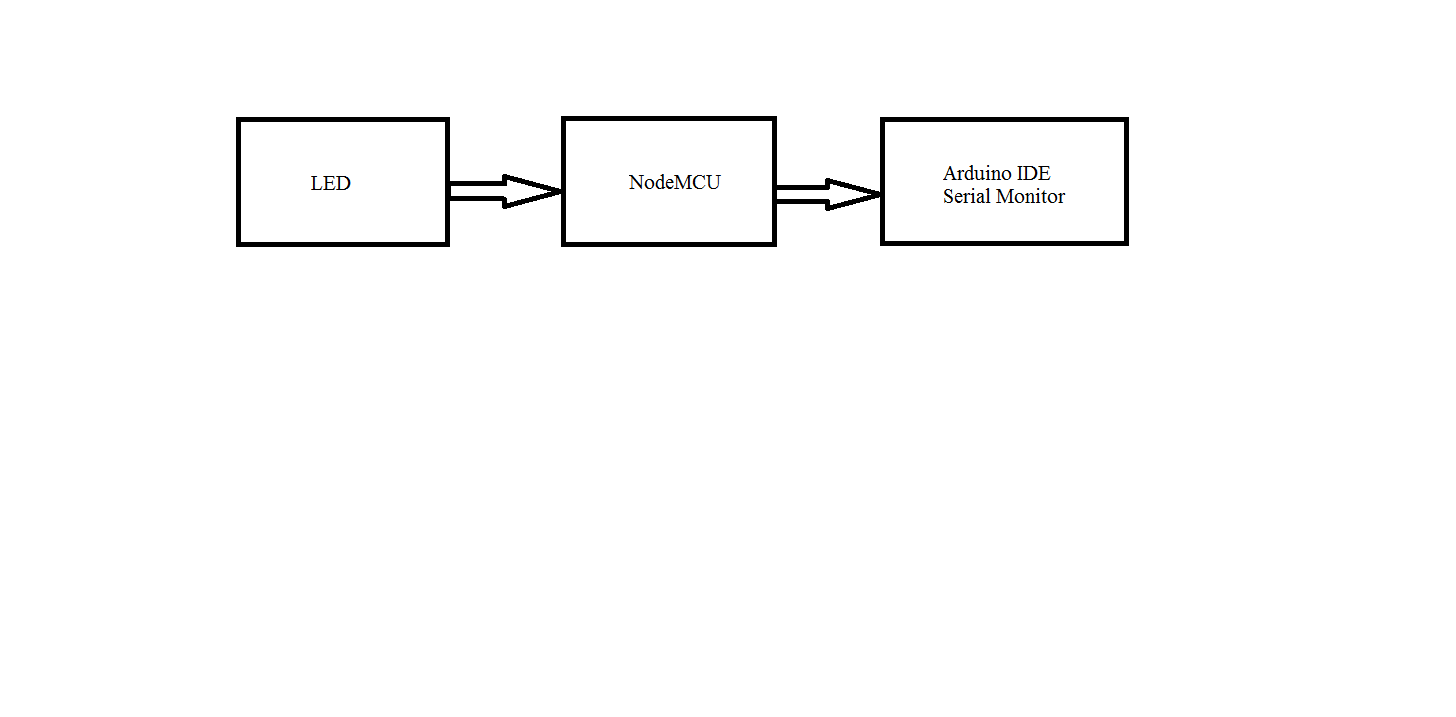
**GROUP 9**

**EXPERIMENT 1A**

**OBJECTIVE:** LED BLINK ON/OFF

**BLOCK DIAGRAM:**

****

**FIG 1A**

**EXPLANATION:**

The LED is connected with the resistor in breadboard. Then LED is connected to D0 pin (GPIO16) of NodeMCU and based on a high or low voltage output, the LED switches on and off.

**APPARATUS:**

* LED
* Resistor
* NodeMCU
* Jumper wire
* Arduino IDE
* PC
* Micro USB cable

**PROGRAM:**

#define LED D3 // Led in NodeMCU at pin GPIO16 (D0).

void setup () {

pinMode (LED, OUTPUT); // set the digital pin as output.

}

void loop () {

digitalWrite (LED, HIGH); // turn the LED off. (Note that LOW is the voltage level but actually

//the LED is on; this is because it is active low on the ESP8266.

