Journal of Industrial Technology, 2017, 2 (4): 78 - 89
Journal Homepage: www.erjournals.com
ISSN Online: 0184-7937

Development of a Microcontroller Based Temperature and Humidity Controller for Infant Incubator

Oyebola, B. O.[1]; Anih M. E[2]. & Odueso, V. T. [3].
[1,2]Department of Computer Engineering Technology, Gateway (ICT) Polytechnic Saapade, Nigeria.
[3]Department of Electrical/Electronic Engineering, University of Ibadan, Nigeria.
Corresponding email: E-mail: blessedolalekan@gmail.com

ABSTRACT
Infants born prematurely are generally kept in special chambers referred to as "incubators", which are enclosures with controlled temperature and humidity. For some special treatment some new-born babies are placed in the incubators. Since the introduction of incubators and its advancement in technology, the survival rate of premature births and new-borns has increased significantly. Advanced incubators are very expensive making it difficult for hospitals, especially small ones, to procure them especially with economic crunch and resource constraints for capital equipment. This paper provides the details for the design and development of a microcontroller based temperature and humidity sensor for an infant incubator.

Keywords: Development; Incubator; Infant; Microcontroller.

Citation: Oyebola B. O. and Odueso V. T. (2017), Design and Development of Microcontroller Based Temperature and Humidity Controller for Infant Incubator. *Journal of Industrial Technology*, 2 (4): 78 –87

1.0 Introduction

Infant incubator system is a vital and critical area because it deals with premature infant or illness baby. It is essential to detect any abnormal conditions that can occur in the incubator as soon as possible (Changela, Daxini and Parmar, 2016). Temperature, Humidity and Oxygen concentration are the main parameters which must be controlled in the incubator. This paper deals with the measurement of temperature and humidity inside the incubator as they are one among the most-measured process variable in the human body. Most commonly, a temperature sensor and humidity sensor is used to convert temperature and humidity value to electrical value (Changela, Daxini and Parmar, 2016).

It is quite evident that for a proper growth of a pre-mature baby the environment for the development of the baby should be in an environment that is same that of the mother's womb. In such environment the baby is tends to develop into a matured baby. This work scope of study is limited to the design and development of a microcontroller based temperature and humidity controller for an infant incubator, which is designed to control the temperature and humidity within an incubator to give the baby a proper development after birth. The design gives the operator the ability to adjust the temperature and humidity to the normal state just like the womb. In hospitals baby incubator is one of essential life supportive equipment for the premature babies.

Unfortunately, there is a lack of low cost infant incubators in the developing world. The persistence of this paper is to design and develop a microcontroller based temperature and humidity control in an infant incubator. Advances in electronic techniques couple with economical prices make humidity and temperature control cost-effective with highly accurate and stable performance provide the required environment for the growth of the premature baby.

2.0 The Review

In the early years, several previous technology implemented before the introduction of temperature and humidity controller in an infant incubator. Before the industrial revolution, premature and ill infants were born and cared for at home and either lived or died without medical intervention. "In the mid-nineteenth century, the infant incubator was first developed, based on the incubators used for chicken eggs by Dr. Stephane Tarnier who is generally considered to be the father of the incubator, having developed it to attempt keeping premature infants warm. The first Tarnier's incubator housed several infants who were warmed over a hot-water reservoir attached to an external heating source" (Jefferey, 2000). Figure 2.1 explains the construction of Dr. Stephen which its apparatus is simplified into a single infant model heated by hot-water bottles which are replaced manually every 3 hours (Jefferey, 2000).

