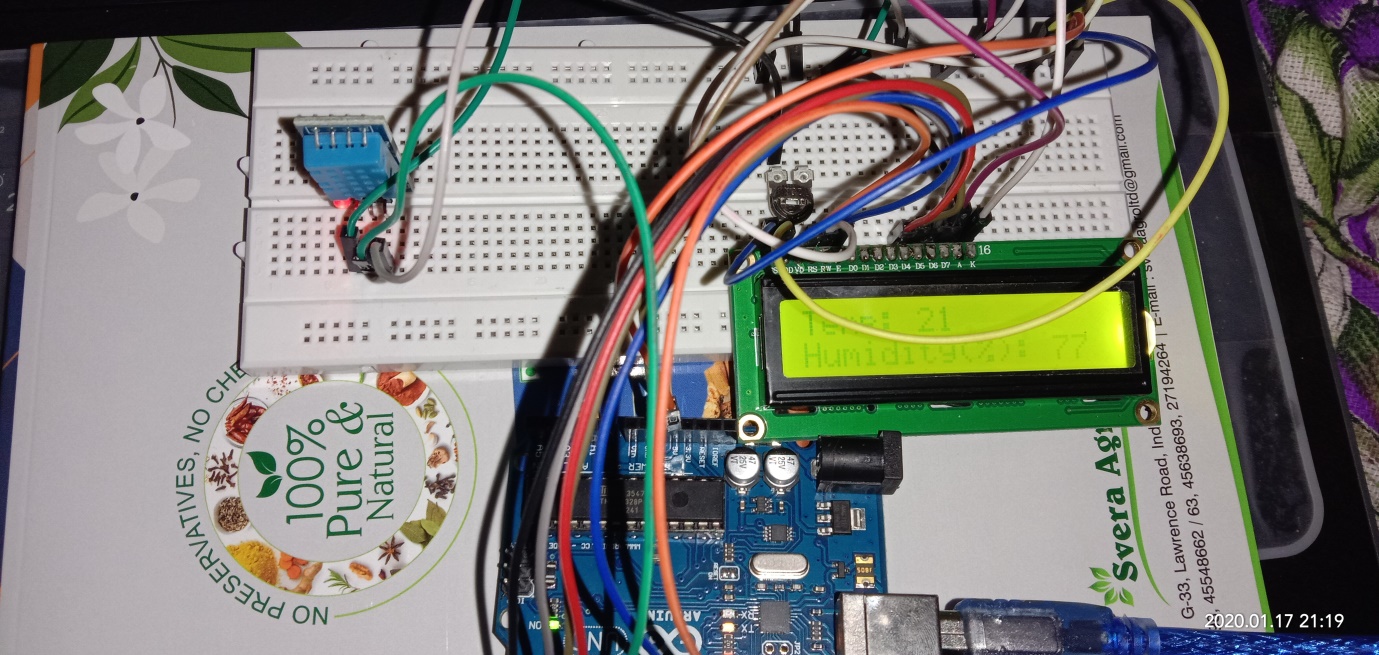
**Weather station using arduino**

Using the display to view the temperature and humidity of our environment can be possible using the dht11 with easy to use arduino microcontroller platform and that’s the goal of our project.