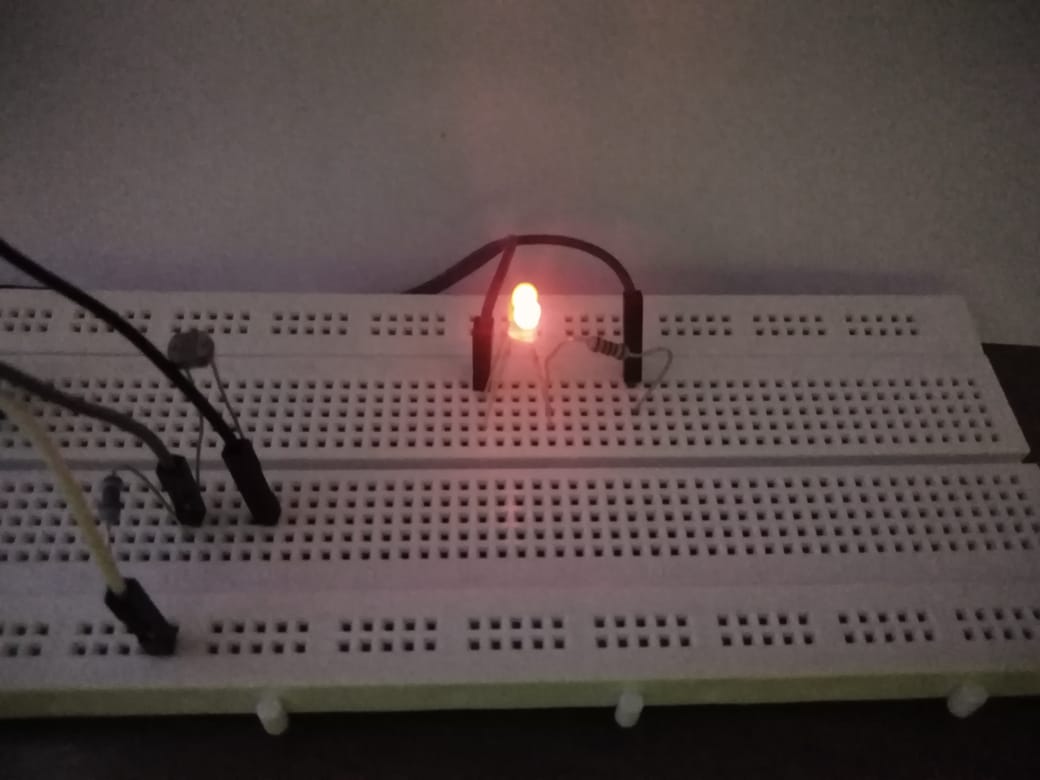
delay (1000); // wait for 1 second.

digitalWrite (LED, LOW); // turn the LED on.

delay (1000); // wait for 1 second.

