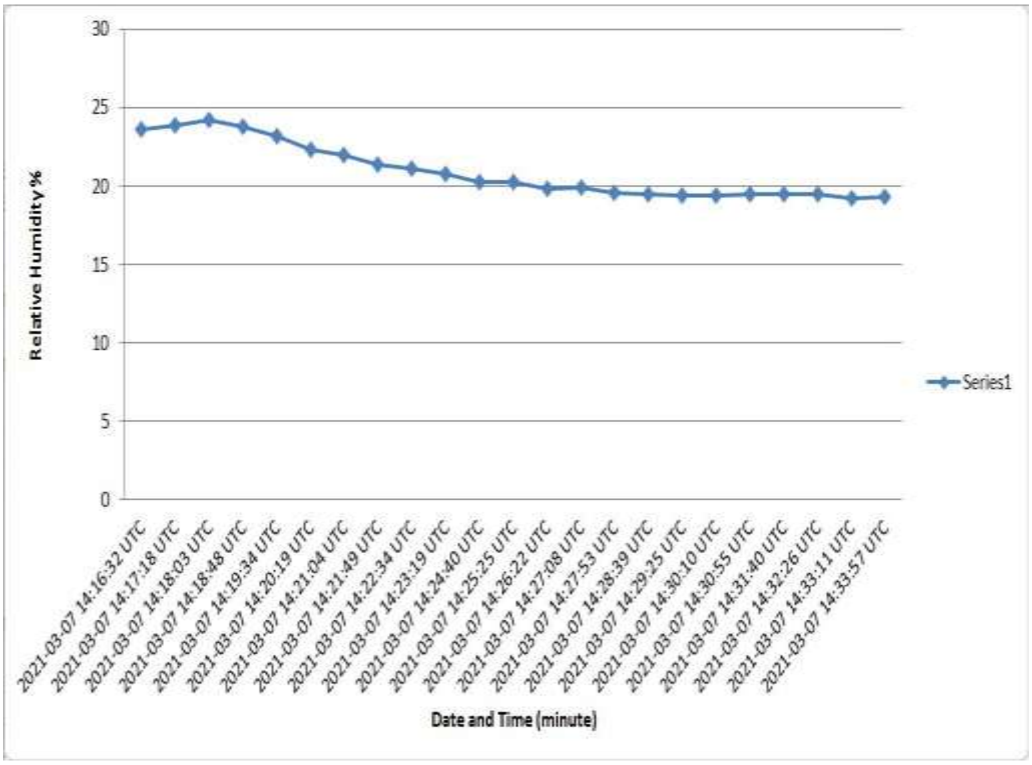


Figure 4.17: Relative humidity graph

The graph has relative humidity on the vertical axis, while time and date is on the horizontal axis. The graph on 4.17 shows that relative humidity was maximum at 24.2% and minimum at 19.2% corresponding to 2:18:03pm and 2:33:11pm of the 7th March, 2021 respectively.

The curve of the graph shows a drop in relative humidity with time.

4.2.2.3. Atmospheric pressure readings and analysis

Atmospheric pressure readings obtained by developed weather station as monitored via Thingspeak platform were extracted into Microsoft Excel as tabulated in Table 4.3

Table 4.3: Atmospheric pressure readings from the developed weather station

Date and Time (minute (Pa)	ID Number	Atmospheric Pressure
2021-03-07 14:16:32 UTC	420	975.75
2021-03-07 14:17:18 UTC	421	975.75
2021-03-07 14:18:03 UTC	422	975.76
2021-03-07 14:18:48 UTC	423	975.79
2021-03-07 14:19:34 UTC	424	975.21
2021-03-07 14:20:19 UTC	425	975.1
2021-03-07 14:21:04 UTC	426	975.1
2021-03-07 14:21:49 UTC	427	974.99
2021-03-07 14:22:34 UTC	428	974.99
2021-03-07 14:23:19 UTC	429	974.9
2021-03-07 14:24:40 UTC	430	974.81
2021-03-07 14:25:25 UTC	431	974.8
2021-03-07 14:26:22 UTC	432	974.71
2021-03-07 14:27:08 UTC	433	974.68
2021-03-07 14:27:53 UTC	434	974.67
2021-03-07 14:28:39 UTC	435	974.62
2021-03-07 14:29:25 UTC	436	974.58
2021-03-07 14:30:10 UTC	437	974.56
2021-03-07 14:30:55 UTC	438	974.59
2021-03-07 14:31:40 UTC	439	974.6
2021-03-07 14:32:26 UTC	440	974.51
2021-03-07 14:33:11 UTC	441	974.46

To analyse the atmospheric pressure variation, a graph of atmospheric pressure against time was plotted as shown in Figure 4.18