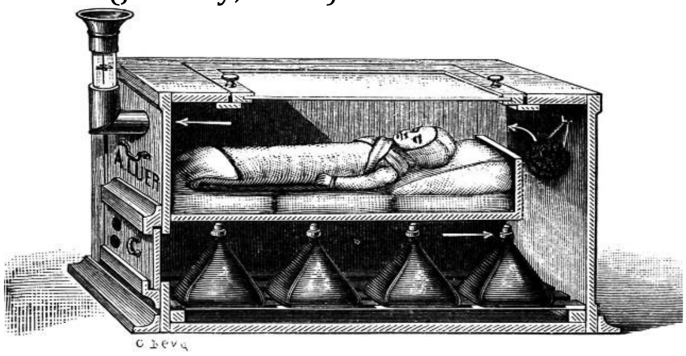

Figure 2.1: Tarnier's Incubators *(Jefferey, 2000)*

Alexandra Lion, a physician and a son of an inventor, developed in 1890s a much more sophisticated incubator than that of Tarnier (Jefferey, 2000). It is a large metal apparatus equipped with a thermostat and an independent forced ventilation system, the Lion incubator was designed to compensate for less than-optimal nursing environment. Unfortunately, none of its features came cheaply, limiting its appeals to charity or government supported hospitals (Jefferey, 2000). Julius Hess in 1914 developed his own version of incubator; it was an electrical heated bed reminiscent of crede's design that surrounded the infant in a metal jacket containing hot water. Hess while many of his contemporaries had rejected incubators, he realised that they were in fact useful but had to incorporate into a supportive context. He expanded the function of the incubator itself into an oxygen chamber, and developed an automobile-based transport system to address the problem of treating out born infants (Jefferey, 2000).

By the year 1970s when Neonatal Intensive Care Units (NICUs)was established, in part of hospitals in the developed world. In Britain, some early units ran community programmes, sending experienced nurses to help care for premature babies at home. But increasingly technological monitoring and therapy meant special care for babies became hospital-based (Mohammed, 2011). Not only careful nursing, but also new techniques and instruments now played a major role. As in adult intensive care units, the use of monitoring and life support systems became routine. These needed special modifications for small babies, whose bodies were tiny and often immature; the Figure 2.2 explains its structure. Adult ventilators, for example, could damage babies' lungs and gentler techniques with smaller pressure changes were devised. The many tubes and sensors used for monitoring the baby's condition, blood sampling and artificial feeding made some babies scarcely visible beneath the technology (Mohammed, 2011).

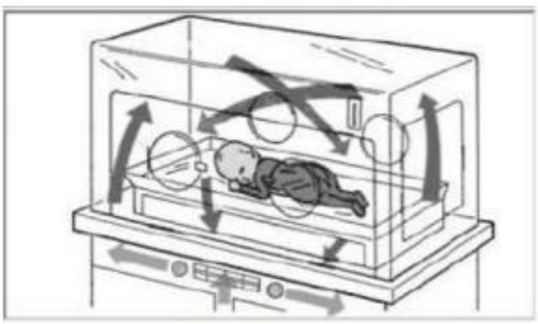

Figure 2.1: Infant *Incubator (Ahmed, 2011)*

3.0 The Technology Used

The approach used in the design and development of a microcontroller based temperature and humidity controller were as illustrated in the following sections. The design and development of a microcontroller based temperature and humidity controller uniqueness is found in its ability to monitor and sensor the change in the temperature and humidity within an incubator, the Figure 3.1 explains procedural steps on how the system was design.

3.1 Hardware Details of the Project

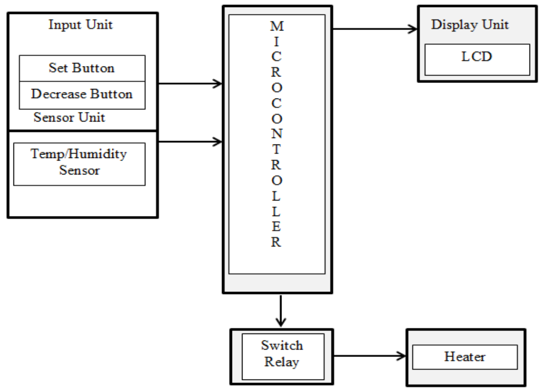

Figure 3.1: Block diagram of micro-controller based temperature and humidity controller.

