DESIGN AND CONSTRUCTION OF A MICROCONTROLLER BASED AC/DC AUTOMATIC HAND DRYER

BY

EGBORO GABRIEL O

(MAT.NO: COT/0410/2014)

THIS PROJECT REPORT IS SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ELECTRONICS ENGINEERING, COLLEGE OF TECHNOLOGY FEDERAL UNIVERSITY OF PETROLEUM RESOURCES P.M.B 1221 EFFURUN DELTA STATE NIGERIA

IN PARTIAL FULFILLMENT FOR THE AWARD OF BACHELOR OF ENGINEERING DEGREE IN ELECTRICAL ELECTRONICS ENGINEERING

NOVEMBER, 2019.

DECLARATION

I hereby declare that this project report is original and have not been submitted in part or full to any other institution for the award of a Bachelor of Engineering degree or any other award.

…………………………….

EGBORO GABRIEL O

MAT.NO: COT/0410/2014

NOVEMBER, 2019.

CERTIFICATION

This is to certify that the design and construction of this microcontroller based automatic hand dryer was carried out by **EGBORO GABRIEL O** under the supervision of **ENGR. DR .G .O .UZEDHE** and has been duly approved for acceptance by the Department of **ELECTRICAL ELECTRONICS ENGINEERING.**

………………………………

ENGR. DR. G. O. UZEDHE ……………………..

(Project Supervisor) Date

……………………………

ENGR. DR. G.O. UZEDHE …..…………………

(Head of Department) Date

DEDICATION

This project is dedicated to God Almighty.

ACKNOWLEDGEMENT

I wish to express my sincere gratitude to God Almighty and to my parents for their financial support and prayers.

To my Project supervisor Engr. Dr G. O. Uzedhe and all my lecturers in the department who prepared me academically throughout the duration of my study and to all my departmental mates for their advice and encouragement**.**

ABSTRACT

This project presents the design and implementation of a microcontroller based AC/DC automatic hand dryer. Hand drying is a necessary culture that needs everybody to be involved for effective hand hygiene after hand washing. However, the hand drying methods are not usually effective because of the nature of materials used. This project has provided an electrical/electronic means of automatically drying hands using AC/DC heating system with air that blows the hand. The system entails the logic unit for control and a charging unit which charges the batteries through a circuitry when there is power from AC mains in order to ensure effective operation of the hand dryer system. The system also incorporates a liquid crystal display (LCD) to display the battery level and also the state of the hand dryer at a particular point in time, either in idle state or active state. The hand drying system designed in this project was able to dry single wet hands in approximately 45 seconds. Based on the battery power provided the system can last for 2.6 hours drying approximately 208 hands.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

DC- Direct Current

AC- Alternating Current

MCU- Microcontroller

LCD- Liquid Crystal Display

I/O- Input/output

CHAPTER ONE

INTRODUCTION

1.1 Background Study

The basic utility of a hand dryer is to dry hands using an electric source. Hand hygiene is a very important aspect in preventing spread of communicable diseases. For instance, a public restroom is a place where the presence of disease-causing microorganisms is very high as compared to other places (Sonkushre, 2018). Since many disease-causing germs spread through touch, hand dryers were devised so that people can dry their hands without actually touching the roll of paper towel available for common use in public restrooms. Hand drying machines does not only help in maintaining hygiene but also help in reducing waste produced by the use of paper towels.

A typical air dryer lasts seven to ten years (Kenney, 2017), so any environmental production costs is negligible over its lifetime. Environmentalists are chiefly concerned with the electricity required to operate them. The average air dryer uses about 2,200 watts of power while running. If you spend 30 seconds drying your hands, you use approximately 0.018 kilowatt hours of electricity (Kenney, 2017). Although the installation of an air dryer is more expensive than that of a paper towel dispenser, the air dryer will save more money over time. The cost of running an air dryer for 30 seconds is approximately 0.2 cents per use, according to the website Green and saves, compared with 1.5 cents on average for paper towels. Green and Save estimates an air dryer can save a business $475 over five years compared with paper towels. Indirect costs associated with paper towels include increased bathroom cleaning, trash removal and dispenser filling. (Kenney, 2017)

Earlier, hand dryers were switched on using a button. When the button was pressed, the device started blowing warm air for drying hands. But new-age hand dryers have infrared motion sensors installed in them. The device detects motion and automatically turns on when hands are placed for drying. Similarly, it automatically turns off when hands are moved away from the dryer, thereby preventing wastage of electricity. This also makes the hand-dryer more hygienic to use because you do not have to push a button that countless other people have also touched.

This system design addresses the issue of unstable power supply by creating provision for a back-up source (battery) to power the hand-dryer while there is outage from AC mains. The relevant circuitry for charging the battery while there is power from the AC mains is also provided in the system design. The system is directly powered from AC mains when there is supply (mains mode) and switches automatically to the back-up source (battery mode) when there is an outage from the AC source. In effectively accomplishing this configuration, the use of microcontroller and discrete components where employed.

1.2 Problem Statement

In a developing country, taking Nigeria as a case study where there is unstable supply of electricity, hand dryers would be less effective in promoting good hand hygiene. Water used in washing our hands can be readily available because they can be pumped and stored in overhead tanks when there is power and can be used when there is a power outage. But hand dryers become useless when there is outage of electric power.

1.3 Aims and Objectives

The aim of this project is to design and develop a microcontroller based AC/ DC automatic hand dryer.

The specific objectives are;

1. To design a hand dryer system with the use of a microcontroller for efficiency and flexibility.

2. To design a hand dryer system that is AC/ DC powered.

3. To design a hand dryer system with a liquid crystal display (LCD), for the presentation of information.

4. To design a hand dryer system that switches automatically with the use of a motion sensor.

1.4 Project Scope

This project work focuses on the design and implementation of a DC automatic hand dryer. It covers the use of the microcontroller and discrete components used in the design of circuitry needed in the actualization of this project. This project does not cover the design of the blower used in the hand-dryer.

1.5 Applications

DC hand dryers can be used in restrooms, restaurants/bars, laboratories, hospitals, events centers and other places where applicable to promote good hand hygiene effectively.

1.6 Outline of the Remaining Chapters

Chapter two is basically literature review and theories relevant to the project. Chapter three is the main body of my project work, comprising of methodology and design analysis. Chapter four contains implementation, the laboratory/workshop tests, results and observation. Chapter five is the concluding part of the project work where conclusions and recommendations are made.

CHAPTER TWO
LITERATURE REVIEW

This chapter describes and discusses the research from various sources such as textbooks, articles and the internet. It consists of the history of hand dryers, related works, and some important details on the main components involved in implementing this project.

2.1 HISTORY OF HAND-DRYER

The earliest hand dryer was patented in 1921 by R.B.Hibbard, D.J.Watrous and J.G Bassett for the Air-dry Corporation of Groton New York. (Christinab, 2017) This machine was sold as a built in model or freestanding floor unit that consisted of an inverted blower (much like a handheld blow dryer) that was controlled by a floor pedal. Known as “Air-dry the Electric Towel”, these units were used in restrooms, barber shops and factories. Air-dry corporation moved to Chicago and San Francisco in 1924 to centralize their distribution (Mlekei, 2019).

The hand dryer was later popularized in 1948 by George Clemens (Gagnon, 2011). Mitsubishi electric introduced a new type of hand dryer that blows jets of air on both sides of the hand, pushing the water off rather than evaporating it (Tokyo, 2017). Clemens’ work on the electric hand dryer led to the founding of World Dryer a commercial hand dryer manufacturing company, the company that most commonly produced the devices for decades. But while it had a lock on the larger market, the company’s slow rate of innovation cut up with it, and soon competitors started to loosen its grip. (Smith, 2015)

The problem was that while early hand-dryers worked, they were loud and fairly inefficient, taking nearly a minute to dry hands. But with time companies were slowly introducing new methods of automated hand-drying to the public at large.

2.2 RELATED WORKS

Ansari et al., 1991 carried out a comparative study of the efficiencies of paper, cloth and electric warm air drying in eliminating viruses and bacteria from washed hands and discovered that, irrespective of the hand-washing agent used, the electric air drying system produced the highest reduction in numbers of both viruses and bacteria. The electric drying system here is AC controlled and makes use of a manual switch which reduces the effectiveness of the system with respect to hand hygiene. Since the switch is to be put off manually after drying the hands, there is a possibility of contacting disease through touch.

Hareendran, 2014 constructed an automatic hand dryer machine with temperature display, to eliminate the dangers and problems associated with the manual process, like contracting diseases with the use of same hand towels by many people in restaurants and our health care facilities. He designed his system with active and passive component sensors (LM35 and LDR). His work was purely hand-wired, no use of micro program controls. Purely hand-wired circuit increases the complexity and cost of the system.

Stephen, 2018 designed a microcontroller based automatic hand dryer system. In his design he used both hardware and software, the hardware components are coordinated by AT89S51 microcontroller chip while the programs are embedded in the chip. This system designed is purely AC controlled making it inactive when there is a power outage from the AC mains supply.

Cunrui et al., 2012 carried out a systematic review of research works and analyzed the hygienic efficacy of different hand drying methods. After the analysis, it was discovered that the use of hand dryers were more effective than the use of paper towels.

Montalbol et al., 2011 carried out a study that compared seven different hand-drying methods by performing life cycle analysis on each system. Their work showed that the hot air dryer is most suitable.

Rochester Institute of Technology New york investigated the environmental impacts of paper towels versus t (Christinab, 2017)hose of hand dryers in a college campus via life cycle analysis. Their findings showed that the use of a hand dryer reduced the environmental impact and carbon footprint and also more economical.

2.3 REVIEW OF SOME COMPONENTS USED

Some components used in implementing this project are reviewed and discussed briefly.