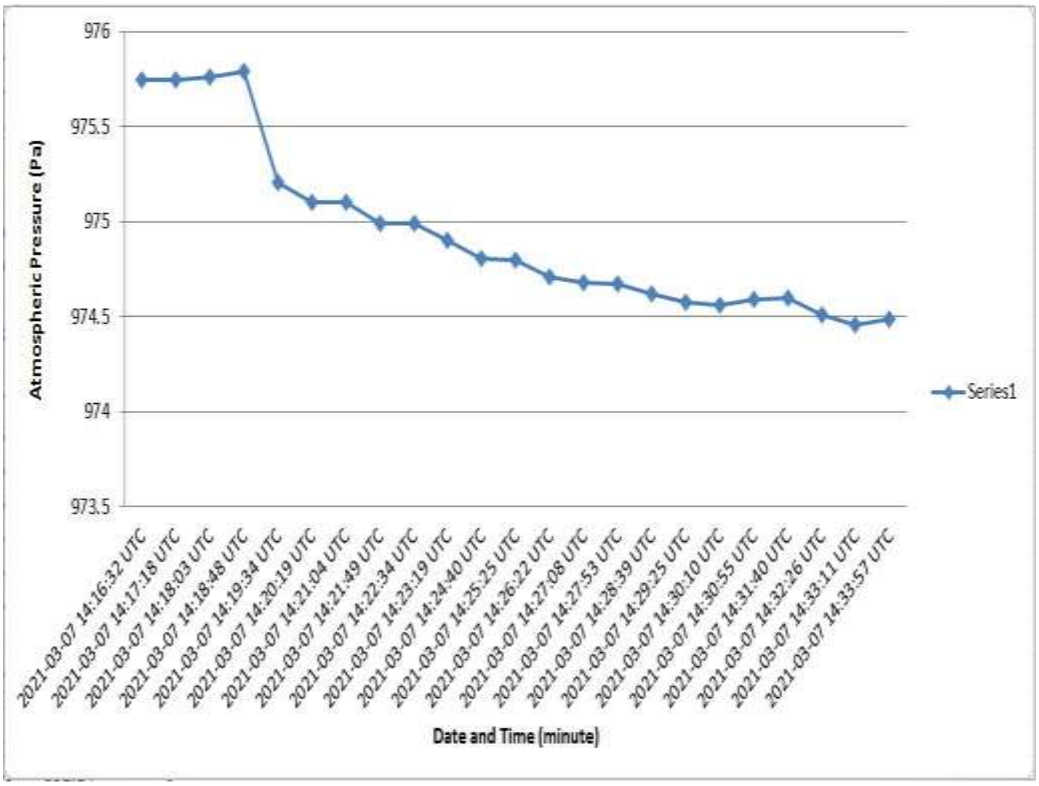


Figure 4.18: Atmospheric pressure graph

The graph has atmospheric pressure on the vertical axis, while time and date is on the vertical axis. From Figure 4.18 atmospheric pressure is maximum at 975.79Pa and minimum at 974.46Pa corresponding to 2:18:48pm and 2:33:11pm of 7th March, 2021 respectively. This is visible on the curve of the graph which shows a steady but gradual decrease in atmospheric pressure after the initial peak at 975.79Pa.

4.2.2.4. Wind speed readings and analysis

Wind speed readings retrieved by developed weather station as monitored via Thingspeak platform were extracted into Microsoft Excel and tabulated in Table 4.4.

Table 4.4: Wind Speed Readings from the Developed weather station

Date and Time (minute	ID Number	Wind Speed (m/s)
2021-03-07 14:16:32 UTC	420	123.05
2021-03-07 14:17:18 UTC	421	124.37
2021-03-07 14:18:03 UTC	422	122.09
2021-03-07 14:18:48 UTC	423	226.86
2021-03-07 14:19:34 UTC	424	348.5
2021-03-07 14:20:19 UTC	425	347.62
2021-03-07 14:21:04 UTC	426	347.18
2021-03-07 14:21:49 UTC	427	348.5
2021-03-07 14:22:34 UTC	428	347.62
2021-03-07 14:23:19 UTC	429	347.62
2021-03-07 14:24:40 UTC	430	348.06
2021-03-07 14:25:25 UTC	431	346.92
2021-03-07 14:26:22 UTC	432	352.99
2021-03-07 14:27:08 UTC	433	353.34
2021-03-07 14:27:53 UTC	434	350.35
2021-03-07 14:28:39 UTC	435	351.14
2021-03-07 14:29:25 UTC	436	350.79
2021-03-07 14:30:10 UTC	437	350.26
2021-03-07 14:30:55 UTC	438	350.79
2021-03-07 14:31:40 UTC	439	350.26
2021-03-07 14:32:26 UTC	440	352.81
2021-03-07 14:33:11 UTC	441	352.02

To examine in detail, the Wind speed variation, a graph of wind speed against time was plotted as shown in Figure 4.19.

