Scientific Journal of Impact Factor (SJIF): 5.71

e-ISSN (O): 2348-4470
p-ISSN (P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 5, Issue 04, April -2018

DEVELOPMENT OF DHT11 AND 8X8 DOT MATRIX LED LIBRARIES FOR TEMPERATURE DISPLAY SYSTEM BASED ON ARDUINO

Mr. Vihang H Trivedi[1] Ms. Parveen Sheikh[2]

[1 2]*Assistant Professor , Department of Electronics & Communication K.J. Institute of Engineering & Technology, Savli*

Abstract - *This paper is about how to write Libraries for DHT11 (Temperature & Humidity Sensor) and 8x8 Dot matrix LED (1088BS). DHT11 has single data pin which can be used as bidirectional pin, so before reading DHT11 Arduino has to send the signal for getting back data from DHT11. DHT11 provides 40 bits (5 bytes) of data. 2 bytes for integral and decimal part of humidity, 2 bytes for integral and decimal part of temperature and final byte will provide checksum.The pins of 8x8 dot matrix LED needs to be identified first. Once the pins are known the library can be written for it. For this pape,r 8x8 dot matrix LED doesn't require driver circuit. 8x8 dot matrix LED can be connected through wires with the port pins of Arduino. It can display temperature up to 2 digits means 0 to 99 ℃.*

Key Words*- Arduino, DHT11, 8x8 dot matrix LED, library files, Temperature display*

I. INTRODUCTION

To develop an application that can display temperature on 8x8 dot matrix LED, first thing is to write library for both DHT11 and dot matrix LED. This paper explores section by section how to write library, followed by a common Arduino code describing interfacing with DHT11 and 8x8 dot matrix LED. DHT11 require 3 pins to be connected with Arduino if no extra power supply is available. DHT11 requires around 4ms to complete the communication process via serial communication [1]. 8x8 dot matrix LED requires 16 pins to be connected when no driver circuit is used. Arduino pins 3 to 18 are occupied for dot matrix LED. Pin 19 is used to read from DHT11. The scenario is quite simple but development of library is the most critical part for Arduino. Without developing library, the code for the application is lengthy and hence ambiguous sometimes .

II. DHT11 Library Development for Arduino

The Power supply requirement for DHT11 is 3.3-5V which is to be connected to pin 1 of DHT11. Pin 2 will be used for Data and Pin 4 is connected to ground as shown in figure 2.1. Pin 2 requires a pull-up resistor of 10kΩ.

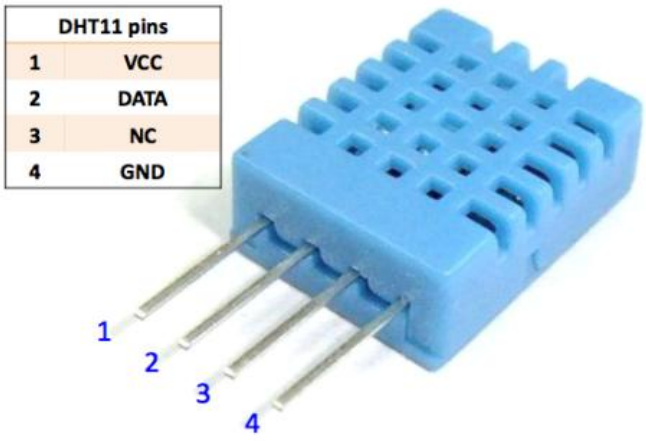

Figure 2.1 DHT11 pinout [1]

Pin 19 of Arduino is connected with pin 2 of DHT11. Communication will start when Arduino pin 19 will send "start" signal of around 18 ms of Low level and then will go High again. After sending signal, Arduino should wait for response from DHT11. DHT11 will transmit data bit by bit after every 50µs. Logic "1" and "0" is being recognized via width of the pulse.