2.3.1 BATTERY

Batteries are a collection of one or more cells whose chemical reactions create a flow of electrons in a circuit. All batteries are made up of three basic components: an anode (the negative side), a cathode (the positive side), and some kind of electrolyte (a substance that chemically reacts with the anode and cathode). When the anode and cathode of a battery is connected to a circuit, a chemical reaction takes place between the anode and the electrolyte. This reaction causes electrons to flow through the circuit and back to the cathode where another chemical reaction takes place. When the material in the cathode or anode is no longer able to be used in the reaction, the battery is unable to produce electricity. It can therefore be charged if it is a secondary battery, or should be disposed if it is a primary battery. (Hymel, 2013)

2.3.1.1 DEEP CYCLE BATTERY

Deep cycle batteries are designed to provide a small amount of power continuously for many hours. Batteries are classed according to how much electricity they can store. The measure used is the ampere-hour (Ah). If a battery delivers 1A of current continuously for 100 hours it has provided 100 Ah. If a battery delivers 10A continuously for 10 hours, it has also delivered 100 Ah. If 5 A is delivered continuously for 20 hours that is 100 Ah too. Ampere-hours are equal to the continuous current being taken from the battery times the number of hours it is delivered. If a current of 7 A is delivered for 6 hours that will be $6 \times 7 = 42$Ah. (Ofualagba, 2019)

2.3.2 AIR HEATER

An air heater works by passing air across a heated element to elevate the temperature of the air. Every air heater is comprised of two main components:

1. A source of air flow

   The source of air can be a fan, blower, or heat compressor. The best source depends on the application and type of heater. The source of air helps in circulating the heated air.

2. A heating element

It is normally a coil of wire with a high resistivity. By passing an electrical current through the wire, heat is produced. The configuration of the resistance wire is dependent on the purpose of
the heater. The element should not be powered without air flow, or it may overheat and burn out. (Stanmech, 2017)

2.3.3 LIQUID CRYSTAL DISPLAY (LCD)

A Liquid crystal display is an electronic display module which uses liquid crystal to produce a visible image. LCD modules are very commonly used in most embedded projects, the reason being its cheap price, availability and programmer friendly. In relation to this project, a 16x2 LCD is used. 16x2 LCD is named so because it has 16 Columns and 2 Rows .So it will have (16x2=32) 32 characters in total and each character is made up of 5x8 Pixel Dots. The LCD should be instructed about the position of the pixels. Hence it would be a hectic task to handle everything with the help of a microcontroller unit, hence an interface IC like HD44780 is used, which is mounted on the back side of the LCD Module itself. The function of the IC is to get the Commands and Data from the Microcontroller unit and process them to display meaningful information onto our LCD Screen.

The 16x2 LCD consists of 16 pins as described below;

- Pin 1(GND): This pin helps in connecting the LCD to the Ground.
- Pin 2(VCC): The VCC pin is used to supply the power to the LCD
- Pin 3(VEE): This pin is used for adjusting the contrast of the LCD by connecting the variable resistor in between the VCC and Ground.
- Pin 4(RS): The RS is known as register select and it selects the Command/Data register. To select the command register the RS should be equal to zero. To select the Data register the RS should be equal to one.
- Pin 5(R/W): This pin is used to select operations of Read/Write. To perform the write operations the R/W should be equal to zero. To perform the read operations the R/W should be equal to one.
- Pin 6(EN): This is an enable signal pin if the positive pulses are passing through a pin, then the pin function as a read/write pin.

- Pin 7(DB0 to DB7): The pin 7 contains 8 pins which are used as a data pin of LCD.
- Pin15(LED+): This pin is connected to VCC and it is used for the pin 16 to set up the glow of backlight of LCD
- Pin16 (LED-): This pin is connected to Ground and it is used for the pin 15 to set up the glow of backlight of the LCD.

2.3.4 MICROCONTROLLER PIC18F2520

The PIC18F series microcontroller is a chip designed to perform various functions, it converts analogue data to digital values and carries out operation on the values, and it performs data storing, timer and counter operations and many other applications. The PIC18F2520 from microchip introduces design enhancements that make this microcontroller a logical choice for many high-performances and power sensitive operations.

Some special features of this microcontroller are;

- Enhanced Addressable USART: This serial communication module is capable of standard RS-232 operation and provides support for the LIN bus protocol. Other enhancements include automatic band rate detection and a 16-bit Band Rate Generator for improved resolution. When the microcontroller is using internal oscillator block, the EUSART provides stable operation for applications that talk to the outside world without using an external crystal (or its accompanying power requirement)
- I/O ports (3 bidirectional ports on 28-pin devices)
- Flash program memory 32 Kbytes
- 10 bit A/D Converter: This module incorporates programmable acquisition time, allowing for a channel to be selected and a conversion to be initiated without waiting for a sampling period and thus, reducing code overhead
- Wide operating voltage range ; 2.0V to 5.5V

CHAPTER THREE

METHODOLOGY AND SYSTEM DESIGN

3.1 METHODOLOGY

In designing the hand dryer system, various methods were used for actualization of the system. Choices were made on what software component, hardware component and the mode in which the system would operate. These choices where made based on some certain criteria such as reliability, simplicity, maintainability and flexibility. Methods such as the use of motion sensor and relays for automatic switching and DC power application were adopted for this project.

3.2 SYSTEM DESIGN ANALYSIS

The system design analysis is the process conducted in order to break the system down into its components to ensure that all the components of the system work efficiently to accomplish their purpose. It shows the various units of the hand dryer system in order to have more insight into the system. This system was broken down into different units as in Figure 3.1.

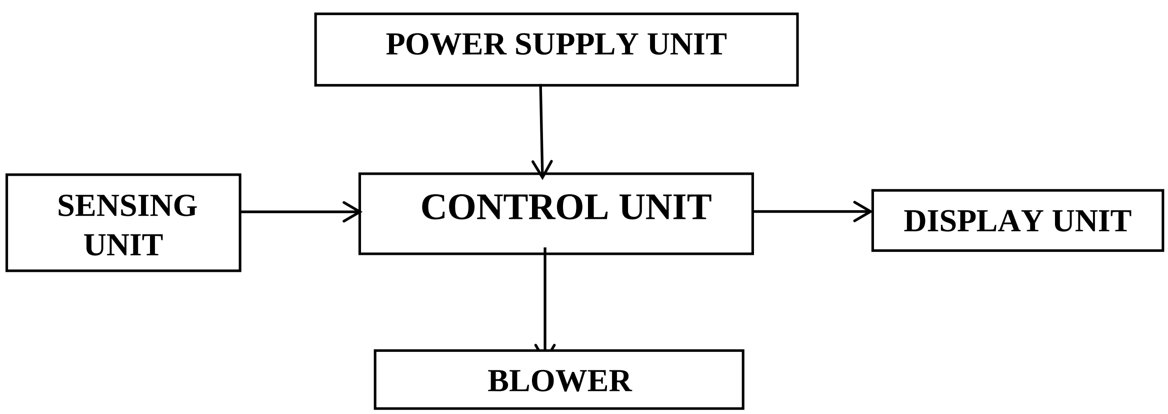


Figure 3.1: Block diagram of the hand dryer system

Design analysis were carried out for the following hardware units of the DC hand dryer system

- The Logic or control circuit
- Sensing unit

- Power supply unit
- Display unit
- Blower unit

3.2.1 Control Unit

The control or logic unit of the system functions to control the ON/OFF operation of the blower with respect to sensor inputs. It comprise of the microcontroller PIC18F2520. The microcontroller is the brain of the hand dryer system. It receives information from the sensing unit and makes decision based on its predefined knowledge base. In this system, the microcontroller is responsible for the analog to digital conversions such as the battery level sensing and detection of rectified voltage from the mains.

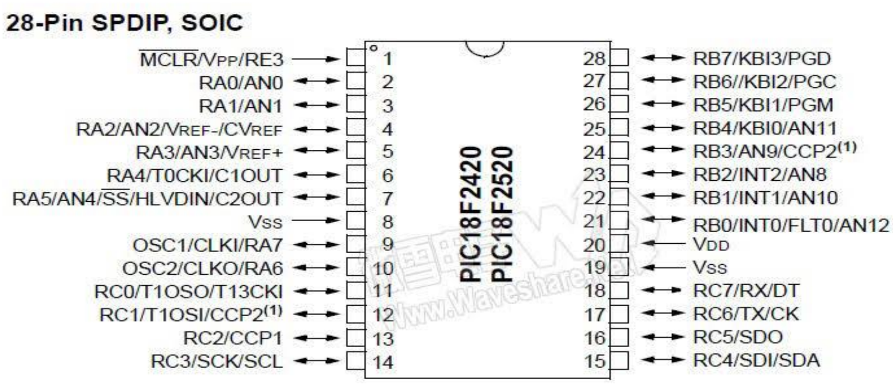


Fig 3.2: Pin out of the PIC18F2520 microcontroller

3.2.2 SENSING UNIT

In detecting motion of the hands, a HC-SR04 ultrasonic sensor as shown in Figure 3.4 was used. The HC-SR04 ultrasonic distance sensor is a sensor used for detecting the distance to the hands. It consists of a transmitter, receiver and a control module as shown in Figure 3.3. The transmitter emits a high frequency ultrasonic sound which bounces off when the hands are close to the vent of the blower and the receiver listens to any return echo. That echo is then processed by the control circuit to calculate the time difference between the signal being transmitted and received. This time can subsequently be used along with some mathematical calculation to determine the distance between the sensor and the reflecting object (hands). The maximum sensing distance of the sensor as programmed is 10cm.

The formula is given by;

Distance to object = T x Speed of sound / 2

T= time between when an ultrasonic wave is emitted and when it is received

Scientists have determined that the speed of sound is approximately 341m/s in air

Division by 2 is because the sound wave has to travel to the object and back.