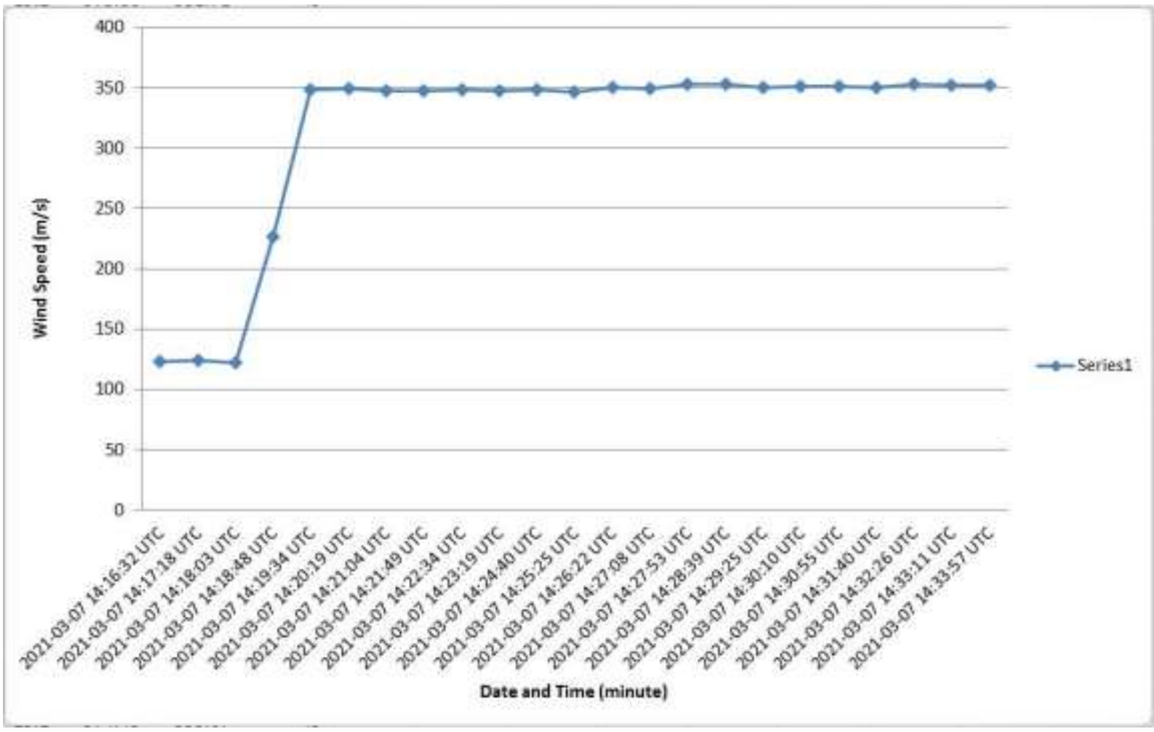


Figure 4.19: Wind speed graph

The graph has wind speed on the vertical axis, while time and date is on the horizontal axis. From figure 4.19, the wind speed was maximum at 353m/s and minimum at 122.09m/s corresponding to 2:28:39pm and 2:18:03p, of 7th March, 2021 respectively. The Curve shows an initial steady speed which was later increased and subsequently assumed a steady speed with little incensement at intervals.

4.2.2.5. Wind Direction on Readings and Analysis

Wind direction readings retrieved from the developed weather station as monitored via the Thingspeak platform were extracted into Microsoft Excel as tabulated in Table 4.1

Table 4.5: Wind direction on readings from the developed weather station

Date and Time (minute	ID Number	Wind Direction (0^0)
2021-03-07 14:16:32 UTC	420	-1
2021-03-07 14:17:18 UTC	421	90
2021-03-07 14:18:03 UTC	422	90
2021-03-07 14:18:48 UTC	423	45
2021-03-07 14:19:34 UTC	424	180
2021-03-07 14:20:19 UTC	425	180
2021-03-07 14:21:04 UTC	426	-1
2021-03-07 14:21:49 UTC	427	-1
2021-03-07 14:22:34 UTC	428	270
2021-03-07 14:23:19 UTC	429	225
2021-03-07 14:24:40 UTC	430	-1
2021-03-07 14:25:25 UTC	431	0
2021-03-07 14:26:22 UTC	432	-1
2021-03-07 14:27:08 UTC	433	-1
2021-03-07 14:27:53 UTC	434	-1
2021-03-07 14:28:39 UTC	435	0
2021-03-07 14:29:25 UTC	436	0
2021-03-07 14:30:10 UTC	437	0
2021-03-07 14:30:55 UTC	438	0
2021-03-07 14:31:40 UTC	439	0
2021-03-07 14:32:26 UTC	440	315
2021-03-07 14:33:11 UTC	441	-1

To study the wind direction variation, a graph of wind direction against time was plotted as shown in Figure.