The input unit consist of Set Button, Increase Button and Decrease Button. The input unit is connected directly to the microcontroller, and they are designed using push buttons, the increase button or decrease button is used to increase or decrease the value of the temperature/humidity should in case the temperature/humidity is below or the above required, whereas the set button is a functional button that when pressed sends the pre-set temperature/humidity to the micro-controller, as described in figure 3.2.

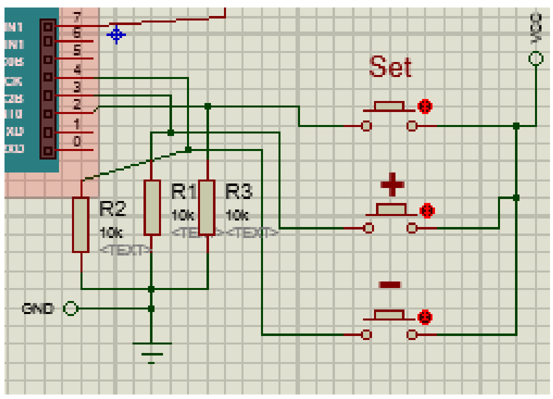

Figure 3.2: Input Unit

3.1.1 Sensor Unit

The choice of transducer or sensor to measure the temperature of baby in a microcontroller based infant incubator, the temperature of water reservoir and humidity of baby chamber is very critical, the sensor required must be able to sense the temperature and the humidity within the incubator. Figure 3.3 is an LM356 which is used to sense the temperature and humidity.

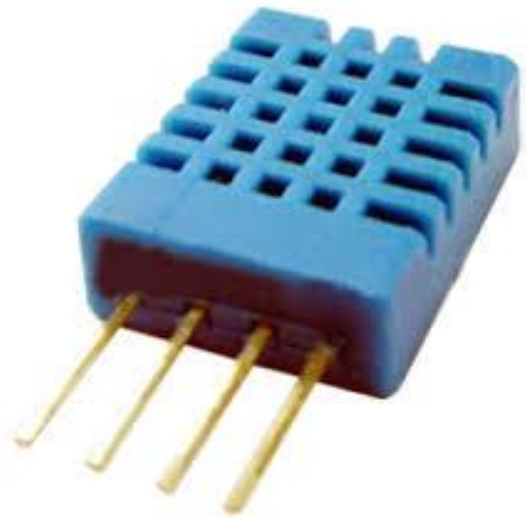

Figure 3.3: Sensor *(Electronic Tutorials, 2013)*

3.1.2 Switch Relay

The switch relay is used to switch OFF the heat controller socket when the temperature within the incubator is up to expected value while ON when the temperature is lower than expected value

3.1.3 Control unit

It is the central part of system design; it is designed in such a way that it controls all other unit in the design. It controls all the units that are connected to it so as to give out a desired output. For the sake of this project I made use of ATMEGA 328 microcontroller. Figure 3.4 explains the each pins of the microprocessor which lead use to using it based on the features it provides.

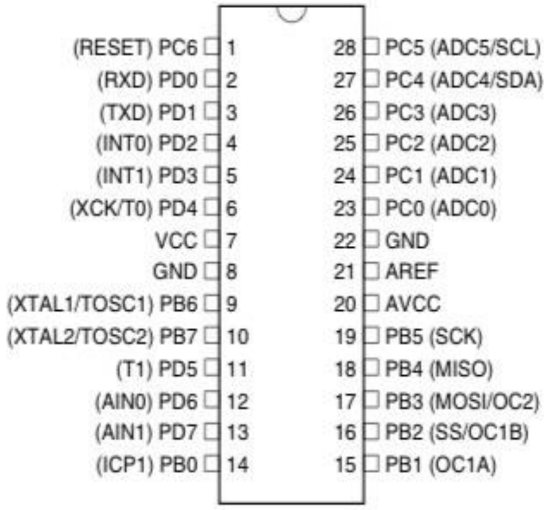

Figure 3.4: ATMEGA328 (Control Unit) *(Atmel, 2009)*

3.1.4 Display Unit

The display unit for the design of this project is a Liquid Crystal Display. As shown in figure 3.5 the figure shows the functions of each pin, the display is connected directly to the microcontroller.