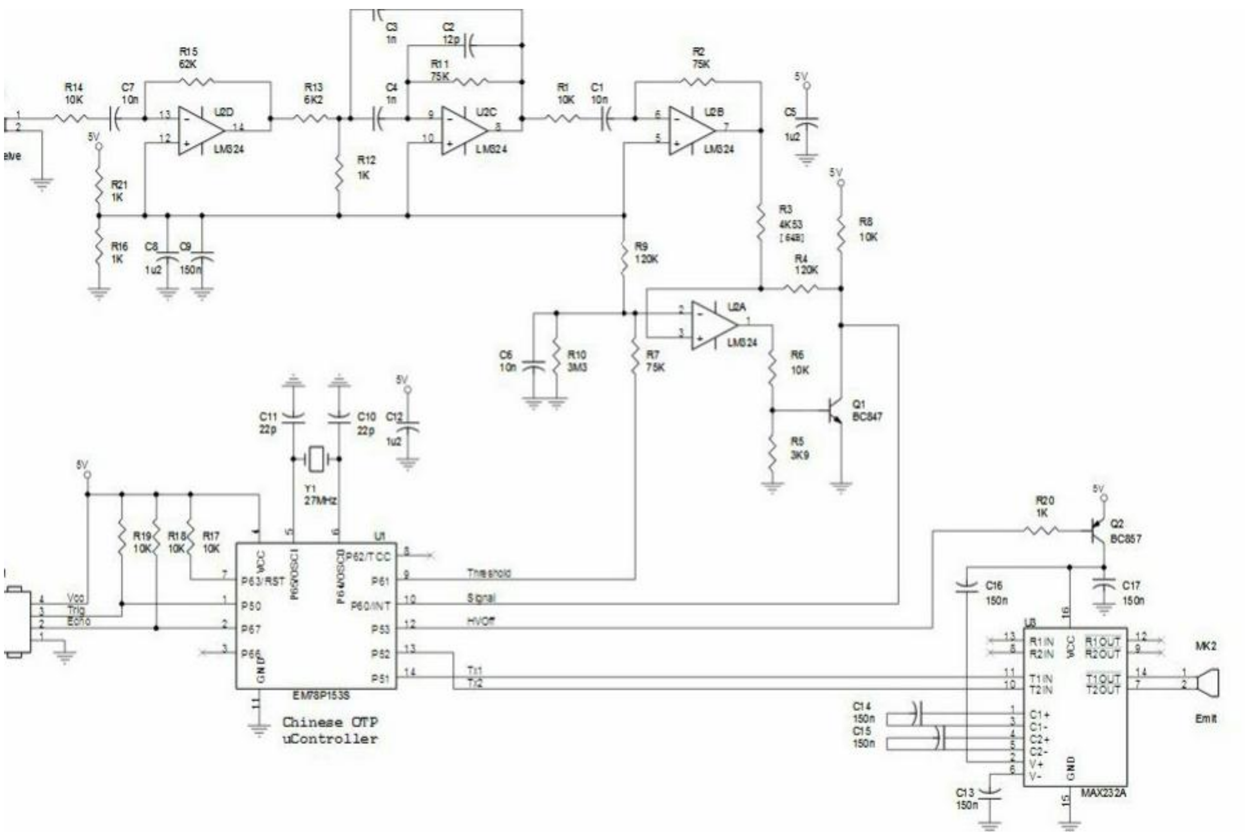


Fig 3.3: Circuit diagram of the HC SR04 Sensor

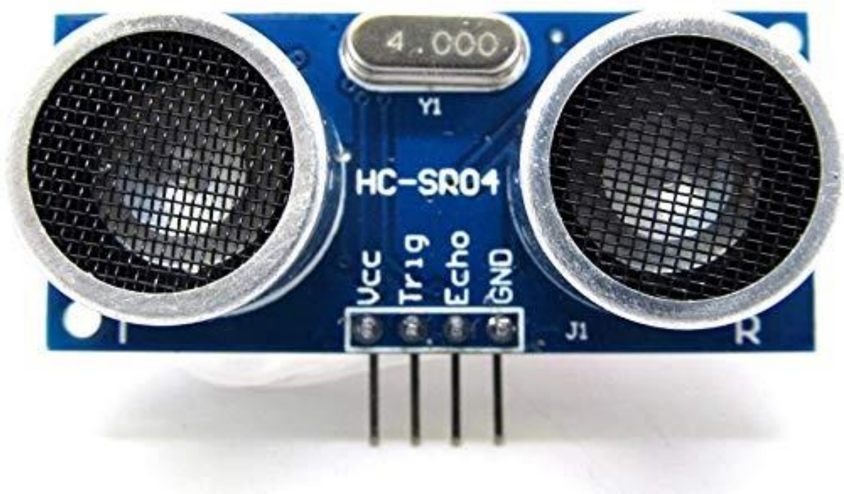


Fig 3.4: HC-SR04 motion sensor

3.2.3 POWER SUPPLY UNIT

The power supply circuit is the circuit which provides the desired 5VDC and 12VDC voltage to run the circuits. The hand-dryer system designed is an AC/DC system which is powered from rectified mains voltage and four 12volts, 12AH batteries connected in parallel to give an output of 12volts, 48AH. The batteries used are charged from a charging unit.

AC current is one which is constantly changing. The current flows in one direction and then in the opposite direction again and again, continuously. In Nigeria, the AC supply mains voltage is 220volts with a frequency of 50 hertz

The AC mains feed the charging unit and in turn, the charging unit does the rectification thereby providing a constant DC voltage which charges the batteries and powers the blower at the same time as shown in Figure 3.6. When the batteries are fully charged, the charging unit cuts off automatically from charging the batteries and then supplies the blower only. When the mains go off, the power supply to the blower switches to the batteries with the aid of a relay.

Maximum current of air heater blower=18amps

Voltage of Air heater blower=12VDC

Air heater blower power consumption=Current*Voltage **(1)**

=18*12=216watts

Battery capacity;

Four 12v, 12Ah deep discharge batteries were readily available. Since the system is a 12v system, for maximum power efficiency the batteries where connected in parallel resulting to an output voltage of 12v and an output current of 48Ah.

Watt hour = voltage x amps hour **(2)**

12v*48Ah=576wh

To determine the time the system would last on battery power, we divide the result of equation 1 from equation 2;

576wh/216w = 2.6 hours

The blower dries up wet hands in approximately 45 seconds. Therefore for 2.6 hours (2.6 x 3600 = 9360 seconds), the system will dry the hands of 9360/45 (208) individuals.

Therefore on battery mode, neglecting losses, the system would last for 2.6 hours drying 208 hands approximately. The battery connections are shown in Fig 3.5.

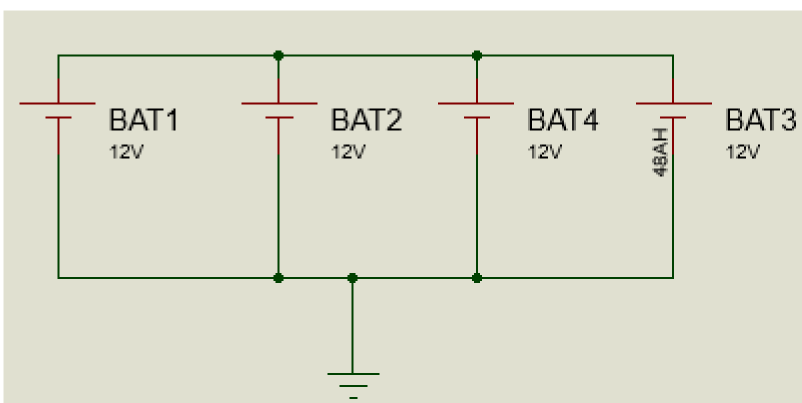


Fig 3.5: Battery connections in parallel

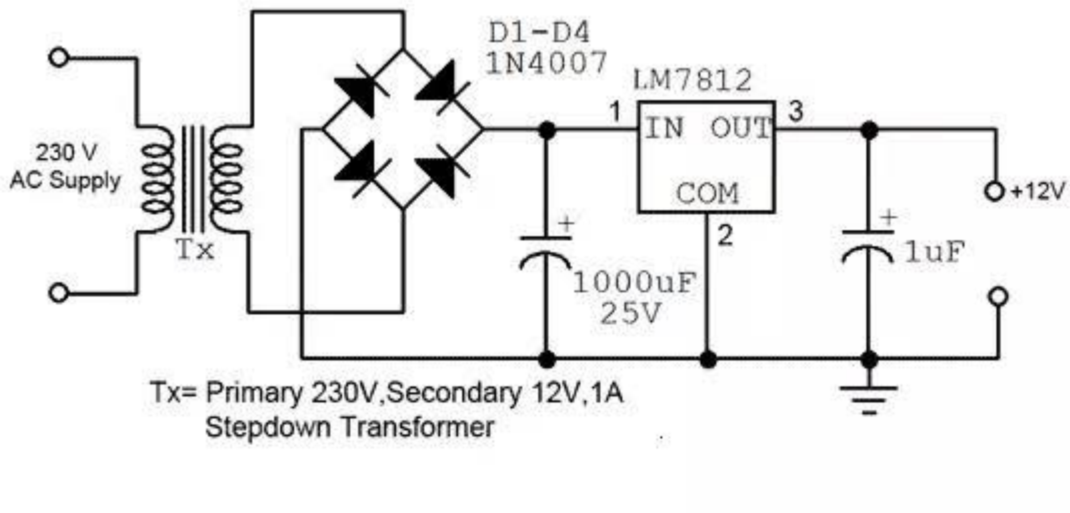


Fig 3.6: Mains to DC voltage circuit diagram

3.2.4 DISPLAY UNIT

The liquid crystal display (LCD) is a 16x2 display signifying 16 rows and two columns. It shows the operation mode at a particular point in time (either idle or working mode) and also shows the battery level. The information to be displayed is fed from the microcontroller to the LCD through the data pins. The connection is as shown in Figure 3.6.

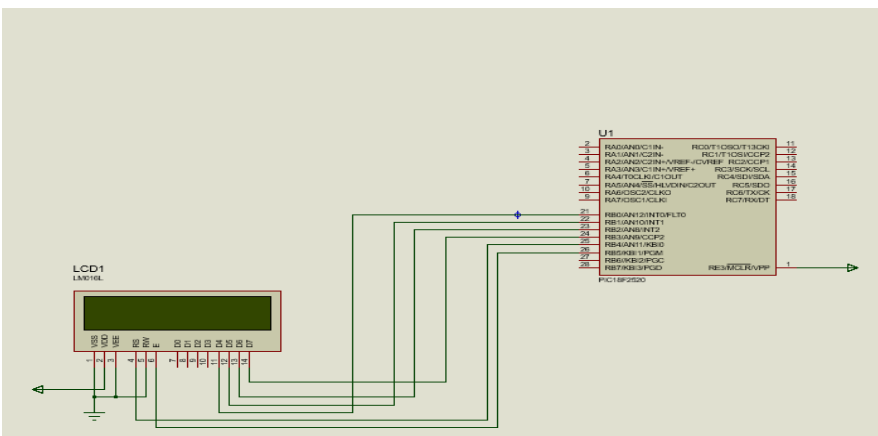


Fig 3.7: Connection of the LCD to the MCU

3.2.5 BLOWER UNIT

The blower unit consists of the air blower. The air blower operates by passing air across a heated element to elevate the temperature of the air which is responsible for drying up wet hands as shown in Figure 3.8. The hot air blower used for this project is a DC powered blower of 12volts, 18Amps. The blower is connected to pin 28 of the microcontroller as shown in Figure 3.9.