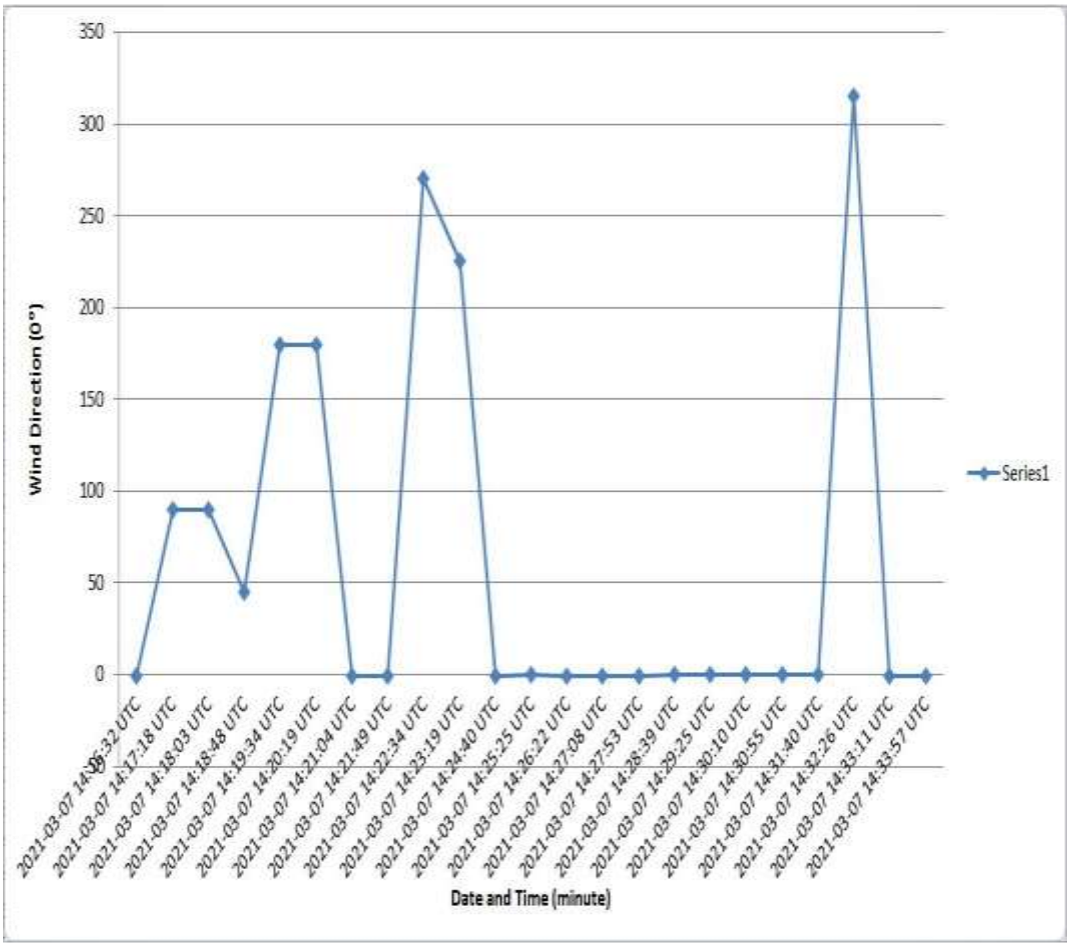


Figure 4.20: Wind direction graph

The graph on Figure 4:20 has wind direction on the vertical axis, while time and date is on the horizontal axis. From the graph, the wind direction was changing continuously, 14:16:32pm to 14:24:40pm but was static between 14:24:40pm to 14:31:40pm and later raises to it peak at 14:32:26pm.

CHAPTER FIVE

5.0 CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In concluding this work, it is very important to recall the objective of this research and to assess how well the objectives were achieved.

The objectives are:

1. design an electronic circuit using a microcontroller as it central processing unit which will retrieve and record atmospheric temperature, relative humidity, atmospheric pressure, wind speed and wind direction from respective electronic sensors and display the values for on-site users.
2. interface a Wi-Fi module into the design to serve as a transmission link between acquired atmospheric data and a webpage for off-site users.
3. utilise miniature, low-cost and locally available components to construct the design, using solar energy as it source of power supply.
4. to evaluate the performance of the constructed weather station if it meets design objectives.

The aim of this work is to develop a portable solar powered, real-time microcontroller-based weather station.

The developed weather station utilised microcontroller as it central processing unit to retrieve analogue and digital signals from the atmospheric sensors through it appropriate input pin connections with the sensors. The retrieved analogue and digital signals by the microcontroller were processed into useful format and displayed on the LCD connected to the output pins of the microcontroller for on-site users of the weather station. The LCD displayed real-time atmospheric temperature, relative humidity, atmospheric pressure, wind speed and wind direction.

For off-site monitoring of atmospheric parameters acquired by the weather station, a Wi-Fi module was interfaced with microcontroller output pins to act as a transmission link between the microcontroller and a webpage. Real-time atmospheric temperature, relative humidity, atmospheric pressure, wind speed and wind direction was monitored on Thingspeak platform in graphical and digital format. The atmospheric data were also archived in excel format for further analysis.

Portability of the weather station was archived by connecting miniature cost effective electronic sensors to the input pins of the microcontroller, and also making using of locally available components. Since the weather station was design for round the clock operation, solar energy was interfaced with the weather station in order to change the backup battery of the weather station.

To evaluate the weather station, it was deployed to the field and real-time atmospheric data were logged. These data were monitored on the LCD for on-site users and on Thingspeak platform for off-site users

With objectives 1,2,3, and 4 fully achieved, the necessary condition to achieving the aim has been satisfied. This work has made available a portable, real-time solar powered microcontroller weather station. For the acquisition of atmospheric parameters.

5.2 Recommendation

In order to further this work, the following suggestions are recommended:

1. Upgrading the weather station by incorporating more parameter sensors so as to measure additional atmospheric parameters.
2. Thorough performance evaluation involving long-term data acquisition and analysis using the developed microcontroller-based weather station.
3. A weather prediction software such as ANN can be programmed into the device to achieve forecasting capability using previous data measured and save in the cloud.

REFERENCES

Abubakar, I. M & Sulaiman, M.B, (2018). Micro-controller based mobile weather monitor system *American Journal of Embedded Systems and Applications*; 6(1): 23-29.

Ami S, Manju K & Shukla K. N (2015). .Experimental exergetic performance evaluation of solar PV module. *International Journal of Scientific and Research Publications:* 5(1):1-9

Anthony U. A, Segun I. P, Onyedika M. C, Abel E. A & Aderemi A. A, (2017). Smart weather station for rural agriculture using meteorological sensors and solar energy. *Proceedings of the World Congress on Engineering* 2017 Vol I WCE 2017, July 5-7, 2017, London, U.K.

Anusha T, Nirmal R, Surya T, Rochan B, & Bijay R, (2016).Wi-Fi based weather information system. *International journal of Science, Technology and Management*: 5(5):269-274

Anuj, K, Singh, I. P, & Sud, S. K (2010). Design and development of mult-channel data logger for built environment. *Proceedings of the international Multi conference of Engineers and Computer Scientist*. 2(5):78-87

Beth M. A, & Kong, L, (2017). Monitoring on wind speed and wind direction. *International Journal of Applied Engineering Research*. 12(24): 14914-14926

Diarah, R. S., Egbune. D, & Adedayo B. A., (2014). Design and implementation of a microcontroller based automatic door and visitors counter. *International Journal of Scientific Research and Education,* 2, 2321-7545. Retrieved from http//ijsea.in.

Dipak V. S., & Ajij, D. S., (2016). Weather monitoring station: A review. *Int. Journal of Engineering Research and Application*: 6(6):55-60.

Devaraju, J. K. R. Suhas, H. K. Mohana, & Vijaykumar A. P, (2015). Wireless portable microcontroller based weather monitoring station. *Journal of the international Measurement Confederation*: (6):189-200

Edgar M. C., & Juan G. R. R. (2015). Portable weather system for measuring and monitoring temperature, relative humidity and pressure based on bluetooth communication. *American Journal of Networks and communication*, 4 (3): 49-53.