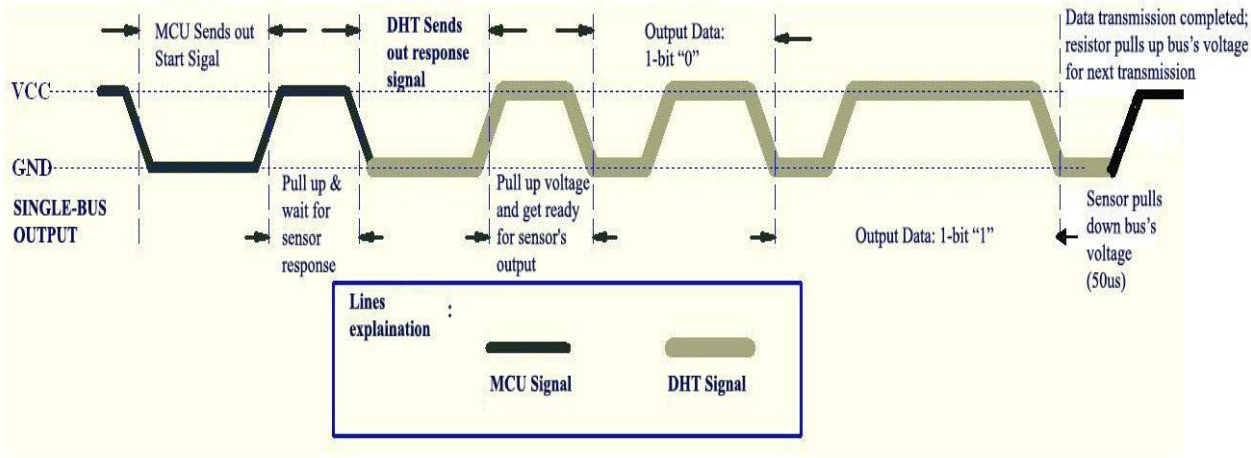

Figure 2.2 Communication Protocol for DHT11 [1]

The following code is required to develop library of DHT11 based on the above mentioned description:

```
for (int i=0;i<5;i++) bytes[i]=0;
        pinMode(pin,OUTPUT);
        digitalWrite(pin,LOW);
        delay(18);
        digitalWrite(pin,HIGH);
        delayMicroseconds(40);
        pinMode(pin,INPUT);
```

The above mentioned part of the code describes how a pulse is sent to data pin of DHT11 so that it can start to respond. After this Arduino has to wait for response, and once Arduino receives the response, it must start reading from DHT11.

```
        unsigned int cnt=10000;
                while(digitalRead(pin)==LOW)
                        if(cnt-- == 0) return 0;
        cnt==10000;

                while(digitalRead(pin)==HIGH)
                if(cnt-- == 0) return 0;
                for(int i=0;i<40;i++)
                {
                cnt=10000;
                        while(digitalRead(pin)==LOW)
                                if(cnt-- ==0) return 0;
                        unsigned long t= micros();
                        cnt=10000;
                        while(digitalRead(pin)==HIGH)
                                if(cnt-- ==0) return 0;
                        if((micros()-t)>40) bytes[k] |= (1<<cnt);
                        if(cnt==0)
                        {
                        cnt=7;
                        k++;
                        }
                        else cnt--;
                }
                humidity=bytes[0];
                temperature=bytes[2];
                uint8_t sum=bytes[0] + bytes[2];
                if(bytes[4] != sum)
                        return 1;
        }
```

The above code is part of function named read(). For writing Arduino code, use *ObjectName*.read(pin number).
To complete the library, header file must be written. In header file, class is defined as follows:

```
class dht11
{
```

```
public:
dht11(int pin);
int read(int pin);
int humidity;
int temperature;

private:
int _pin;
};
```

After writing both files save it as dht11.h and dht11.cpp. Create zip files so that library can be added to Arduino.

III. 8x8 Dot Matrix LED library development for Arduino

8x8 dot matrix LED has 16 pins. Without driver circuit these 16 pins are needed to be connected with Arduino. Before connecting LED to Arduino, first thing is to decide mapping between row and column with LED pins. There are 8 pins for 8 rows and 8 pins for 8 columns. Using connectivity in digital multimeter pins are found as below.

Table 3.1 Pins of 8x8 dot matrix LED (pin 1 is on right corner of written number side)

8-Row6	7-Row3	6-Column5	5-Row1	4-Column3	3-Column2	2-Row 2	1- Row4
9-Row8	10-Column4	11-Column6	12-Row5	13-Column1	14-Row7	15-Column7	16-Column8

Figure 3.1 8x8 dot matrix LED [2]

To display 2 digits, columns are divided in to 2, so that 4 columns are given to each digit. A 2-D array of Left digit and Right digit is stored in library.

```
uint8_t L[10][8]={
                    {0x00,0x70,0x50,0x50,0x50,0x50,0x50,0x70},
                    {0x00,0x60,0x20,0x20,0x20,0x20,0x20,0x70},
                    {0x00,0x70,0x10,0x10,0x70,0x40,0x40,0x70},
                    {0x00,0x70,0x10,0x10,0x70,0x10,0x10,0x70},
                    {0x00,0x50,0x50,0x50,0x70,0x10,0x10,0x10},
                    {0x00,0x70,0x40,0x40,0x70,0x10,0x10,0x70},
                    {0x00,0x70,0x40,0x40,0x70,0x50,0x50,0x70},
                    {0x00,0x10,0x10,0x10,0x10,0x10,0x10,0x10},
                    {0x00,0x70,0x50,0x50,0x70,0x50,0x50,0x70},
                    {0x00,0x70,0x50,0x50,0x70,0x10,0x10,0x70}
                    };

uint8_t R1[10][8]={
                    {0x00,0x07,0x05,0x05,0x05,0x05,0x05,0x07},
                    {0x00,0x06,0x02,0x02,0x02,0x02,0x02,0x07},
                    {0x00,0x07,0x01,0x01,0x07,0x04,0x04,0x07},
                    {0x00,0x07,0x01,0x01,0x07,0x01,0x01,0x07},
                    {0x00,0x05,0x05,0x05,0x07,0x01,0x01,0x01},
                    {0x00,0x07,0x04,0x04,0x07,0x01,0x01,0x07},
                    {0x00,0x07,0x04,0x04,0x07,0x05,0x05,0x07},
                    {0x00,0x01,0x01,0x01,0x01,0x01,0x01,0x01},
                    {0x00,0x07,0x05,0x05,0x07,0x05,0x05,0x07},
                    {0x00,0x07,0x05,0x05,0x07,0x01,0x01,0x07}
                    };
```