}

**RESULT:**

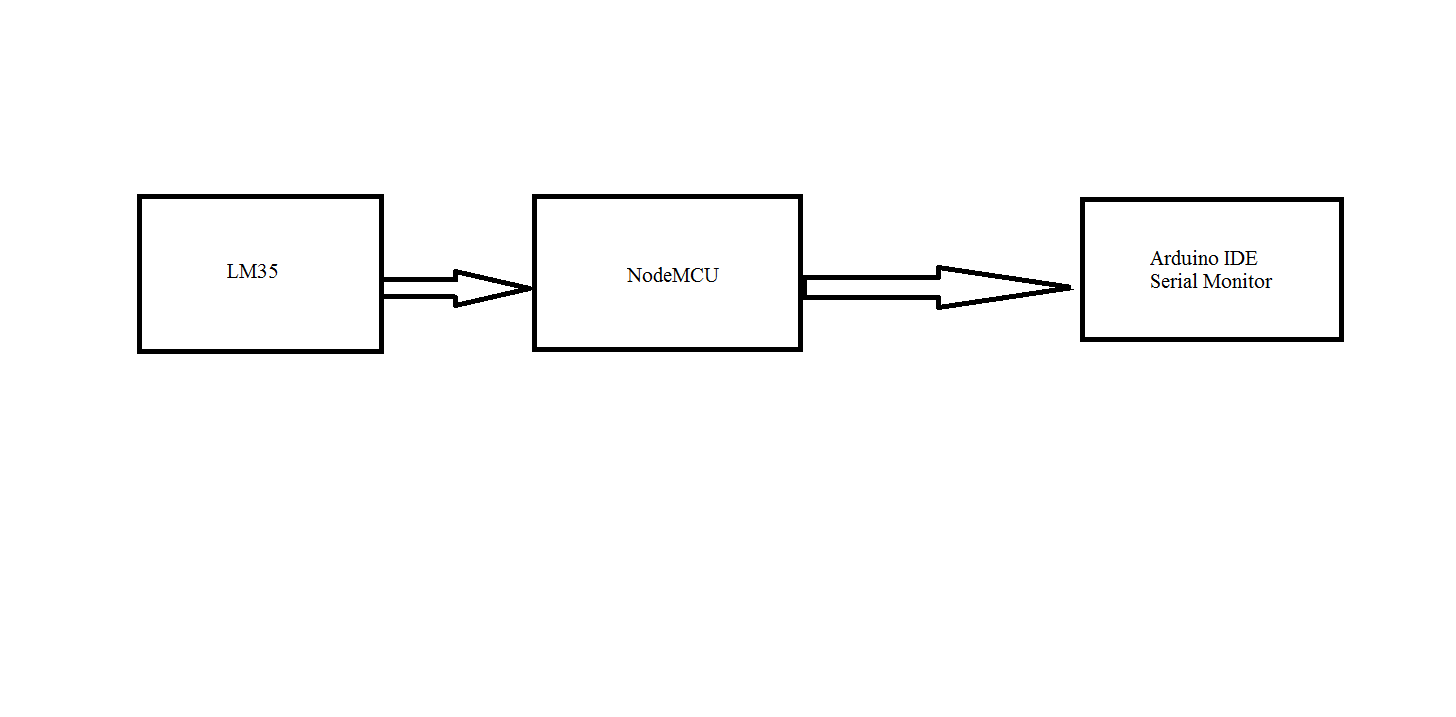
****

**Fig 1B**

**EXPERIMENT 1B**

**OBJECTIVE:** READ LM35 IN ARDUINO SERIAL MONITOR

**BLOCK DIAGRAM:**

****

**FIG 2A**

**EXPLANATION:**

LM35 is connected via NodeMCU and the room temperature value detected by it is displayed in the Arduino IDE Serial Monitor.

**APPARATUS:**

* LM35 temperature sensor
* Jumper wires
* NodeMCU
* Micro USB Cable
* PC
* Arduino IDE

**PROGRAM:**

#include <ESP8266WiFi.h>;

const int sensor=A0; // Assigning analog pin A0 to variable 'sensor'

float tempc; //variable to store temperature in degree Celsius

float tempf; //variable to store temperature in Fahrenheit

float vout; //temporary variable to hold sensor reading

void setup ()

{

pinMode (sensor, INPUT);

Serial.begin (115200);

}

void loop () {

vout=analogRead (sensor);

vout= (vout\*3300)/1024;

tempc=vout/10; // Storing value in Degree Celsius

tempf= (vout\*1.8) +32; // Converting to Fahrenheit

Serial.print ("in DegreeC=");

Serial.print ("\t");

Serial.print (tempc);

Serial.print ();

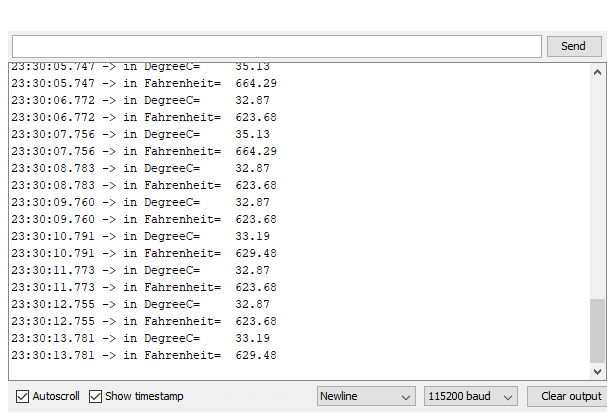
Serial.print ("in Fahrenheit=");

Serial.print ("\t");

Serial.print (tempf);

Serial.print ();