Emmanuel, G. D., Stephen B. J., Digima M, & Birma I. H, (2018) Microcontroller based remote Weather Monitoring System.*Journal of Scientific and Engineering Research.* 5(4):276-287.

Gouda, K. C. Preetham, V. R, & Swamy, M. N. S. (2014). Microcontroller based real time weather monitoring device with GSM. *International Journal of Science, Engineering and Technology Research (IJSETR)*.3(7): 1960-1963.

Gauri, S., Pankaja P., Kiran, M., Aarti J, & Karishma, K, (2017). Environment monitoring system based on IoT, *International Journal of Innovative Research in Computer and Communication Engineering*.5(1):1083-1091

Hamid, M. A., & Alaa. H. O, (2015). Floor identification using smartphone barometer sensor for indoor positioning. *International Journal of Engineering Sciences & Research Technology*.4(3):384-391

https.//ams.confex.com.

Ibrahim A. G. (2018). Postgraduate lecture note on “Physics of the lower atmosphere”, Department of Physics, Federal University of Technology, Minna, Nigeria.

Iswanto & Helman M., (2012). Weather monitoring station with remote radio frequency wireless communication. *International Journal of embedded systems and application, Indonesia*, 2 (7): 99-104.

Jacob, T., Mohammed, K. M., Abdullatif, G. R., Abraham, U. U., & Immanuel N. J., (2018). Solar powered DC refrigerator with a monitoring and control system, IEEE PES/IAS Power Africa 5(3):591-594

Jahedul, I., Umme H., Humayun, K., Kazi G.M, Fahmida, A., Farhana, H., Mohammod A. S. H., Mahbubul, H., M., & Abdul, M. C., (2018). Design and development of microcontroller based wireless humidity monitor, *Journal of Electrical and Edlectronics Engineering (IOSR-JEEE), 13 (2), 41-46*

Karthik, K., Suraj, T., Lokesh, K., & Arun, P. (2015). Arduino based weather monitoring System. *International Journal of Engineering Research and General Science, 3(2):*452-458.

Karishma, P., Mansi, M., Rashmi, G., Shraddha, R., Gaurav, G. (2016). Weather Monitoring System using Microcontroller,*International Journal on Recent and Innovation Trends in Computing and Communication*. 4(1): 78-80.

Kamarul, A. N., Chow, C. O., & Mohamad, F. I., (2006) Low-cost microcontroller-based weather monitoring system. *CMU Journal* 5(1):33-38

Kondamudi, S. S. R., & Gupta, A. N. P. S. (2016). IOT based data logger system for weather monitoring using wireless sensor networks. *International Journal of Engineering Trends and Technology (IJETT),* 32(2), 71-75

Kirit V, Jignesh P, HardikP,(2016). Real Time Environmental Data Monitoring with Help of Wi-Fi Module.*International Journal of Advanced Research in Computer and Communication Engineering*. 5(9):158-161

Lay, N. S., Kyaw, S. L., & Hla M. T., (2014). Implementation of microcontroller based sensing unit in transmitter for wireless weather station. *International journal of science, Engineering and Technology Research*. 3 (6): 1613-1616.

Maulik, A. P & Irshad, G. (2018). Internet of things based smart weather monitoring system. *International Conference on Internet of Things and Current Trends in Libraries*. 4(4):24-29

Mohammad, S. A., & Sumit, D. (2015). Design and implementation of an automatic voltage regulator with a great precision and proper hysteresis. *International Journal of Advanced Science and Technology* .75(7):21-32

Nhiverkar .G. S., & Mudholker, R. R. (2011). Data logger and remote monitoring system for multiple parameter measurements applications. *E-Journal of Science and Technology, http:/e-jst.teiath.gr*. 3(6):55-62

Odeyemi, F. M., & Bakare, B. I. (2015). Design and implementation of an improved wind speed meter (anemometer). *Journal of Electronics and Communication Engineering.* 10(1):36-43

Okonkwo, G. N., & Nwokoye, A. O. C., (2014). Analysis of solar energy parameters in Bida, Nigeria, *European Scientific Journal, 10(4):67-83. Retrieved from http://www.eujournal.org*.

Poonam, T., Rahul, M. P., & Sunita, I. P. (2015). Hardware specification for wireless sensor node for real time data acquisition over Thingspeak. *International Journal of Advances in Science Engineering and Technology*. 3(2):47-51

Rafi, S., Bagus, R. R., Satya, H. S., & Rifqi, F. (2018). Arduino based weather monitoring telemetry system using NRF24L01+.*IOP Conf. Series: Materials Science and Engineering*. 3(3):60-69.

Sarmad, N. M., & Forat, F. H. (2017). Design of weather monitoring system using arduino based database implementation. *Journal of Multidisciplinary Engineering Science and Technology (JMEST), 4(4*):7109-7117.

Saleh, U. A., Haruna, Y. S., & Onuigbo, F. I (2015). Design and procedure for stand-alone photovoltaic power system for ozone monitor laboratory at Anyigba, North Central Nigeria. *International Journal of Scientific and Research Publications*. 5(1):41-52

Shewale, S. D., & Gaikwad, S. N (2017). An IOT based real-time weather monitoring system using raspberry Pi. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*. 6(6):30-37

Subair, S., & Abraham, L., (2014). Intelligent pressure measuring system. *International Journal of Computer Science and Mobile Computing*. 3(8):*129 – 134*

Tanmay, P., Shobhit, G., Akash, G., & Thakare Y. B. (2016) Weather monitoring System using Wi-Fi. *International journal of Science and Research*, 5 (11): 891-892.