Here L array is for left digit and R1 is for Right digit.These 2-D arrays have total 10 elements each having 8 row values to complete the display of a single digit. For example to display number 7 on left, we require to fetch L[7][0] to L[7][7]. Where L[7][0] for ROW 1 and L[7][7] for ROW 8. Finally note that, for combining both values together just do OR operation as follows: L[3][0] | R1[7][0] to L[3][7] | R1[7][7]. This OR operation will display 37 on dot matrix LED. To get input from Arduino, Library must include function which will accept 2 inputs.

```
DMLed8by8::displayLed(uint8_t k,uint8_t k1)
{
uint8_t qq[8];
uint8_t qq1;
for(uint8_t cnt1=0;cnt1<255;cnt1++)
{
for (uint8_t q=0;q<8;q++)
        {qq[q]=L[k][q] | R1[k1][q];
        qq1=~qq[q];
                for(uint8_t q1=0;q1<8;q1++)
                { digitalWrite(C[q1],((qq1>>q1)&0x01));}
                digitalWrite(R[q],HIGH);
                delay(2);
                digitalWrite(R[q],LOW);
}
}
```

Where DMLed8by8 is class defined in header file. k and k1 are inputs taken from main Arduino code. 2 for loops are used, first one for Rows and Second one for Columns. Finally save header file and c++ file with name same as class name. Save it to the same location and create .zip file of it so that Arduino can add it as library. Header file should include the following:

```
class DMLed8by8
{
public:
DMLed8by8(uint8_t pin1,uint8_t pin2,uint8_t pin3,uint8_t pin4,uint8_t pin5, uint8_t pin6,uint8_t pin7, uint8_t pin8,uint8_t pin9, uint8_t pin10, uint8_t pin11, uint8_t pin12,uint8_t pin13,uint8_t pin14,uint8_t pin15,uint8_t pin16);
displayLed(uint8_t lTemp,uint8_t rTemp);
private:
uint8_t R[8],C[8];
};
```

Pin1 to Pin 16 can be defined discrete. “displayLed” function will get Left digit and Right digit from Arduino code.

IV. Temperature Display System

To develop Temperature display system both library of DHT11 and 8x8 dot matrix LED must be included.Once it is included, follow the following flow chart for the system.