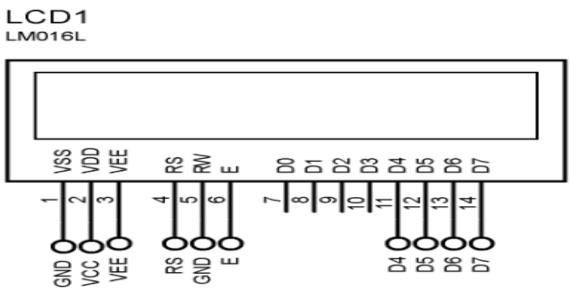

Figure 3.5: Liquid Crystal Display (Vishay, 2002)

3.1.5 Heater Unit

The heater is used to ensure that the temperature within the incubator is up to expected value, it heats up the environment once the humidity is lower than expected. For the sake of this project created a place where the heater can be connected during the implementation.

3.2 Software Development

MikroC compiler is used to edit and compile C code for PIC16F877A. Only hex, file is downloaded into programmer in the microcontroller chip. Proteus simulator is used to design and simulate electronic circuits, also help to testing the code. The developed control codes were as in Appendix A.

4.0 Result, Testing and Discussion

The device is designed in such a way that it works with normal power source supply rather than battery, reason been that in a situation where the battery is low it will affect the reading of the temperature. Figure 4.1 (the circuit diagram) shows the different units that made up the construction and where each component is connected, to ensure each perform expected output.

Figure 4.2 explains the steps followed by the program from the initially stage of the program and also shows the possible conditions.

Figure 4.3 shows the completion of the designed project and figure 4.4 shows the internal circuit of the construction.

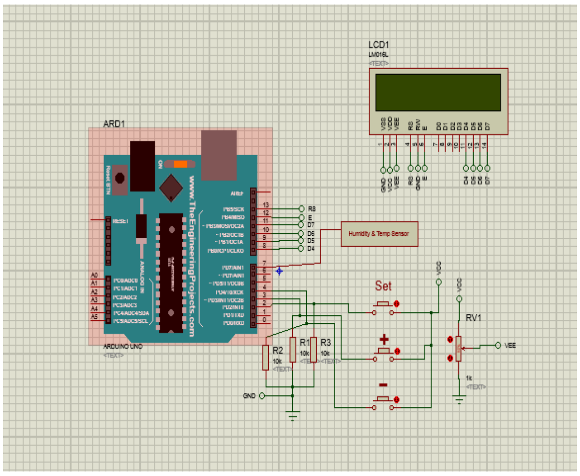

Figure 4.1: The circuit diagram

The Atmega328 microcontroller combines 32KB ISP flash memory with read-while write capabilities, 1KB EEPROM, 2KB general purpose I/O lines, 32 general purpose working registers, three flexible timer/counter with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, 6-channel 10-bit A/D converter (8-channel in TQFP and QFN/MLF packages), programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The device operates between 1.8-5.5 volts. The device achieves throughput approaching 1 MIPS per MHz (Atmega, 2017).