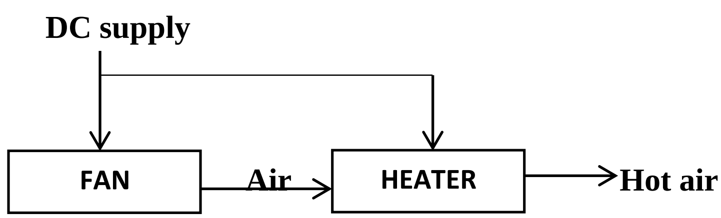


Fig 3.8: Block diagram of the air blower

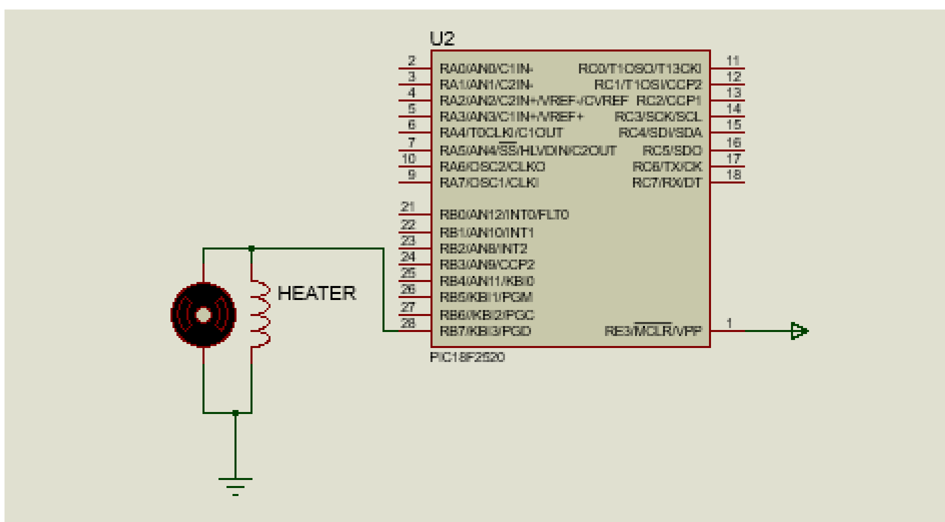


Fig 3.9: Connection of blower to the MCU

After the design of the various units, the units are then interconnected together to form the complete circuit diagram as shown in Figure 3.9.

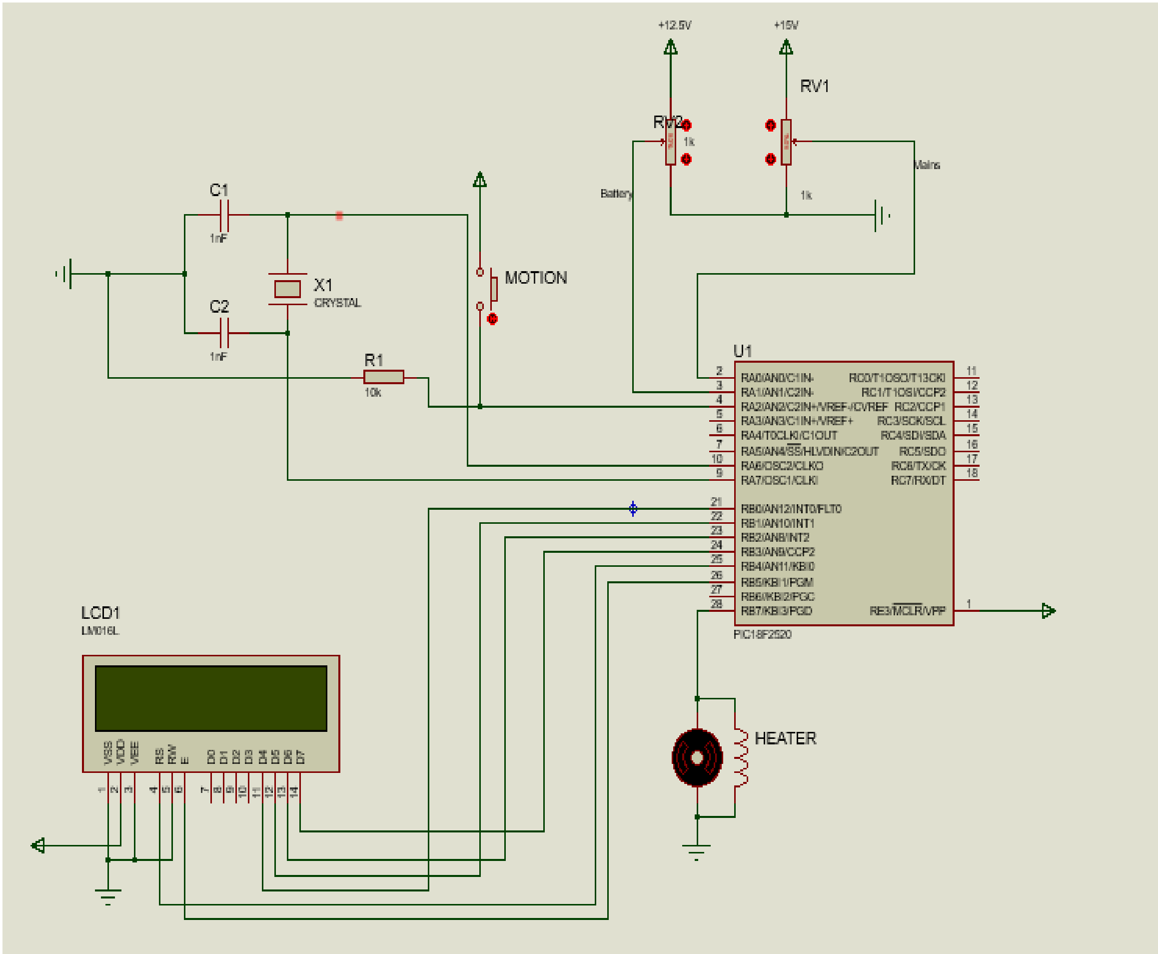


Fig 3.10: Complete circuit diagram

3.3 SOFTWARE DESIGN

Software implementation is the brain behind the system functionality and operation, without a program installed in the hardware device, the intended operation would be impossible to implement. The programs implemented are written in embedded C language using mikro C PRO for PIC environment. An algorithm is written on how the system should operate and a program is written with respect to this algorithm. Basic software applications used in the software design are;

- MikroC PRO for PIC
- Proteus 8 professional

3.4 SYSTEM SOFTWARE FLOW CHART

When the circuit is powered, the microcontroller begins to act according to the code written and uploaded to it. If the proximity sensor senses the hands less than or equal to 10cm away then the logic is true thereby turning on the heater automatically, else if the distance is greater than 10cm the heater remains off. The program source code is given in Appendix A. The flow chat diagram for the embedded software running inside the microcontroller is as shown in figure 3.10.

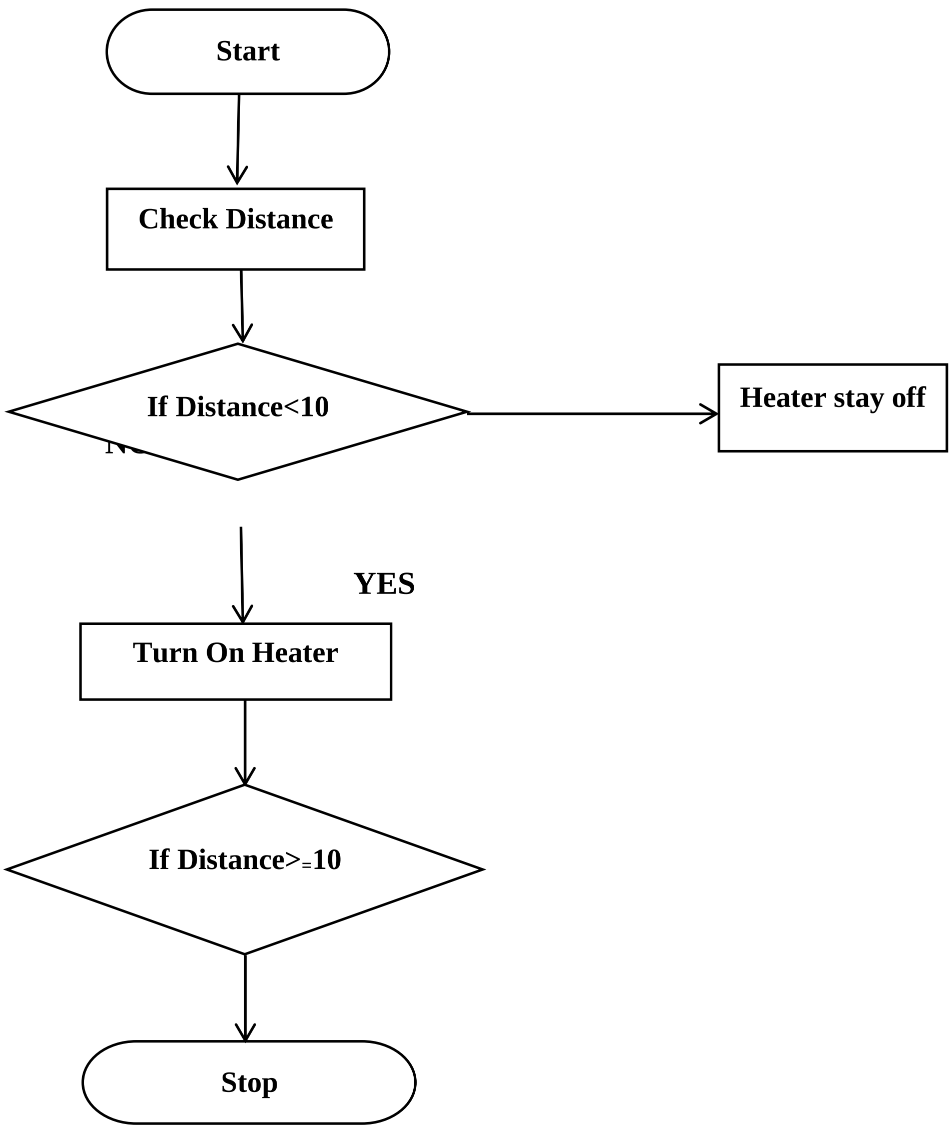


Fig 3.11: System software flow chart

CHAPTER FOUR

IMPLEMENTATION, TESTS AND RESULTS

This chapter seeks to show the implementation, tests and results obtained.

4.1 SOFTWARE IMPLEMENTATION

The software implementation can be grouped into two stages.