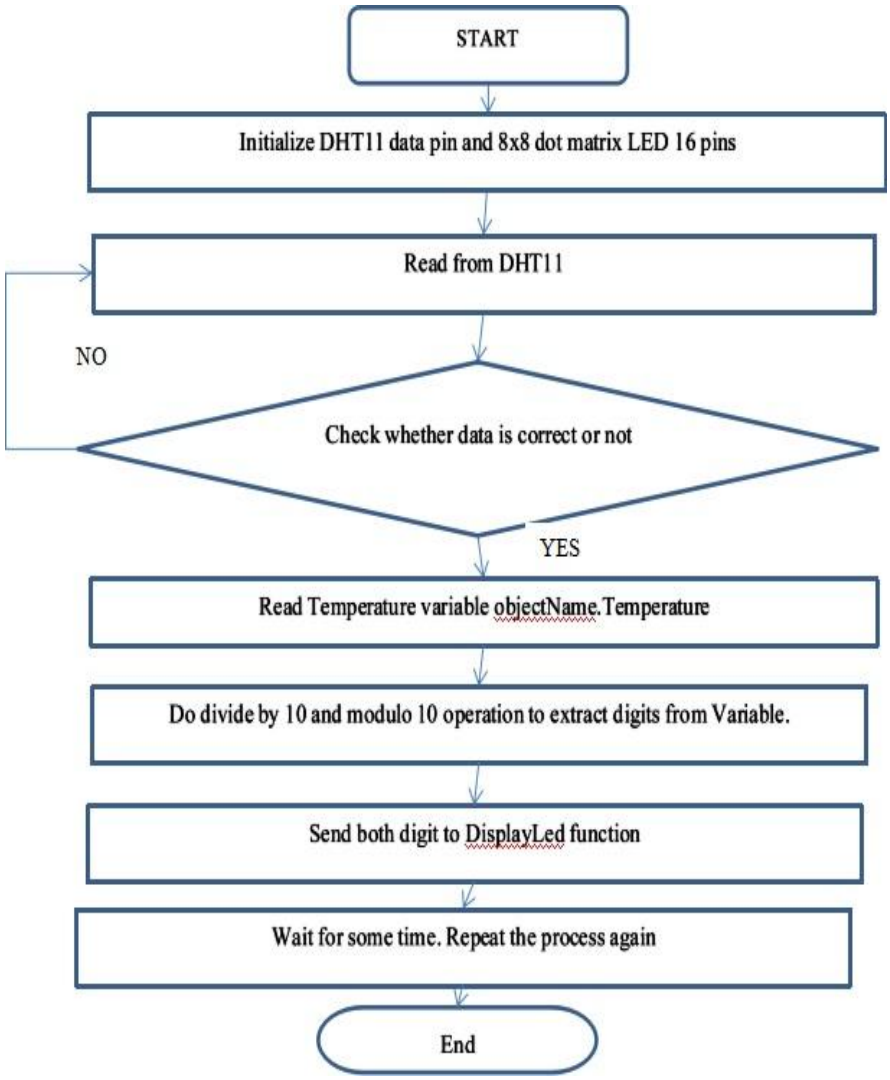

Figure 4.1 Flowchart of Temperature Display System

After following the flow chart Arduino code will look like

```
#include <dht11.h>
#include <DMLed8by8.h>
int chk;
uint8_t lTemp,rTemp;
dht11 d1(19); //pin 19 will connect with pin number 2 of DHT11
DMLed8by8 dml1(3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18); // pins 3 to 10 for Row1 to Row 8, pins 11 to 18
                                                          //for Column 8  to column 1
void setup() {
}
void loop() {
chk=d1.read(19); // chk variable will have value 0 or 1. if 1 then data correct otherwise false
if(chk!=0)
{
 lTemp=(d1.temparature)/10; //Higher Digit
  rTemp=(d1.temparature)%10; // Lower Digit
  dml1.displayLed(lTemp,rTemp);
  delay(5);
}
}
```


V. Result

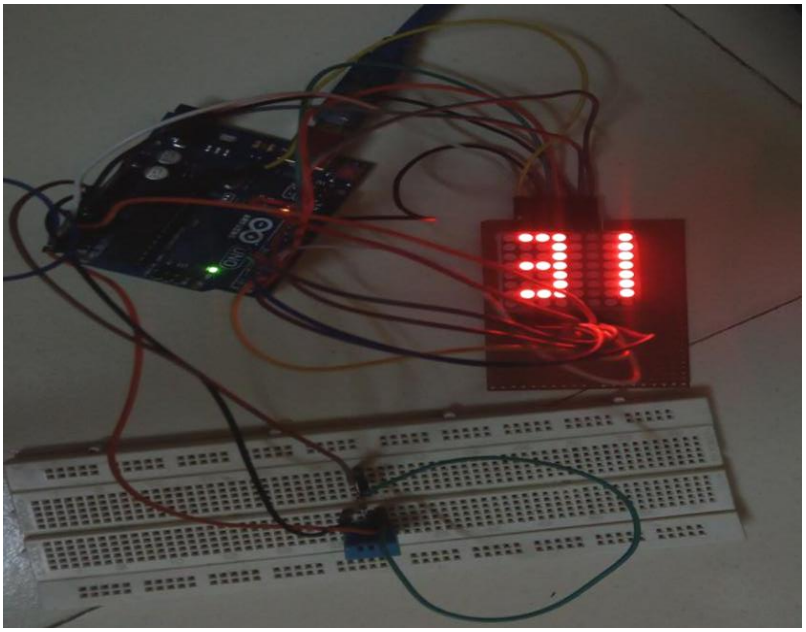

Figure 5.1 Output of Temperature display system

This paper hence have described about how a 8x8 dot matrix LED and DHT11 are interfaced with Arduino to display temperature sensed by DTH11 by using libraries which has reduced complexity of the code and has made it quite handy.

Conclusion

It is quite easy to implement the system when library has already being developed. Here we have just displayed Temperature value without its unit and also we can display humidity data on it. To do so either we can add in library files or add Arduino code. We can also increase number of displays but logic will be same only extra driver circuits will be required.

References

[1] Datasheet of DHT11 "http://www.micropik.com/PDF/dht11.pdf"
[2] Datasheet of 1088BS 8x8 Dot matrix LED
[3] Datasheet of ATMEGA328P http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf