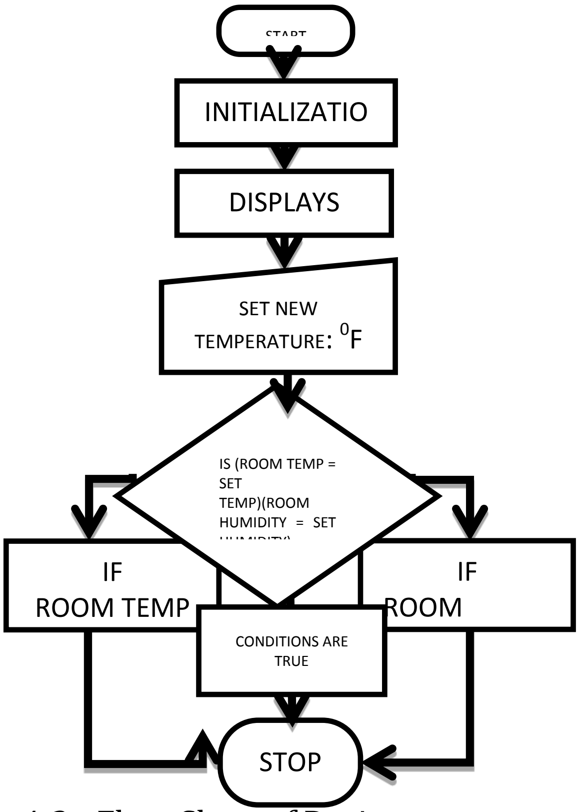

Figure 4:2 - Flow Chart of Design

Figure 4 3: Device after Construction

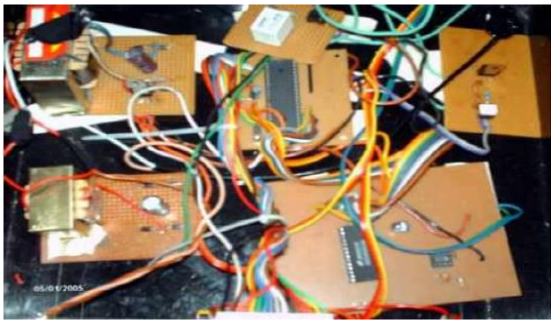

Figure 4. 4: Internal Circuit

Once the device is connected to a power supply and is powered on, the device first of all initialises and senses the temperature and humidity within the environment at that moment. As explained in figure 4.5

Figure 4. 5: Reading of the Current Room Temperature

Immediately after it has displayed the current room temperature it will also display the current environment humidity as shown in figure 4.6

Figure 4 6: Reading of the Current Room Humidity

Now you have you have the value of the temperature and humidity within the incubator. The operator is expected here to input the value for the temperature and humidity at which he/she want for the incubator, the operator can do so with the aid of the input push button on the device, the operator should hold down the SET BUTTON till we have "SET NEW TEMP" as shown in figure 4.7

Figure 4. 7: Temperature Set Display

After the required temperature is set, "SET NEW HUMIDITY" will appear as shown in figure 4.8

Figure 4. 8: Humidity Set Display

Once the humidity is set to the required humidity then the set button is pressed to implement the device. The system designed for the temperature monitoring of the infant incubator worked successfully on a dummy model. However the system is yet to be tested in a practical incubator.

4.3 Discussions

The device is very easy to operate because the microcontroller does most the work in the design. I will elaborately discussion the design of the device, how the device operates and how it performs its functions.

4.3.1 The Device Operation

Microcontroller based temperature and humidity sensor for infant incubator. Basically without writing anything we should be able to understand how the device works. I will briefly explain how the device performs its function. Once the device is ON as shown in figure 4.2 and the sensor has sensed the temperature and humidity within the environment at that moment, the operator then needs to set the new value for the temperature and for the humidity as shown in figure 4.7 and figure 4.8 respectively. Once the temperature and humidity has been set the microcontroller starts it main function, it checks firstly if the temperature of the room corresponds to the New set temperature and the humidity. Once the temperature of the room is lower that means the humidity is higher the microcontroller turns on the relay switch and power on the heater, and heat up the environment till it gets to the set temperature and humidity the heater is then turned off, just as figure 4.2 explained in details.

4.3.2 Area of Applications

Basically the device is designed and developed to monitor and control the temperature and humidity using a microcontroller within an infant incubator. The device can also be used in Office, Home, and Hospital in an environment where temperature is very critical for the growth and to sustain life.