- Simulation stage
- Coding stage

4.1.1 SIMULATION STAGE

The design of this project was done in Proteus 8 professional electronic workbench. Proteus was used because of its simplicity in simulation with a vast collection of electronic components, both digital and analogue. The designing of circuits in proteus is simple in that it uses a drag and drop mechanism in designing and also allows the import of written programs from program source files into programmable integrated circuits within its library. The goal of the simulation was to provide an overview of the potential problems that may be encountered during hardware implementation. Figure 4.1 shows the circuit designed using the proteus 8 professional design suite.

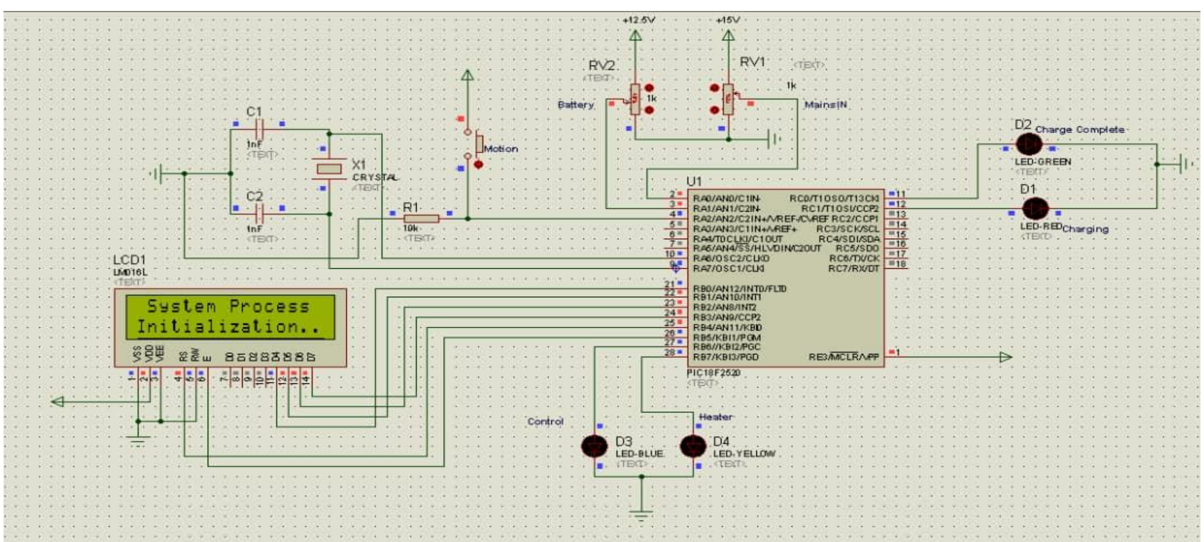
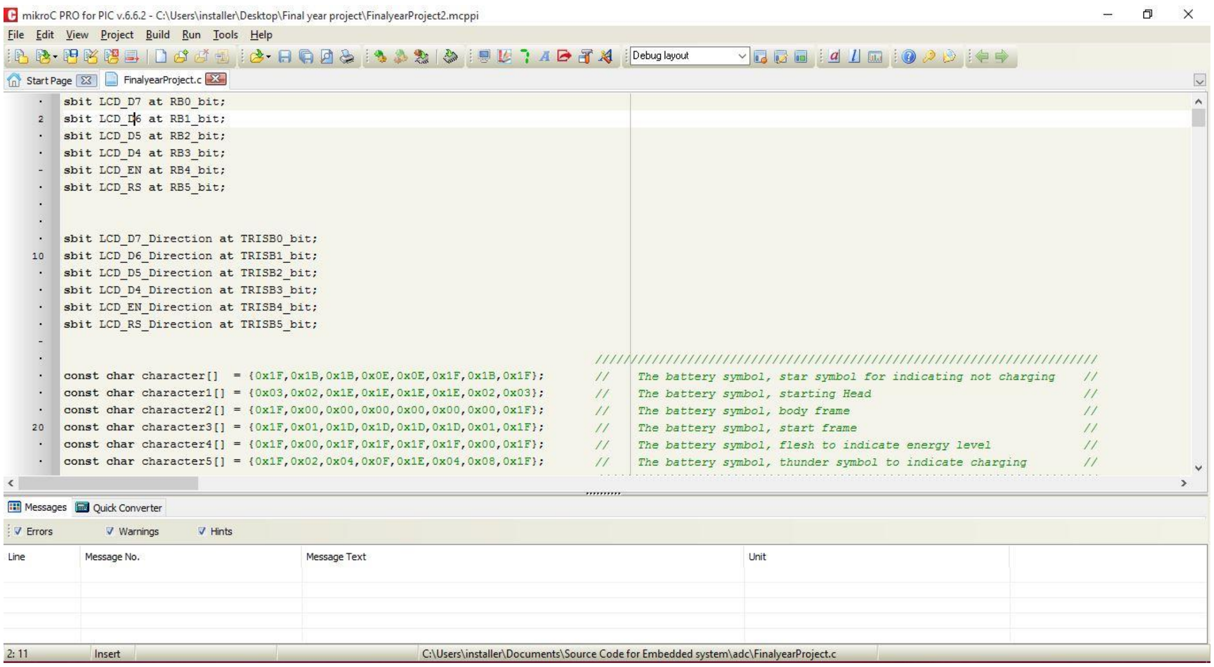


Fig 4.1: Circuit diagram on proteus

4.1.2 CODING STAGE

The codes were written using the MikroC PRO compiler and a programmer was used to upload the written codes to the PIC18F2520 microcontroller chip. Some of the results are shown in fig 4.2.



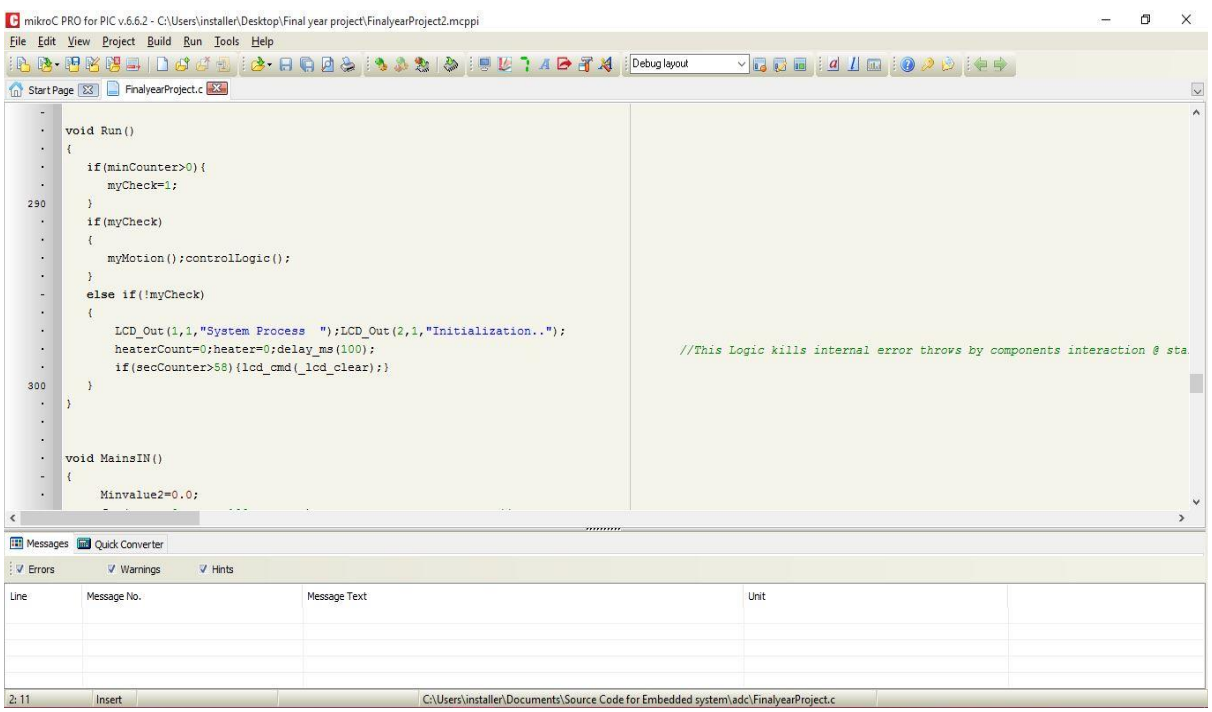


Fig 4.2: MikroC compiler for writing control codes

4.2 HARDWARE IMPLEMENTATION

Implementation of the project design is categorized into various hardware units with each unit composing of various components connected together for the overall efficient performance of the system.

The implementation of this project involved several stages which are;

- Soldering of the various components such as relay, resistors, capacitors etc. properly to the Vero board
- Connection of a 1000 microfarad capacitor in parallel with the air heater blower in order to clear ripples
- Connection of the air heater blower to the soldered circuit on Vero board
- Connection of the charging unit to the soldered circuit on Vero board
- Connection of the batteries in parallel and then to the relevant points on the soldered circuit
- Soldering and insulating of all electrical joints properly

The various components were soldered on Vero board according to the circuit diagram implemented on proteus. The pictorial view of the soldered circuit is shown in Fig 4.1

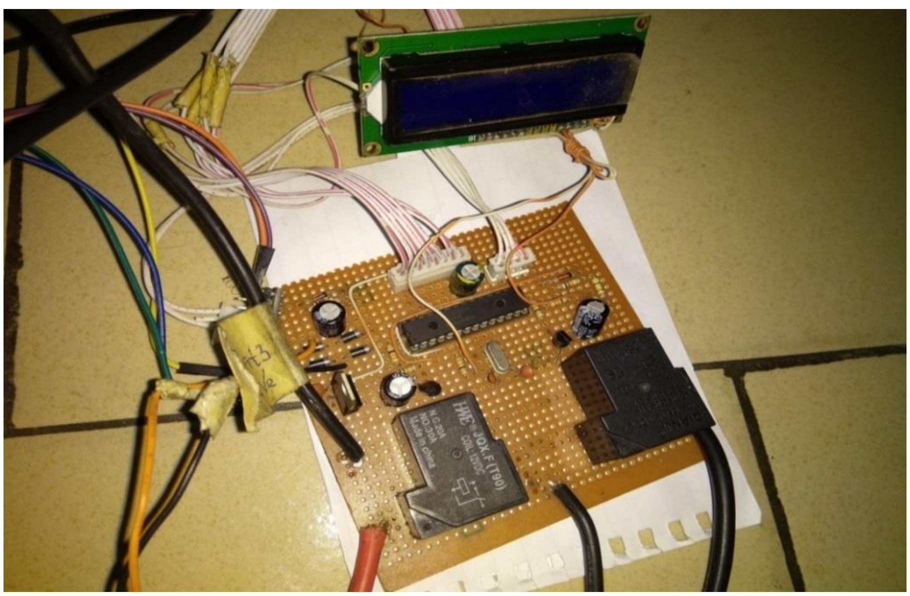


Fig 4.3: Soldered components on Vero board

In order to filter ripples from the DC power supply to the air blower, a 50volts 1000uF capacitor was connected in parallel to the air blower as shown in Fig 4.2

Fig 4.4: Connection of a 1000microfarad capacitor to the air blower

The batteries used in powering the system were connected in parallel in order to increase the output current thereby making the system power to last longer as opposed to if connected in series. A pictorial view of the parallel connections of the batteries is shown in Fig 4.3.