Tajedinn, A. B., & Abdelrasoul, J. K. A. (2015). Design and implementation of a real-time remote measurement and monitoring of weather parameters system. *International organization of Scientific Research. 5(1):* 70-73

Ughanze, I. J., Ibrahim, A. G., & Eichie, J. O. (2019). Design presentation of solar powered Microcontroller-Based Weather station for the acquisition of atmospheric parameters. *2nd School of Physical Science Biennial International Conference FUTMinna*. pp(1104-1115)

Ukhurebor, .K. E., Abiodun.I. C., Azi, S. O., Otete.I. & Obogai, L .E. (2017). A cost effective weather monitoring device, *Archives of Current Research International Journal 7(4):1-9.*

Walthall, C. L., Hatfield, J., Backlund, P., Lengnick, L., Marshall, E., Walsh, M., & Ziska, L. H. (2012). Climate change and agriculture in the United States: Effects and adaptation (*USDA Technical Bulletin 1935). Washington*, DC: USDA.

Weerasinghe, R. M., Aroos, M. S. M., Pannila, A. S., Jayananda, M .K., & Sonnadara, D. U. J, (2011). Construction of an automated weather station for ground level weather measurements. *Annual Transactions of IESL*. 3(5):450-455

Yahya, S., & Khraisat, L. A. (2012). Design a Wireless Meteorological Station in Jordan. *Published by Canadian Center of Science and Education. 5(1): 87-96.*

(www.ucsusa.org/smartclimatchoice).