delay (1000); }

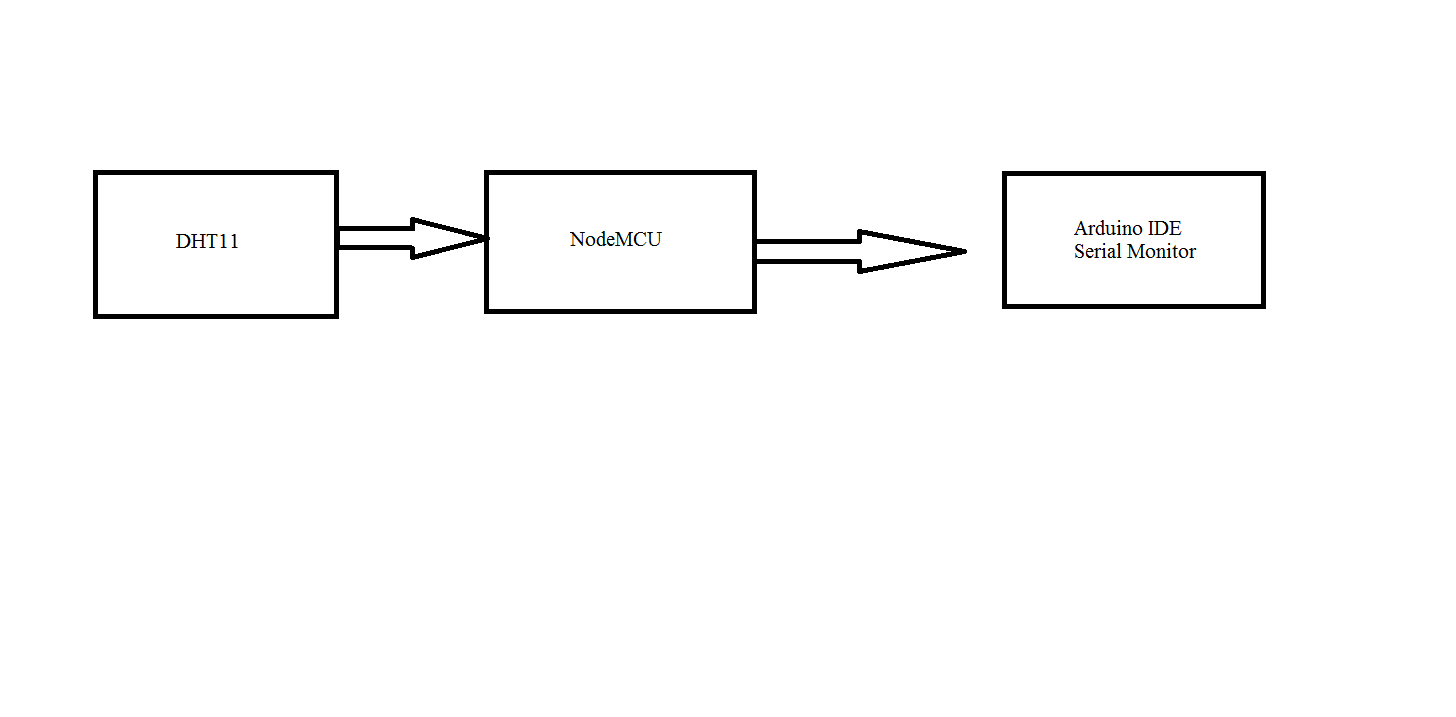
**RESULT: **

**FIG 2B**

**EXPERIMENT 1C**

**OBJECTIVE:** READ DHT11 IN ARDUINO SERIAL MONITOR

**BLOCK DIAGRAM:**

****

**Fig 3A**

**EXPLANATION:**

DHT11 is connected via NodeMCU and the temperature value detected by it is shown through the Arduino IDE Serial Monitor.

**APPARATUS:**

* DHT11 humidity sensor
* Jumper wires
* NodeMCU
* Micro USB Cable
* PC
* Arduino IDE

**PROGRAM:**

#include <WiFiClient.h>;

#include <ESP8266WiFi.h>;

#include <DHTesp.h>;

#include <stdlib.h>;

#defineDHTpin D3

DHTespdht;

WiFiClient client;

void setup ()

{

Serial.begin (115200);

dht.setup (DHTpin, DHTesp::DHT11);

}

void loop ()

{

float h = dht.getHumidity ();

float t = dht.getTemperature ();

// set the fields with the values

delay (1000);

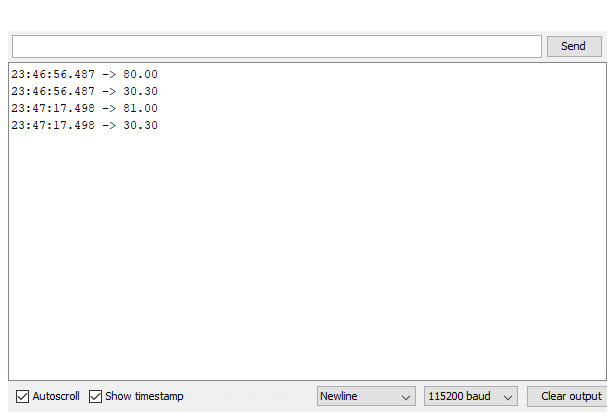
Serial.print (h);

Serial.print (t);

delay (20000); // Wait 20 seconds to update the channel again

}

**RESULT:**

****

**FIG 3C**