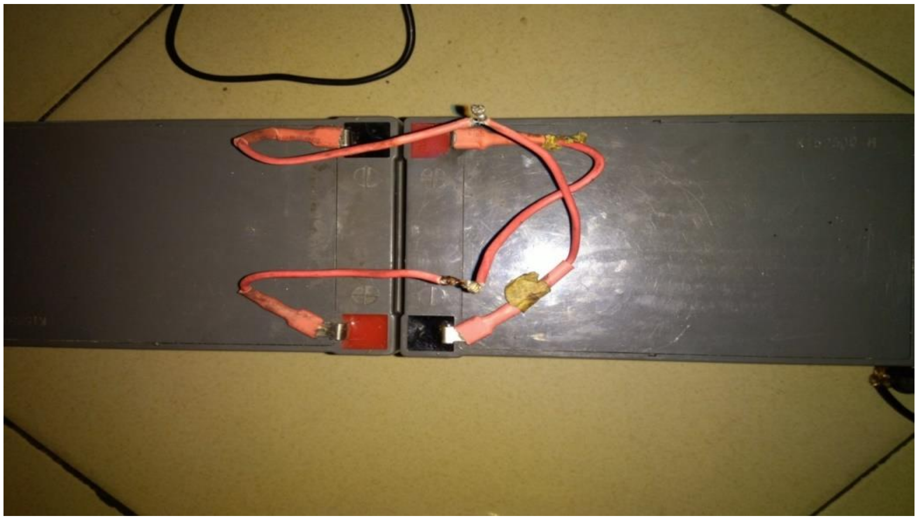


Fig 4.5: Connection of the batteries in parallel

After the various units were established, they were properly connected to each other as shown in Fig 4.6.

\

Fig 4.6: Implementation of all units properly connected

4.3 TESTS

After construction, various tests were conducted to ensure the project met necessary requirements and was built in compliance with the design objectives.

4.3.1 Short-circuit test

This test was carried out using a multimeter set to continuity, for isolation of short circuited part(s) in order to avoid damage to system components when the system is powered.

4.3.2 Open-circuit test

This test was also carried out using a multimeter set to continuity to ensure that all points needed to be properly connected are continuous with each other.

4.3.3 Operational test

This test was conducted to determine how the system reacts to certain inputs. The various operational tests conducted are as follows;

- State of the system when it is powered
- State of the system when it is on battery mode without any input signal to the sensor
- State of the system when it is on battery mode with input signal to the sensor

When the system is powered, the LCD comes ON and displays system process initialization as shown in Fig 4.7.

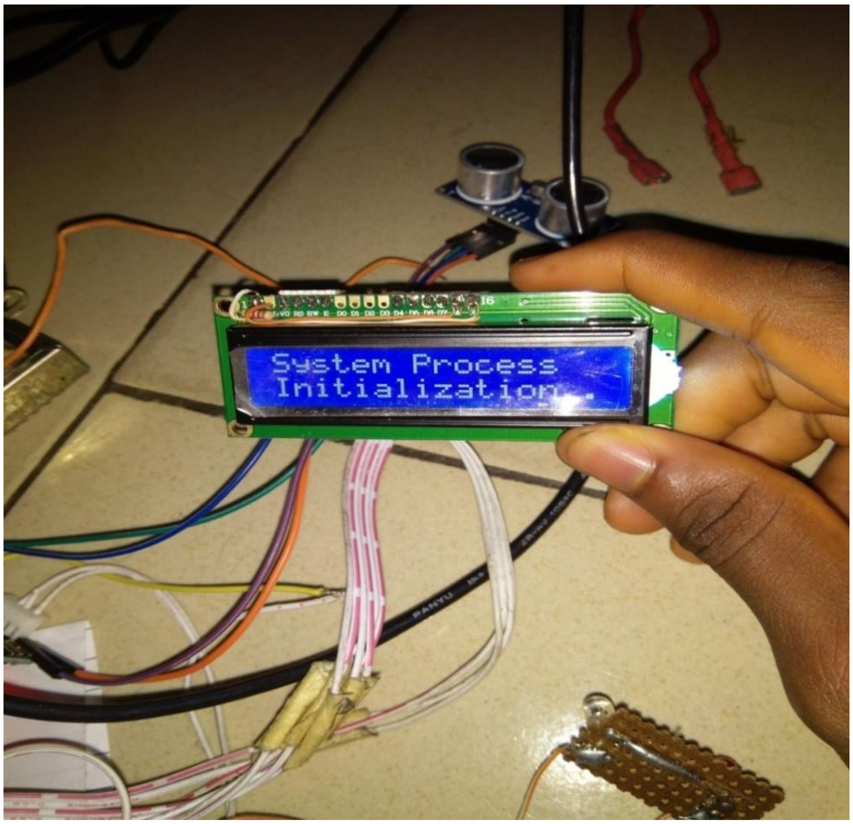


Fig 4.7: First instance when the system is powered

When the system is on battery mode, it shows the battery percentage left and if the blower is idle, it shows idle state as shown in Fig 4.8.

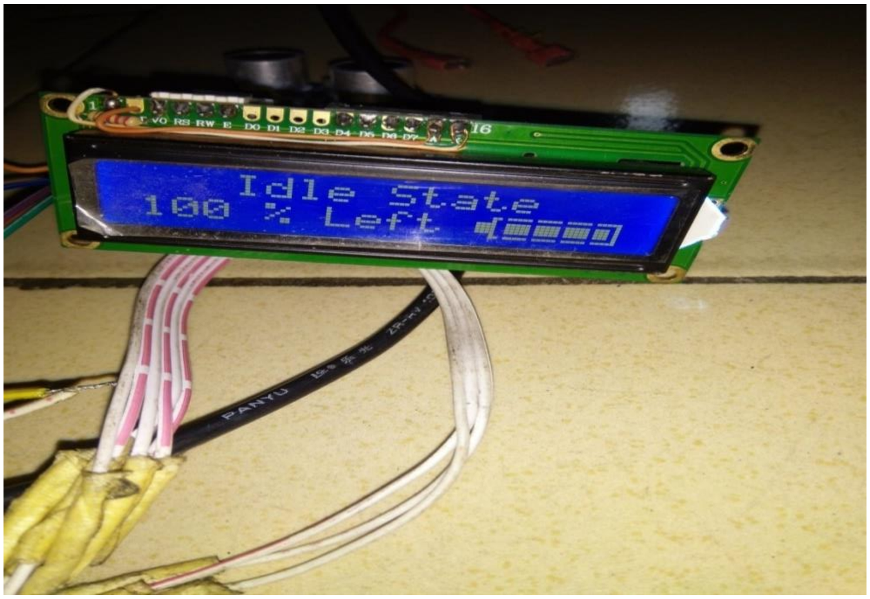


Fig 4.8: When the system is on battery mode and the blower is idle

When the system is on battery mode and the blower is active, it displays active state on the LCD and the battery percentage as shown in Fig 4.9

Fig 4.9: When the system is on battery mode and the blower is active

4.4 PACKAGING

The hand dryer system was neatly packaged in a mobile stand made with iron which houses all components that makes up the system. It was packaged in a mobile format so that it can be used both indoors and outdoors(for events or special occasions) as shown in Fig 4.10.

Fig 4.10: Mobile stand

4.5 PRACTICAL RESULT

The final result of the design and construction of the AC/DC hand dryer system is as shown in fig 4.10. The mode of operation of this system is shown in Appendix B of this report.

Fig 4.11: Complete hand dryer system

4.6 BILL OF ENGINEERING MEASUREMENT AND EVALUATION

Table 4.1 Bill of engineering measurement and evaluation

S/N	**COMPONENT**	**UNIT COST(#)**	**QUANTITY**	**TOTAL COST**
1	BATTERY	7,500	4	30,000
2	CHARGING UNIT	25,000	1	25,000
3	AIR BLOWER	8,000	1	8,000
4	RELAY	400	2	800
5	LCD MODULE	1000	1	1000
6	RESISTORS	10	10	100
7	SOLDERING LEAD	700	1	700
8	JUMPER WIRES	500	-	500
9	CABLES	1000	-	1000
10	ULTRASONIC SENSOR	1000	1	1000
11	PIC18F2520	2500	1	2500
12	TRANSISTORS	20	2	40
13	CAPACITORS	-	7	200
14	IC SOCKET	100	1	100
15	VERO BOARD	200	1	200
16	CRYSTAL OSCILLATOR	50	1	50
17	TRANSFORMER	500	1	500
18	DIODES	10	6	60
19	CASING	15000	1	15000
	MISCELLANEOUS	2000	-	2000
TOTAL				88,750

CHAPTER FIVE

CONCLUSIONS AND RECOMMENDATIONS

5.1 CONCLUSION

Conclusively, the aim of this project which is to design and implement a microcontroller based AC/DC automatic hand dryer system was fully achieved. The operational test shows that the system gives the desired result. The system automatically switches from battery mode to mains mode when mains voltage is detected and vice-versal. This project has therefore helped in promoting good hand hygiene through electrical/electronic means.

5.2 PROBLEMS ENCOUNTERED

During the design and implementation of the system, few problems were encountered.Viz;

- Passive infra-red(PIR) sensor was initially used to detect motion which could not give the desired result. HC-SR04 was therefore used which gave the desired result
- A charging system was designed using transformer and power mosfets but wasn't compatible with the system. A transformerless charger was therefore used which was compatible with the system.
- Some of the components and accessories needed where not readily available in local shops. Therefore orders where made from online markets.

5.3 RECOMMENDATIONS

Although the aim of the project was achieved which has an immense advantage to hand hygiene, the design can be improved in its efficiency thus the following recommendations;

- Higher battery capacity should be used for more durability of the system
- A DC input port can be included in the system for charging the batteries from DC sources such as solar panels.
- To make the blower more effective, the vent should be made wider.
- For easy mobility, a lighter material other than iron should be used in casing the system.