Appendix I

Microcontroller Programme Code (C-Language)

```
#include <SFE_BMP180.h>
#include <Wire.h>
#include <DHT.h>
#include <SPI.h>
#include <Adafruit_GFX.h>
#include <Adafruit_PCD8544.h>

#define DC_VOLTAGE_PIN          A0
#define SE_DIRECTION            A1
#define E_DIRECTION             A2
#define NW_DIRECION             A3
#define W_DIRECION              A4
#define NE_DIRECION             A5
#define S_DIRECION              A6
#define SW_DIRECION             A7
#define N_DIRECION              A8
#define RELAY                   A9
#define WindSpeedPin            A14
#define DHTPIN                  7
#define DHTTYPE DHT22
#define ALTITUDE 1655.0 // Altitude of SparkFun's HQ in Boulder, CO. in meters

DHT dht(DHTPIN, DHTTYPE);
SFE_BMP180 pressure;

// Hardware SPI (faster, but must use certain hardware pins):
// SCK is LCD serial clock (SCLK) - this is pin 13 on Arduino Uno
// MOSI is LCD DIN - this is pin 11 on an Arduino Uno
// pin 5 - Data/Command select (D/C)
// pin 4 - LCD chip select (CS)
// pin 3 - LCD reset (RST)
Adafruit_PCD8544 display = Adafruit_PCD8544(A10, A11, A12);
//Adafruit_PCD8544 display = Adafruit_PCD8544(D/C, CS, RST);
// Note with hardware SPI MISO and SS pins aren't used but will still be read
// and written to during SPI transfer.  Be careful sharing these pins!

void initMyDisplay(void);
void getTemperature_AND_Humidity(void);
void getBatteryVoltage(void);
void printPressure(double AP, double RP);
void getPressure(void);
void UpdatLCD(void);
void EspInit(void);
intgetWindDirection(void);
booleanthingSpeakWrite(float value1, float value2, float value3, float value4, int
    value5);
```

```
String apiKey = "VX6I5D809ZMD2O58"; //"2MPR1ZIF1K6PMFM5";
String ssid="etisalatMi-Fi";
String password ="13766212";

intTemp_AND_Humid_SampleInterval = 0;
intPressureSampleCounter = 0;
intUpdatLCD_TimeCounter = 0;
intBatterySampleCounter = 0;
intDataSendingInterval = 0;
intDsampleTimer = 0;
booleanDataSentFlag = false;

unsigned intchargingTimer = 0;
booleanfullCharge = false;

double AbsolutPressure = 0;
double RelativePressure = 0;
float humid = 0;
float temp = 0;
float BatteryVoltageLevel = 0.9;
float WindSpeed = 0;
unsigned intWindSpeed_ADC = 0;
intWindDirection = 0;
intWindSpeedSampleCounter = 0;

void setup(){
delay(10);
pinMode(RELAY, OUTPUT);   digitalWrite(RELAY, LOW);
Serial.begin(9600);
  Serial1.begin(115200);
display.begin();
display.setContrast(60);
dht.begin();
  if (!pressure.begin()){
        display.clearDisplay();
        display.setTextSize(1);              display.setTextColor(BLACK);
        display.setCursor(0,14);        display.println("BMP180 ERROR");
        display.setTextSize(1);              display.setTextColor(BLACK);
        display.println("Initialisation");
        display.display();
        Serial.println("PREASURE SENSOR NOT WORKING.");
        Serial.println("xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx");
        while(1); // Pause forever.
  }

Serial.println("Serial COM. Initialisation Complete.");
delay(5000);
initMyDisplay();

}
```

```
void loop(){

if(Temp_AND_Humid_SampleInterval< 2000)
        Temp_AND_Humid_SampleInterval++;
  else getTemperature_AND_Humidity();

if(PressureSampleCounter< 5000) PressureSampleCounter++;
  else getPressure();

if(BatterySampleCounter< 50) BatterySampleCounter++;
  else getBatteryVoltage();

if(UpdatLCD_TimeCounter< 500) UpdatLCD_TimeCounter++;
  else UpdatLCD();

if(DsampleTimer< 1000) DsampleTimer++;
else{
        WindDirection = getWindDirection();
                DsampleTimer = 0;
                if(WindDirection == 0) Serial.println("N Direction");
                  else if(WindDirection == 45) Serial.println("NE Direction");
                  else if(WindDirection == 90) Serial.println("E Direction");
                  else if(WindDirection == 135) Serial.println("SE Direction");
                  else if(WindDirection == 180) Serial.println("S Direction");
                  else if(WindDirection == 225) Serial.println("SW Direction");
                  else if(WindDirection == 270) Serial.println("W Direction");
                  else if(WindDirection == 315) Serial.println("NW Direction");
                  else Serial.println("ERROR Direction");
     }

WindSpeed_ADC = WindSpeed_ADC + analogRead(WindSpeedPin);
WindSpeedSampleCounter++;
if(WindSpeedSampleCounter>= 5){
WindSpeed = (float)WindSpeed_ADC / (float)WindSpeedSampleCounter;
WindSpeed = ((float)16.7 * WindSpeed) / 38.0;
WindSpeedSampleCounter = 0;
WindSpeed_ADC = 0;
   }

if(DataSendingInterval< 30000) DataSendingInterval++;           // After every 30
        Sec.
else{
        DataSentFlag = thingSpeakWrite(temp, humid, AbsolutPressure, WindSpeed,
        WindDirection);
                if(DataSentFlag == true) DataSendingInterval = 0;
                 else DataSendingInterval = 25000;
     }

chargingTimer++;
```

```
if(chargingTimer>= 1000){
chargingTimer = 0;
if(fullCharge == false &&BatteryVoltageLevel<= 14.4) digitalWrite(RELAY, HIGH);
if(fullCharge == false &&BatteryVoltageLevel>= 14.4){
digitalWrite(RELAY, LOW);
fullCharge = true;
      }
if(fullCharge == true &&BatteryVoltageLevel<= 12.8) fullCharge = false;
   }

delay(1);
}

void initMyDisplay(void){
        display.clearDisplay();
        display.setTextSize(1);                 display.setTextColor(BLACK);
        display.setCursor(0,14);        display.println("Initializing");
        display.setTextSize(1);                 display.setTextColor(BLACK);
        display.println("Wi-Fi Module..");
        display.display();
        Serial.println("\n\nDisplayInitialisation Complete.");

        delay(1000);
        EspInit();

        display.clearDisplay();
        display.setTextSize(1);                 display.setTextColor(BLACK);
        display.setCursor(0,14);        display.println("Initialisation");
        display.setTextSize(1);                 display.setTextColor(BLACK);
        display.println("   complete   ");
        display.display();
        delay(5000);
        Temp_AND_Humid_SampleInterval = 2000;
        PressureSampleCounter = 5000;
        UpdatLCD_TimeCounter = 500;
        display.clearDisplay();

        Serial.println("\nAllInitialisations Completed.");
}

intgetWindDirection(void){
        float Vlimit = 1.5;
        if(((float)analogRead(N_DIRECION)/1023.0) * 5.0 <= Vlimit ) return 0;
        else  if(((float)analogRead(NE_DIRECION)/1023.0) * 5.0 <= Vlimit ) return 45;
        else  if(((float)analogRead(E_DIRECTION)/1023.0) * 5.0 <= Vlimit ) return 90;
        else  if(((float)analogRead(SE_DIRECTION)/1023.0) * 5.0 <= Vlimit ) return
        135;
        else  if(((float)analogRead(S_DIRECION)/1023.0) * 5.0 <= Vlimit ) return 180;
        else  if(((float)analogRead(SW_DIRECION)/1023.0) * 5.0 <= Vlimit ) return
        225;
```

```
        else if(((float)analogRead(W_DIRECION)/1023.0) * 5.0 <= Vlimit ) return
        270;
        else if(((float)analogRead(NW_DIRECION)/1023.0) * 5.0 <= Vlimit ) return
        315;
        else return -1;
}

void getTemperature_AND_Humidity(void){
        humid = dht.readHumidity();
        temp = dht.readTemperature();
        Temp_AND_Humid_SampleInterval = 0;