**Component Required:-**

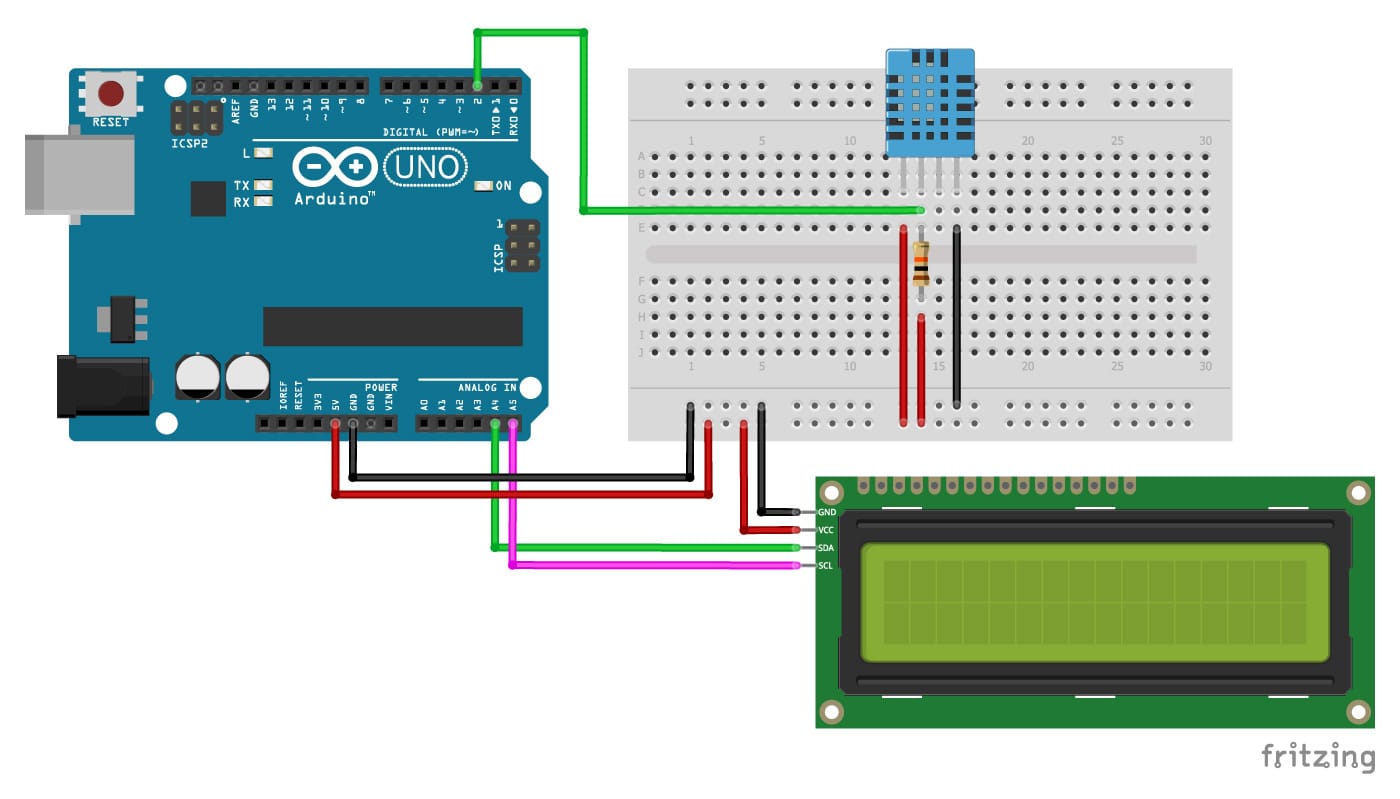
1.Arduino UNO

2.DHT11

3.LCD16\*2

4.Potentiometer

5.jumper wires

**Circuit diagram:**

**Programming for weather station**:

#include &lt;LiquidCrystal.h&gt;

#include &lt;SimpleDHT.h&gt;

//Declaring digital pin no 6 as the dht11 data pin

int pinDHT11 = 6;

SimpleDHT11 dht11;

//Declaring the lcd pins

const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

void setup() {

// Don&#39;t forget to choose 9600 at the port screen

Serial.begin(9600);

//Telling our lcd to start up

lcd.begin(16, 2);

}

void loop() {

//These serial codes are for getting readings on the port screen aswell as the LCD display, since

they&#39;ll offer us a more detailed interface

Serial.println(&quot;=================================&quot;);

Serial.println(&quot;DHT11 readings...&quot;);

byte temperature = 0;

byte humidity = 0;

int err = SimpleDHTErrSuccess;

//This bit will tell our Arduino what to do if there is some sort of an error at getting readings

from our sensor

if ((err = dht11.read(pinDHT11, &amp;temperature, &amp;humidity, NULL)) != SimpleDHTErrSuccess)

{

Serial.print(&quot;No reading , err=&quot;); Serial.println(err);delay(1000);

return;

}

Serial.print(&quot;Readings: &quot;);

Serial.print((int)temperature); Serial.print(&quot; Celcius, &quot;);

Serial.print((int)humidity); Serial.println(&quot; humidity%&quot;);

//Telling our lcd to refresh itself every 0.75 seconds

lcd.clear();

//Choosing the first line and row

lcd.setCursor(0,0);

//Typing Temp: to the first line starting from the first row

lcd.print(&quot;Temp: &quot;);

//Typing the temperature readings after &quot;Temp: &quot;

lcd.print((int)temperature);

//Choosing the second line and first row

lcd.setCursor(0,1);

//Typing Humidity(%): to the second line starting from the first row

lcd.print(&quot;Humidity(%): &quot;);

//Typing the humidity readings after &quot;Humidity(%): &quot;

lcd.print((int)humidity);

delay(750);

}

Application:

1. Automation leads to better monitoring of device.

2. Efficient of save time.

3. Save time.