REFERENCES

A.K.Maini. (2007). *Digital Electronics Principles,Devices and Applications.* JohnWiley&Sons Ltd.

Ansari-et-al. (1991). Comparative Study Of the efficiencies of paper,cloth and electric warm air drying.

Canton, J. (2007). *Microcontroller Programming the Microchip PIC.*

Christinab. (2017). *Full text of "Index of patents issued from the United States Patent Office"*. Retrieved JUNE 23, 2019

D.Ibrahim. (2006). *PIC BASIC Projects using PIC BASIC AND PIC BASIC PRO.* Great Britain: MPG Books Ltd.

Gagnon, D. (2011, JUNE 3). *Exceldryer.com.* Retrieved JUNE 23, 2019, from Exceldryer.com: www.exceldryer.com/PDFs/BuildingservicesManagement-BlowingHotAir.pdf

Hymel, S. (2013, AUGUST 12). *what is a battery?* Retrieved JULY 16, 2019, from SPARKFUN: www.sparkfun.com

John. (2018, AUGUST 14). *LIQUID CRYSTAL DISPLAYS*. Retrieved JULY 18, 2019, from CIRCUITS TODAY: www.circuitstoday.com

Kenney, K. (2017, JUNE 13). *Healthfully*. Retrieved JUNE 27, 2019, from Healthfully: www.healthfully.com/158599-air-dryers-vs-paper-towels.html

Kushagra. (2011, NOVEMBER 23). *RELAY SWITCH*. Retrieved JULY 17, 2019, from ENGINEERS GARAGE: www.enginnersgarage.com

Mlekei. (2019, JUNE 10). Retrieved JUNE 23, 2019, from www.wikepedia.org/wiki/special:Histore/Hand_dryer

Mlekei. (2019, JUNE 10). *Wikipedia*. Retrieved JUNE 23, 2019, from Wikipedia: www.wikipedia.org/wiki/special;

Ofualagba, G. (2019). RENEWABLE ENERGY SYSTEMS.

Smith, E. (2015, AUGUST 24). *THE WEIRD HISTORY OF HAND DRYERS WILL BLOW YOU AWAY*. Retrieved JULY 16, 2019, from atlasobscura.com: www.atlasobscura.com

Sonkushre, P. K. (2018, MARCH 26). *homequicks*. Retrieved JUNE 26, 2019, from www.homequicks.com: www.homequicks.com/how-do-hand-dryers-work.amp

Stanmech. (2017, 2 23). Retrieved JULY 17, 2019, from Stanmech Technologies Inc.

T.D.Green. (2008). *Embedded systems programming.*

Tokyo. (2017, NOVEMBER 2). Retrieved JUNE 23, 2019, from www.mistibushielectric.co.jp/ldg/ja/air/products/jettowel/index_sp.html

