* MATERIALS REQUIRED:

1. Arduino Nano Board

2. ESP8266-01 WiFi Module

3. 16x2 LCD Display

4. Potentiometer 10K

5. Pulse Sensor

6. LM35 Temperature Sensor

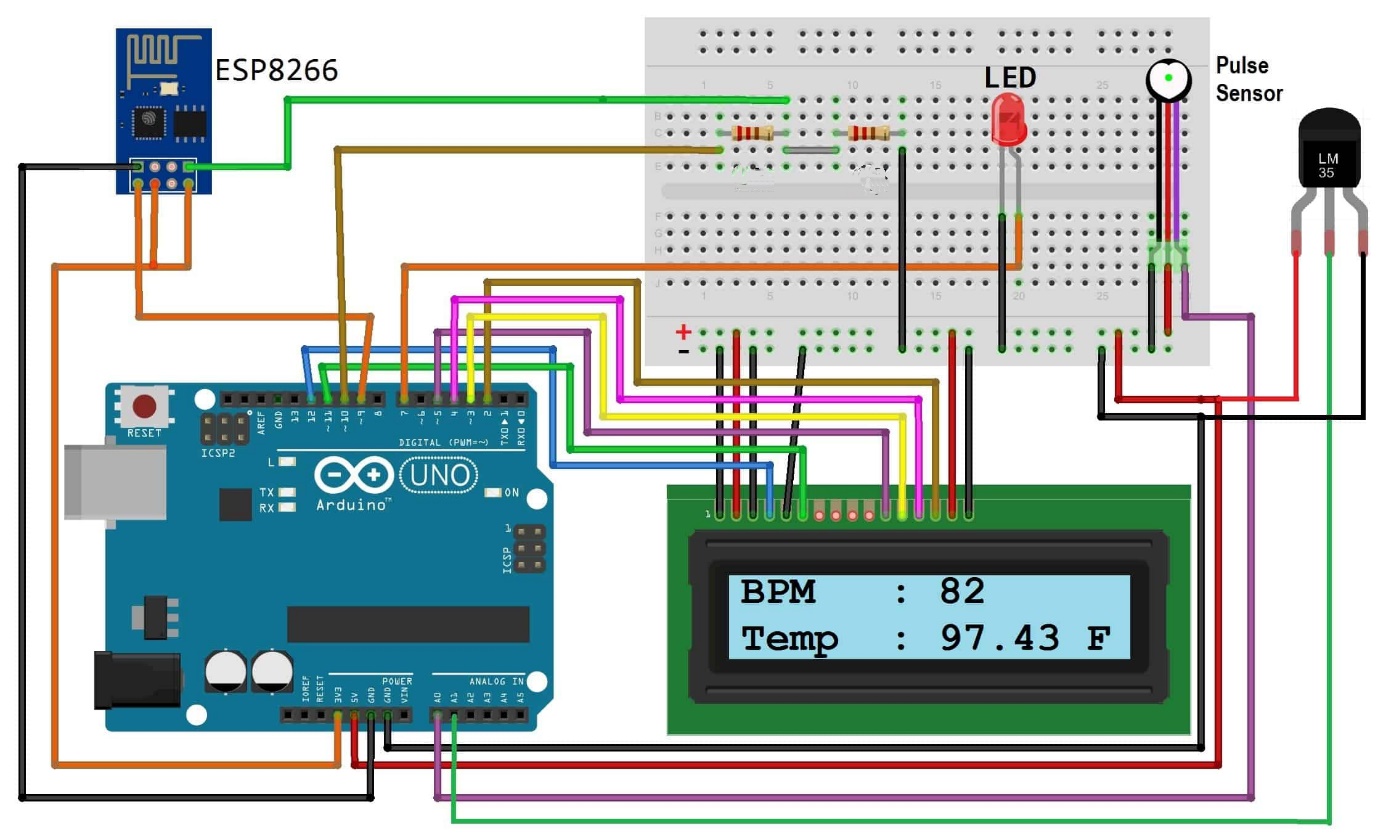
7. 2K Resistor

8. LED

9. Connecting Wires

10. Breadboard

* CIRCUIT DIAGRAM:



* PROGRAM:

#include <LiquidCrystal.h>

#include <SoftwareSerial.h>

#include <OneWire.h>

#include <DallasTemperature.h>

#define USE\_ARDUINO\_INTERRUPTS true

#include <PulseSensorPlayground.h>

SoftwareSerial esp(10, 11);

LiquidCrystal lcd(7, 6, 5, 4, 3, 2);

#define ONE\_WIRE\_BUS 9

#define TEMPERATURE\_PRECISION 12

OneWire oneWire(ONE\_WIRE\_BUS);

DallasTemperature sensors(&oneWire);

DeviceAddress tempDeviceAddress;

int numberOfDevices, temp, buzzer = 8;

const int PulseWire = A0;

int myBPM, Threshold = 550;

PulseSensorPlayground pulseSensor;

unsigned long previousMillis = 0;

const long interval = 5000;

void setup()

{

lcd.begin(16, 2);

Serial.begin(9600);

esp.begin(115200);

sensors.begin();

numberOfDevices = sensors.getDeviceCount();

pulseSensor.analogInput(PulseWire);

pulseSensor.setThreshold(Threshold);

pulseSensor.begin();

pinMode(buzzer, OUTPUT);

digitalWrite(buzzer, HIGH);

lcd.setCursor(0, 0);

lcd.print(" IoT Patient");

lcd.setCursor(0, 1);

lcd.print(" Monitor System");

delay(1500);

digitalWrite(buzzer, LOW);

lcd.clear();

}

void loop()

{

myBPM = pulseSensor.getBeatsPerMinute();

if (pulseSensor.sawStartOfBeat())

{

beep();

lcd.setCursor(0, 1);

lcd.print("HEART:");

lcd.print(myBPM);

lcd.setCursor(9, 1);

lcd.print(" BPM");

delay(20);

}

sensors.requestTemperatures();

for (int i = 0; i < numberOfDevices; i++)

{

if (sensors.getAddress(tempDeviceAddress, i))

{

temp = printTemperature(tempDeviceAddress);

lcd.setCursor(0, 0);

lcd.print("BODY:");

lcd.print(temp);

lcd.print(" \*C");

}

}

upload();

}

int printTemperature(DeviceAddress deviceAddress)

{

int tempC = sensors.getTempC(deviceAddress);

return tempC;

}

void beep()

{

digitalWrite(buzzer, HIGH);

delay(150);

digitalWrite(buzzer, LOW);

}

void upload()

{

unsigned long currentMillis = millis();

if (currentMillis - previousMillis >= interval)

{

previousMillis = currentMillis;

esp.print('\*');

esp.print(myBPM);

esp.print(temp);

esp.println('#');

}

}