4.3.3 Economic importance of infant incubator

The benefit of a microcontroller based temperature and humidity sensor for and infant incubator is that the device aid monitoring when the temperature is (not) favourable for growth for the infant in an incubator, among others:

- Home management insights: There's also something to be said for your ability to tap into insights on how the baby reacts to it environment. (Smart, 2017).
- Lifesaving: The device also provide a uniform airflow and humidity can also be controlled to the desired level. No more wasting money on supervisors to monitor their welfare always (Benefits , 2017).
- Health problem addressed: At birth, an infant's core and skin temperatures tend to drop significantly because of heat loss from conduction, convection, radiation, and water evaporation. Prolonged cold stress in neonates can cause oxygen deprivation, hypoglycaemia, metabolic acidosis, and rapid depletion of glycogen stores (WHO, 2011).

5.0 Conclusion

The set aim of this work was accomplished by developing a microcontroller based humidity and temperature controller for infant incubator. Some of the future aspects of the work in terms of its improvement are discussed below;

- Presently only air temperature control mode which measures temperature from infant's chamber air has been used. We can enhance the accuracy of system by introducing skin temperature control mode.
- Parameters such as pulse measurement can also be introduced for close monitoring.
- Buzzer can also be included to alert the doctor or the operator at any uncomfortable conditions.
- Webcam that helps the mother and the doctor to see the baby to make sure about his health condition.
- The design and development of a microcontroller based temperature and humidity controller for infant incubator can be very easy to implement and operate.

REFERENCES

Ahmed, W. H. (2011). *Development of baby incubator.* Khartoum.

Atmega. (2017, FEBRUARY 9TH). *ATmega328 - wikipedia*. Retrieved APRIL 5th, 2017, from Wikipedia: https://en.wikipedia.org/wiki/ATmega328

Atmel. (2009, MAY). *ATmega48PA/88PA/168PA/328P / Datasheet*. Retrieved JULY 21ST, 2017, from Mouser.com: http://www.mouser.com/pdfdocs/Gravitech_ATMEGA328_datasheet.pdf

Benefits. (2017, August 28). Retrieved from sensicomfort web site: https: // sensicomfort.com/blog/home-automation-benefits

Changela, B., Daxini, B., & Parmar, K. (2016). Temperature Dector For Premature Infant Incubator Using AVR Microcontroller. *International Journal of Engineering Trends and Technology (IJETT)*. Retrieved MARCH 17th, 2017, from http://www.ijettjournal.org/2016/volume-35/number-7/IJETT-V35P268.pdf

Electronic Tutorials. (2013, July 13). Retrieved from Electronicstutorial: www.electronics-tutorials.ws/blog/relay-switch-circuit.html

Jefferey, P. (2000). The Incubator and the Medical Discovery of the Premature Infant. *Journal of Perinatology*. Retrieved MARCH 17TH, 2017, from NEONATOLOGY: http://neonatology.org/pdf/7200377a.pdf

Mohammed, N. (2011). Retrieved MARCH 22nd, 2017, from khartoumspace.uofk.edu:8080/.../Design%20And%20Development%20.pdf?.

Smart. (2017, August 27). *Smart-home-automation.* Retrieved from Bluespeedav: http://bluespeedav.com/blog/item/7-greatest-advantages-of-smart-home-automation.

Vishay. (2002). *16 x 2 Character LCD.* WHO. (2011). Incubator, Infant. *Core Medical Equipment, Information*.