Verle, M. (2010). *PIC Microcontrollers, Mikroelektronical 1st edition.*

APPENDIX A

PROGRAM SOURCE CODE

```
sbit LCD_D7 at RB0_bit;

sbit LCD_D6 at RB1_bit;

sbit LCD_D5 at RB2_bit;

sbit LCD_D4 at RB3_bit;

sbit LCD_EN at RB4_bit;

sbit LCD_RS at RB5_bit;

sbit LCD_D7_Direction at TRISB0_bit;

sbit LCD_D6_Direction at TRISB1_bit;

sbit LCD_D5_Direction at TRISB2_bit;

sbit LCD_D4_Direction at TRISB3_bit;

sbit LCD_EN_Direction at TRISB4_bit;

sbit LCD_RS_Direction at TRISB5_bit;


//////////////////////////////////////////////////////////////////////

const char character[]  = {0x1F,0x1B,0x1B,0x0E,0x0E,0x1F,0x1B,0x1F};       //    The battery symbol, star symbol for indicating not charging    //

const char character1[] = {0x03,0x02,0x1E,0x1E,0x1E,0x1E,0x02,0x03};       //    The battery symbol, starting Head                              //

const char character2[] = {0x1F,0x00,0x00,0x00,0x00,0x00,0x00,0x1F};       //    The battery symbol, body frame                                 //

const char character3[] = {0x1F,0x01,0x1D,0x1D,0x1D,0x1D,0x01,0x1F};       //    The battery symbol, start frame                                //

const char character4[] = {0x1F,0x00,0x1F,0x1F,0x1F,0x1F,0x00,0x1F};       //    The battery symbol, flesh to indicate energy level             //

const char character5[] = {0x1F,0x02,0x04,0x0F,0x1E,0x04,0x08,0x1F};       //    The battery symbol, thunder symbol to indicate charging        //


//////////////////////////////////////////////////////////////////////

char ba3PercentDisplay[6];
```

```
char timer[3];
//char voltageDisplay[6];

void barBad() {
    char i;
    Lcd_Cmd(64);
    for (i = 0; i<=7; i++) Lcd_Chr_CP(character[i]);
    Lcd_Cmd(_LCD_RETURN_HOME);
}

void barHead() {
    char i;
    Lcd_Cmd(72);
    for (i = 0; i<=7; i++) Lcd_Chr_CP(character1[i]);
    Lcd_Cmd(_LCD_RETURN_HOME);
}

void barFrame() {
    char i;
    Lcd_Cmd(80);
    for (i = 0; i<=7; i++) Lcd_Chr_CP(character2[i]);
    Lcd_Cmd(_LCD_RETURN_HOME);
}

void barStart() {
    char i;
    Lcd_Cmd(88);
    for (i = 0; i<=7; i++) Lcd_Chr_CP(character3[i]);
```

```
    Lcd_Cmd(_LCD_RETURN_HOME);
}
void barBody() {
    char i;
    Lcd_Cmd(96);
    for (i = 0; i<=7; i++) Lcd_Chr_CP(character4[i]);
    Lcd_Cmd(_LCD_RETURN_HOME);
}
void barStar() {
    char i;
    Lcd_Cmd(104);
    for (i = 0; i<=7; i++) Lcd_Chr_CP(character5[i]);
    Lcd_Cmd(_LCD_RETURN_HOME);
}
void myCharge()
{
     lcd_chr(1,16,3);      ////////////////////////////////////////
     lcd_chr(1,15,2);      //                              //
     lcd_chr(1,14,2);      //  Setting Up the structure          //
     lcd_chr(1,13,2);      //                              //
     lcd_chr(1,12,1);      ////////////////////////////////////////
     delay_ms(500);        //delay for 0.5 second
     lcd_chr(1,14,5);      //print the body on postion 15
     delay_ms(500);        //delay for 0.5 second
}
void liteCharge()
{
```

```
    lcd_chr(1,16,3);      ///////////////////////////////////////
    lcd_chr(1,15,4);      //                                   //
    lcd_chr(1,14,2);      //  Setting Up the structure         //
    lcd_chr(1,13,4);      //                                   //
    lcd_chr(1,12,1);      ///////////////////////////////////////
    delay_ms(500);         //delay for 0.5 second
    lcd_chr(1,14,5);      //print the body on postion 15
    delay_ms(500);         //delay for 0.5 second
}

void oneBar()
{
    lcd_chr(2,16,3);      ///////////////////////////////////////
    lcd_chr(2,15,2);      //                                   //
    lcd_chr(2,14,2);      //  Setting Up the structure         //
    lcd_chr(2,13,2);      //                                   //
    lcd_chr(2,12,1);      ///////////////////////////////////////
    delay_ms(500);
}
void twoBar()
{
    lcd_chr(2,16,3);      ///////////////////////////////////////
    lcd_chr(2,15,4);      //                                   //
    lcd_chr(2,14,2);      //  Setting Up the structure         //
    lcd_chr(2,13,2);      //                                   //
    lcd_chr(2,12,1);      ///////////////////////////////////////
    delay_ms(500);
}
```

```
void threeBar()
{
    lcd_chr(2,16,3);      ////////////////////////////////////////
    lcd_chr(2,15,4);      //                                    //
    lcd_chr(2,14,4);      //  Setting Up the structure          //
    lcd_chr(2,13,2);      //                                    //
    lcd_chr(2,12,1);      ////////////////////////////////////////
    delay_ms(500);
}
void fullBar()
{
    lcd_chr(2,16,3);      ////////////////////////////////////////
    lcd_chr(2,15,4);      //                                    //
    lcd_chr(2,14,4);      //  Setting Up the structure          //
    lcd_chr(2,13,4);      //                                    //
    lcd_chr(2,12,1);      ////////////////////////////////////////
    delay_ms(500);
}
void fullycharged()
{
    lcd_chr(1,16,3);      ////////////////////////////////////////
    lcd_chr(1,15,4);      //                                    //
    lcd_chr(1,14,4);      //  Setting Up the structure          //
    lcd_chr(1,13,4);      //                                    //
    lcd_chr(1,12,1);      ////////////////////////////////////////
    delay_ms(500);
}
```

```
//////////////////////INPUT////////////////
sbit Motion       at Ra2_bit;
/////////////////////OUTPUT/////////////////
sbit greenLED     at ra3_bit;
sbit control      at rb6_bit;
sbit redLED       at rc3_bit;
sbit heater       at rc6_bit;
///////Variable declaration//////////////////
unsigned int msCounter=0;
unsigned int secCounter=0;
unsigned int minCounter=0;
unsigned int hrCounter=0;
unsigned int secHeater=0;
unsigned int millis=0;
unsigned int Period=0;
unsigned int Reload = 103;
unsigned PeriodFlag=0;
unsigned int myCheck=0;
long unsigned int lowIn;
long unsigned int pause = 5000;
int lockLow            = 1;
int takeLowTime        = 0;
int chargeComplete     = 0;
int cnt[4]             = {0,0,0,0};
int PIRValue           = 0;
int count;
int heaterCount        = 0;
int heaterControl      = 0;
```

```
int x                   = 0;
float MaxValue,MinValue,Minvalue2,Maxvalue2,v;
float Ba3Read,Ba3Cal,Ba3Cal2;float ba3Norm=12.5;
float percentScale=0.0;float Ba3Percent=0.0;
void sysInit();
void myMotion();
void TimeCounters();
void Run();
void controlLogic();
void batteryDisplay(float ba3info, float ba3status);
void init_timers()
{
   T0CON.T0PS2      = 1;               //////////////////////////////////
   T0CON.T0PS1      = 0;               //   Set Prescaler value 1:32   //
   T0CON.T0PS2      = 0;               //////////////////////////////////
   TMR0l           = 103;             //98.98;           // load low bit  Register with decimal
98.979,98.9797,98.979999(slow) 98.98(fast)
   TMR0H           = 0xF0;            // load high bit Register with decimal 240
   TMR0IE_bit      = 1;               // Enable Timer0 interrupt
   GIE_bit         = 1;               // Enable global interrupts
   PEIE_bit        = 1;               // Enable Peripheral interrupts
   PSA_bit         = 0;               // Time Clock Source is from prescaler
   T0CS_bit        = 0;               // Prescaler get clock from cpu
   T08BIT_bit      = 0;               // 16 Bit mode
   TMR0ON_bit      = 1;               // Now start the timer
}

void interrupt()
{
```

```
  if(T0IF)
  {
    INTCON   = 0x20;               // Bit TOIE is set, bit TOIF is cleared
    TMR0l    = 103;                // 98.98;              // load Register with decimal 98.98
    TMR0H    = 0xF0;               // TMR0l    = 0X64;     // load Register with decimal 100
    msCounter++;                   // increment track
    millis++;
  }
  TimeCounters();
}
void TimeCounters()
{
  if (msCounter==250)
  {
     secCounter++;secHeater++;
     msCounter=0;
  }
  if(secCounter==60)
  {
     minCounter++;
     secCounter=0;
  }
  if(minCounter==60)
  {
     hrCounter++;
     minCounter=0;
  }
  if(hrCounter>12)
```

```
  {
    hrCounter = 1;
  }
}
void main()
{
   init_timers();
   PORTA = 0;PORTB = 0;PORTC = 0;
   trisa.f0=1;trisa.f1=1;trisa.f2=1;trisa.f3=0;trisb.f6=0;trisc.f6=0;trisc.f3=0;
//Input pin declaration
   porta.f0=0;porta.f1=0;porta.f2=0;porta.f3=0;portb.f6=0;portc.f6=0;portc.f3=0;
//input pin initialization
   CMCON |= 0x07;
   ADCON1 = 0x0D;
   ADC_Init();
   Lcd_Init();
   barStart();
   barHead();
   barFrame();
   barBody();
   barStar();
   liteCharge();
   barBad();
   Period=1;Periodflag=0;
   LCD_Cmd(_LCD_CURSOR_OFF);
   LCD_Cmd(_LCD_CLEAR);

   for(;;)
   {
```

```
      sysInit();
    }
}
void sysInit()
{
    if(cnt[1]<1)
    {
      heater=0;heaterCount=0;
      LCD_Out(1,1,"   HAND DRYER ");");
delay_ms(3000);LCD_Cmd(_LCD_CLEAR);
      LCD_Out(1,3,"DESIGNED BY");LCD_Out(2,2,"EGBORO
GABRIEL");delay_ms(3000);LCD_Cmd(_LCD_CLEAR);
      LCD_Out(1,1,"   FUPRE, 2019  ");LCD_Out(2,1,"   DELTA STATE
");delay_ms(2000);
      LCD_Cmd(_LCD_CLEAR);cnt[1]++;
    }
    Run();
}

void Run()
{
  if(minCounter>0){
    myCheck=1;
  }
  if(myCheck)
  {
    myMotion();controlLogic();
  }
  else if(!myCheck)
```

```
  {
      LCD_Out(1,1,"System Process  ");LCD_Out(2,1,"Initialization..");
      heaterCount=0;heater=0;delay_ms(100);                              //This Logic kills
internal error throws by components interaction @ start
      if(secCounter>58){lcd_cmd(_lcd_clear);}
  }
}
void MainsIN()
{
    Minvalue2=0.0;
    for(count=0;count<100;count++)                         //cycle through 100 counts
    {
      v = ADC_Read(0);
      v = (v*4.89*11.0*15.45526)/1000.0;                  //  15.0  --> 15.45526  -->  14.1896
//Vx4.89x11.0x20.89322 -->for 9v Transformer
      Maxvalue2 = v+Minvalue2;
      Minvalue2=Maxvalue2;
    }
    Minvalue2=Minvalue2/100.0;
    v=Minvalue2;
}
void battery(){
  Ba3Cal =0.0;
  for(count=0;count<=500;count++){                                        //sample for 1sec
      Ba3Read = ADC_Read(1);                                          // analog read
      Ba3Read = (Ba3Read*4.89*49.3164)/1000.0;                           //49.2564  -->
49.3064 --> 52.7272 -->                    // to read max of 250Vdc, which might occur
during charging. However, actual max voltage 12.5*15 = 187.5V. 10kva system
      Ba3Cal2 = Ba3Read+Ba3Cal;
      Ba3Cal=Ba3Cal2;
```

```
  }
  Ba3Cal=Ba3Cal/500.0;
  Ba3Read=Ba3Cal;
}

void myMotion()
{
   if(Motion)
   {
       if(lockLow)
       {
          lockLow = 0;
          //delay_ms(1500);
       }
       takeLowTime = 1;
   }
   if(Motion)
   {
      heater = 1;
   }
   if(!Motion)
   {
      if(!heater){lcd_out(1,1, "   Idle State   ");}
      if(takeLowTime){millis=0;lowIn = millis;takeLowTime = 0;}
      if(!lockLow && millis - lowIn > pause)
      {
         lockLow = 1;Motion=0;
         delay_ms(1000);
```

```
      }
    }
    /**
    * This logic Resets the alert system to its default state after 2 minutes of ON state
    */
    if(heater)
    {
      lcd_out(1,1, "  Active State  ");
      if(cnt[0]<1)
      {
        if(secHeater>100){secHeater=100;}
        heaterCount=secHeater;cnt[0]=1;
      }
      if(secHeater-heaterCount>=30)                //If after the integer value set
at this point point is exceeded, the system reset the actuators.
      {
        cnt[0]=0;heater=0;secHeater=0;
      }
    }
  }
  void controlLogic()
{
    MainsIN();
    if(v>100.0){
      lcd_out(2,1,"   Mains Mode   ");
      battery(); x=0;
      }
    }else if(v<100.0){
      if(!x){x++;lcd_cmd(_lcd_clear);}
```

```
    battery();batteryDisplay(ba3Norm, Ba3Read); }
}
void batteryDisplay(float normVoltage, float ba3status)                    //normVoltage carries the set battery normal level, while ba3status carries the on-time read-out
{
  Ba3Percent = (ba3status*100.0)/normVoltage;
  percentScale = ((Ba3Percent-100.0)+12.0)/0.12;
  if(percentScale<=0.9){percentScale=0.0;}
  floattostr(percentScale,ba3PercentDisplay);
   if(percentScale>=100.0){
    lcd_out(2,1,"100 % Left ");fullbar();
  }else if(percentScale<100.0 && percentScale>=75.0){
    Lcd_out(2,1,(ltrim(ba3PercentDisplay)));
    Lcd_out(2,3,"% Left   ");fullBar();
  }else if(percentScale<75.0 && percentScale>=50.0){
    Lcd_out(2,1,(ltrim(ba3PercentDisplay)));
    Lcd_out(2,3,"% Left   ");threeBar();
  }else if(percentScale<50.0 && percentScale>=25.0){
    Lcd_out(2,1,(ltrim(ba3PercentDisplay)));
    Lcd_out(2,3,"% Left   ");twoBar();
  }else if(percentScale<25.0 && percentScale>=1.0){
    Lcd_out(2,1,(ltrim(ba3PercentDisplay)));
    Lcd_out(2,3,"% Left   ");oneBar();
  }else{
    lcd_out(2,1,"0 % Left   ");oneBar()
  }
}
```

APPENDIX B

MODE OF OPERATION

The operation of the entire system is divided into different parts; the control unit, the blower and the power supply unit and how they work with each other to achieve the goal of automation.

The working principle of the hand dryer is that the sensor detects the signal (hands); the microcontroller therefore receives a high (5v) from the sensor and outputs a high (5v) to the coils of the relay connected to the blower thereby energizing it. When the coils are energized the common of the relay which is connected to the blower switches from normally closed to normally open which then powers the blower. When the signal (hands) detected by the sensor is removed, the coils of the blower relay are de-energized and the blower goes off automatically.

The power unit is also controlled with a separate relay that switches the system from battery to mains when mains voltage is being detected and vice-versa. The wiring diagram of the system is shown below.

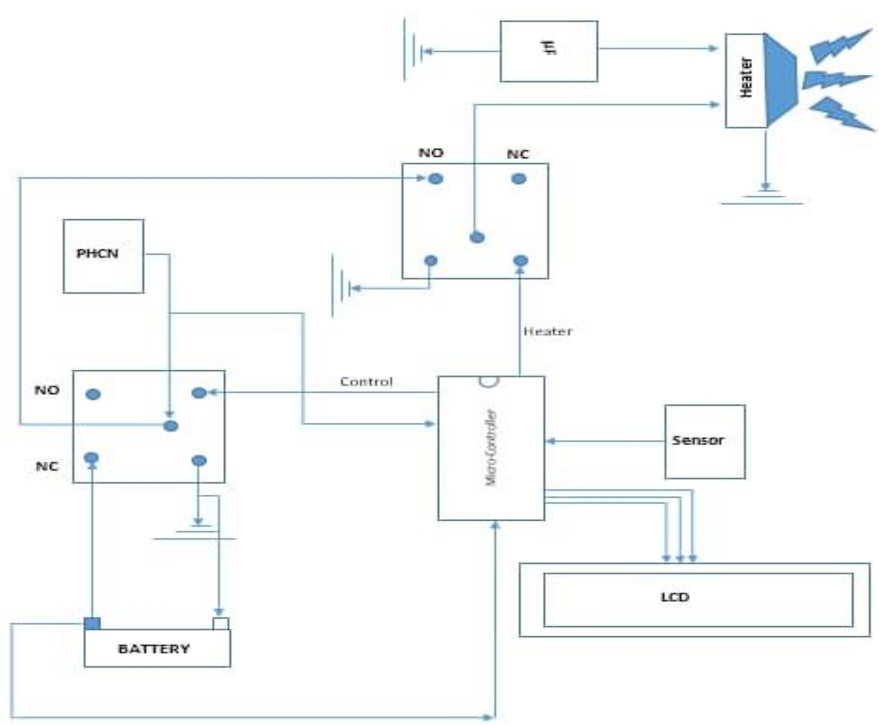


Fig A. Wiring diagram of the hand dryer system