}

void getBatteryVoltage(void){
        static float Battery_Voltage_Read = 0;
        static uint8_t Battery_Voltage_Sample_Counter = 0;

        Battery_Voltage_Read = Battery_Voltage_Read +
        (float)analogRead(DC_VOLTAGE_PIN) * 0.04887586;
        Battery_Voltage_Sample_Counter++;
        if (Battery_Voltage_Sample_Counter>= 20){
                BatteryVoltageLevel = Battery_Voltage_Read / 20.0;
                BatteryVoltageLevel = BatteryVoltageLevel - 0.9;
                Battery_Voltage_Read = 0;
                Battery_Voltage_Sample_Counter = 0;
        }

        BatterySampleCounter = 0;
}

void getPressure(void){
        double myTemp = 0;
        double myAltitude = 0;
        char status;

        status = pressure.startTemperature();
        if (status != 0){
                delay(status);  // Wait for the measurement to complete:
                status = pressure.getTemperature(myTemp);
                if (status != 0){
                        // Temp. was read successfully and stored in myTemp. it can be
        printed if required.
                        // Start a pressure measurement:
                        status = pressure.startPressure(3);
                        if (status != 0){
                                delay(status);  // Wait for the measurement to complete:
                                status = pressure.getPressure(AbsolutPressure,myTemp);
                                if (status != 0){
                                        // Absolute Pressure was read successfully and
        stored in "AbsolutePressure" variable
```

```
// Absolute Pressure is use along with ALTITUDE to compute Relative Pressure.
RelativePressure = pressure.sealevel(AbsolutPressure,ALTITUDE);
myAltitude = pressure.altitude(AbsolutPressure,RelativePressure); // compute Altitude in meters.
}else {AbsolutPressure = 0; RelativePressure = 0;}
}else {AbsolutPressure = 0; RelativePressure = 0;}
}else {AbsolutPressure = 0; RelativePressure = 0;}
}else {AbsolutPressure = 0; RelativePressure = 0;}

PressureSampleCounter = 0;
}

void UpdatLCD(void){

//****************************************** TEMPERATURE AND HUMIDITY ******************************************
display.clearDisplay();
display.setCursor(0, 0); display.setTextSize(1); display.setTextColor(BLACK);
display.print("Temp:"); display.print(temp); display.print("C ");
display.setCursor(0, 9); display.setTextSize(1); display.setTextColor(BLACK);
display.print("Humid:"); display.print(humid); display.print("% ");

//****************************************** PRESSURE ******************************************
display.setCursor(0, 19); display.setTextSize(1); display.setTextColor(BLACK);
display.print("A.P:"); display.print(AbsolutPressure); display.print("hPa ");
//display.print("RP:"); display.print(RP); display.print("hPa ");

//****************************************** WIND SPEED AND DIRECTION ******************************************
display.setCursor(0, 28); display.setTextSize(1); display.setTextColor(BLACK);
display.print("Speed:"); display.print(WindSpeed); display.print("K/Hr");

display.setCursor(0, 37); display.setTextSize(1); display.setTextColor(BLACK);
display.print("Direction:"); display.print(WindDirection);
display.display();
UpdatLCD_TimeCounter = 0;
}

void EspInit(void){
```

```
	String EspReply = "";
	boolean flag = false;

	Serial.println("\n\nInitializing Wi-Fi Module.");
	delay(1000);

Serial.println("Set ESP8266 as a Client:");
	delay(10);
	Serial1.println("AT+CWMODE=1");                                    // set
	esp8266 as client
	EspReply = getRespond();
	Serial.println("RESPOND TO CLIENT:");
	Serial.println(EspReply);
	delay(100);

	do{
		Serial.print("\nConnecting to: "); Serial.print(ssid); Serial.print(", ");
	Serial.print(password); Serial.println(".");
      Serial1.println("AT+CWJAP=\""+ssid+"\",\""+password+"\"");   // set your home
	router SSID and password
		delay(10);
		EspReply = getRespond();

		Serial.println("RESPOND TO THE CONNECTION:");
		Serial.println(EspReply);
		Serial.println("");

		if(EspReply.indexOf("OK") > -1) flag = true;
		else EspReply = "";

	}while (flag == false);

	Serial.println("ESP8266 was Successfully Connected to:");
	Serial.println(ssid);
	Serial.println("Wireless Network\n\n\n");
}

booleanthingSpeakWrite(float value1, float value2, float value3, float value4, int
	value5){
	String EspReply = "";
	Serial.println("\n\n\n&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
	&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&");
	Serial.println("Try New TCP Connection with api.thingspeak.com");
	delay(10);

	String cmd = "AT+CIPSTART=\"TCP\",\"";              // TCP connection
	cmd += "184.106.153.149";                            // api.thingspeak.com
	cmd += "\",80";
	Serial1.println(cmd);
```

```
        EspReply = getRespond();
        Serial.println("SENT COMMAND:");
        Serial.println(cmd);
        Serial.println("RESPOND TO THE SENT COMMAND:");
        Serial.println(EspReply);

        delay(1000);

        if(Serial1.find("Error")) return false;

        Serial.println("\nSENDING MY DATA:");

        String getStr = "GET /update?api_key=";   // prepare GET string
        getStr += apiKey;
        getStr +="&field1=";
        getStr += String(value1);
        getStr +="&field2=";
        getStr += String(value2);
        getStr +="&field3=";
        getStr += String(value3);
        getStr +="&field4=";
        getStr += String(value4);
getStr +="&field5=";
getStr += String(value5);
        // ...
        getStr += "\r\n\r\n";

        // send data length
        cmd = "AT+CIPSEND=";
        cmd += String(getStr.length());
        Serial1.println(cmd);

        Serial.println("SENDING DATA LENGHT:");
        Serial.println(cmd);

        delay(1000);

        if(Serial1.find(">")){
          Serial1.print(getStr);
        Serial.println("DATA WAS SENT SUCCESSFULLY.");
        Serial.println(getStr);
          return true;
        }
        else{
                  Serial1.println("AT+CIPCLOSE");
                Serial.println("ERROR WHILE SENDING DATA.");
                  return false;
           }
}
```

```
String getRespond(void){
	String EspReply = "";

	while(Serial1.available() == 0);
	EspReply = Serial1.readString();
	return EspReply;
}
```

Appendix J

PROJECT BILL OF QUANTITY

S/N	Component	Quantity	Unit cost	Cost
1.	Battery 12V 45AH	1	30,000	30,000
2.	Solar-panel	1	15,000	15,000
3.	Charge controller	1	15,000	15,000
4.	Temperature and RH sensor	3	2,000	6,000
5.	Atmospheric pressure sensor	2	4,000	8,000
6.	Arduino Nano	3	3,000	9,000
7.	Node MCU	2	6,000	12,000
8.	Wi-Fi router(modem)	1	16,000	16,000
9.	LCD	4	5,000	20,000
10.	Variable resistor	3	1,000	3,000
11.	Bulck converter	3	2,500	7,500
12.	Metallic Stand	1	10,000	10,000
13.	jumper	1 lot	500	500
14.	anemometer	1	8,000	8,000
15.	Wind vane	1	13,000	13,000
16.	magnetometer	2	4,000	8,000
17.	Connecting cables		700 per yard	2,100
18.	Light indicators	3	100	300
19.	Casing	1	9,000	9,000
20.	Iot And Thingspeak interface design with hardware		10,000	10,000
21.	Data		1,500	1,500
22	Support of Labour		50,000	50,,000
			TOTAL	**₦2,61,400**