APPENDIX A

```
//Include lcd and temperature sensor library
#include <LiquidCrystal.h>
LiquidCrystallcd(13, 12, 11, 10, 9, 8);
#include "DHT.h"
//Declare connection pins
intswitch_relay = 6;
floatset_value_temp = 71;
floatset_value_humid = 60;
float f, h;
#define DHTTYPE DHT11   // DHT 11
#define DHTPIN 7     // what digital pin we're connected to
// as the current DHT reading algorithm adjusts itself to work on faster procs.
DHT dht(DHTPIN, DHTTYPE);
//Set Configuration
void setup() {
//Set LCD property i.e. 16 x 2
lcd.begin(16,2);
//Declare pin 6 as output pins
pinMode(switch_relay,OUTPUT);
//Initiate the sensor
dht.begin();
//Display this information on the LCD
lcd.print("TEMP & HUMDITY");
lcd.setCursor(2,1);
lcd.print("MONITORING");
delay(1000);                }
//Start
void loop() {
lcd.clear();
lcd.setCursor(1,0);
lcd.print("Set Temp. Value");
lcd.setCursor(5,1);
lcd.print(set_value_temp);
lcd.print((char)223);
lcd.print("F");
delay(1000);
lcd.clear();
lcd.setCursor(1,0);
lcd.print("Set Humd. Value");
lcd.setCursor(5,1);
lcd.print(set_value_humid);
lcd.print("%");
//Delay for 1s (1000ms)
```

```
delay(1000);
//Read humidity value
h = dht.readHumidity();
//Read temperature value in Farahenit
f = dht.readTemperature(true);
// Check if any reads failed and exit early (to try again).
if (isnan(f)||isnan(h)) {
//Serial.println("Failed to read from DHT sensor!");
return;          }
//Set the required regulated temperature
setTempValueAndHumidity();
//Check the temperature and compare with set temperature value
switchOffHeater();
//Check if the temperature of the room is lower than the set value and heat the room
if(!digitalRead(switch_relay)){
if (f <= (set_value_temp-3)){
digitalWrite(switch_relay,HIGH);         }         }
//Show the Room temperature on LCD
ShowValuesOnLCD();
ShowHumdityWarning(); }
//Program for ShowHumdityWarning
voidShowHumdityWarning(){
if(h >set_value_humid){
lcd.clear();
lcd.setCursor(0,0);
lcd.print("HUMIDITY WARNING!!!");
lcd.setCursor(5,1);
lcd.print(h);
lcd.print("%");
delay(1000);                    }         }
voidswitchOffHeater(){
if(f >= set_value_temp){
digitalWrite(switch_relay,LOW);                   }         }
//Program for setting New Temperature and Humidity value for monitoring
voidsetTempValueAndHumidity(){
if(digitalRead(2)){
lcd.clear();
lcd.setCursor(1,0);
lcd.print("Set New set points");
delay(1000);
lcd.clear();
while(1){
if(digitalRead(3)){
set_value_temp++;
//delay(300);        }
```

```
if(digitalRead(4)){
if(set_value_temp>0){
set_value_temp--;
//delay(300);
}else{
set_value_temp =0;  }        }
lcd.setCursor(1,0);
lcd.print("Set New Temp.");
lcd.setCursor(5,1);
lcd.print(set_value_temp);
lcd.print((char)223);
lcd.print("F");
delay(300);
//Set New Humdity Set Point
if(digitalRead(2)){
while(1){
if(digitalRead(3)){
set_value_humid++;
//delay(300);         }
if(digitalRead(4)){
if(set_value_humid>0){
set_value_humid--;
//delay(300);
}else{
set_value_humid =0;         }         }
lcd.clear();
lcd.setCursor(1,0);
lcd.print("Set New Humidity
9");
lcd.setCursor(5,1);
lcd.print(set_value_humid);
lcd.print("%");
delay(300);
if(digitalRead(2))break;   }           }
if(digitalRead(2))break;   }         }         }
//Program for displaying Room temperature and Room Humidity on LCD
voidShowValuesOnLCD(){
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Room Temp Value: ");
lcd.setCursor(5,1);
lcd.print(f);
lcd.print((char)223);
lcd.print("F");
delay(2000);
```

```
lcd.clear();
lcd.setCursor(0,0);
lcd.print("     Humdity: ");
lcd.setCursor(5,1);
lcd.print(h);
lcd.print("%");
delay(2000);
}
```