

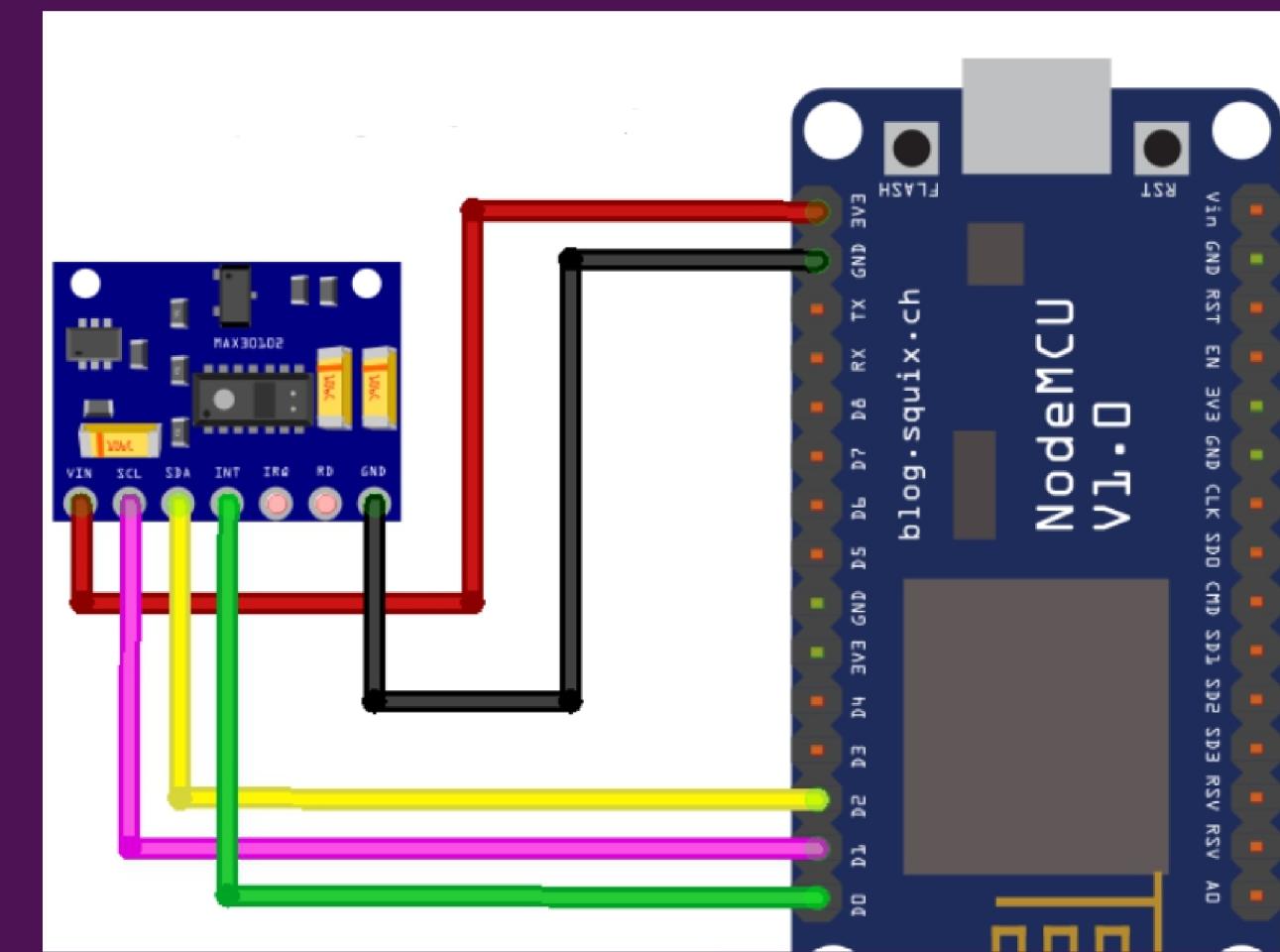
IOT BASED PATIENT HEALTH MONITORING SYSTEM

overview

As IOT is rapidly growing in healthcare industry.keeping track of patient health is becoming very difficult because of busy schedule.specially old age patient should be monitored periodically.so we are proposing a innovative system that can be wearable and can track the patient health in cloud with ease. our device can track parameters like Blood oxygen level along with body temperature. we are using MAX 30102 Pulse Oximeter sensor to measure the Heart Rate/Pulse(BPM) as well as Blood oxygen level(spo2).To get the data in cloud,we are integrating with Thinkspeak which needs only internet connection to Get the Data

Bill of Materials:

ESP 8266(Node Mcu) BOARD
Pulse Oximeter Sensor -MAX 30102
Connecting Wires

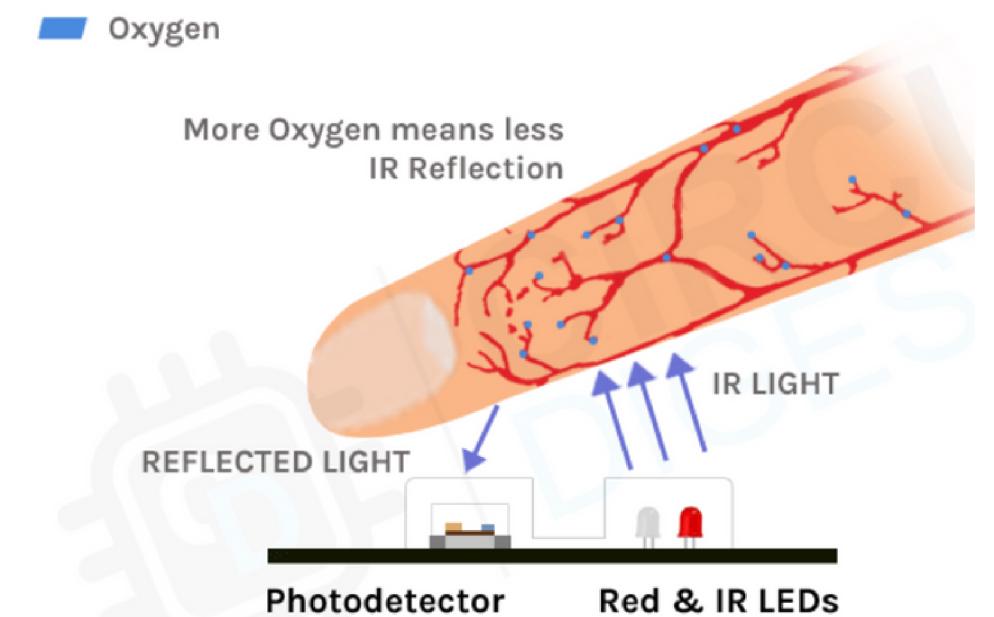


WORKING:

The MAX30102 Pulse Oximeter has two LEDs, in which one emits red light and the other emits infrared light. For pulse rate, only infrared light is required. But, both red light and infrared light are used to measure oxygen levels in the blood. When the heart pumps blood, there is more blood, which results in increased oxygen in the blood. When the heart rests, the amount of oxygenated blood also decreases. The pulse rate is determined by knowing the time between the rise and fall of oxygenated blood.

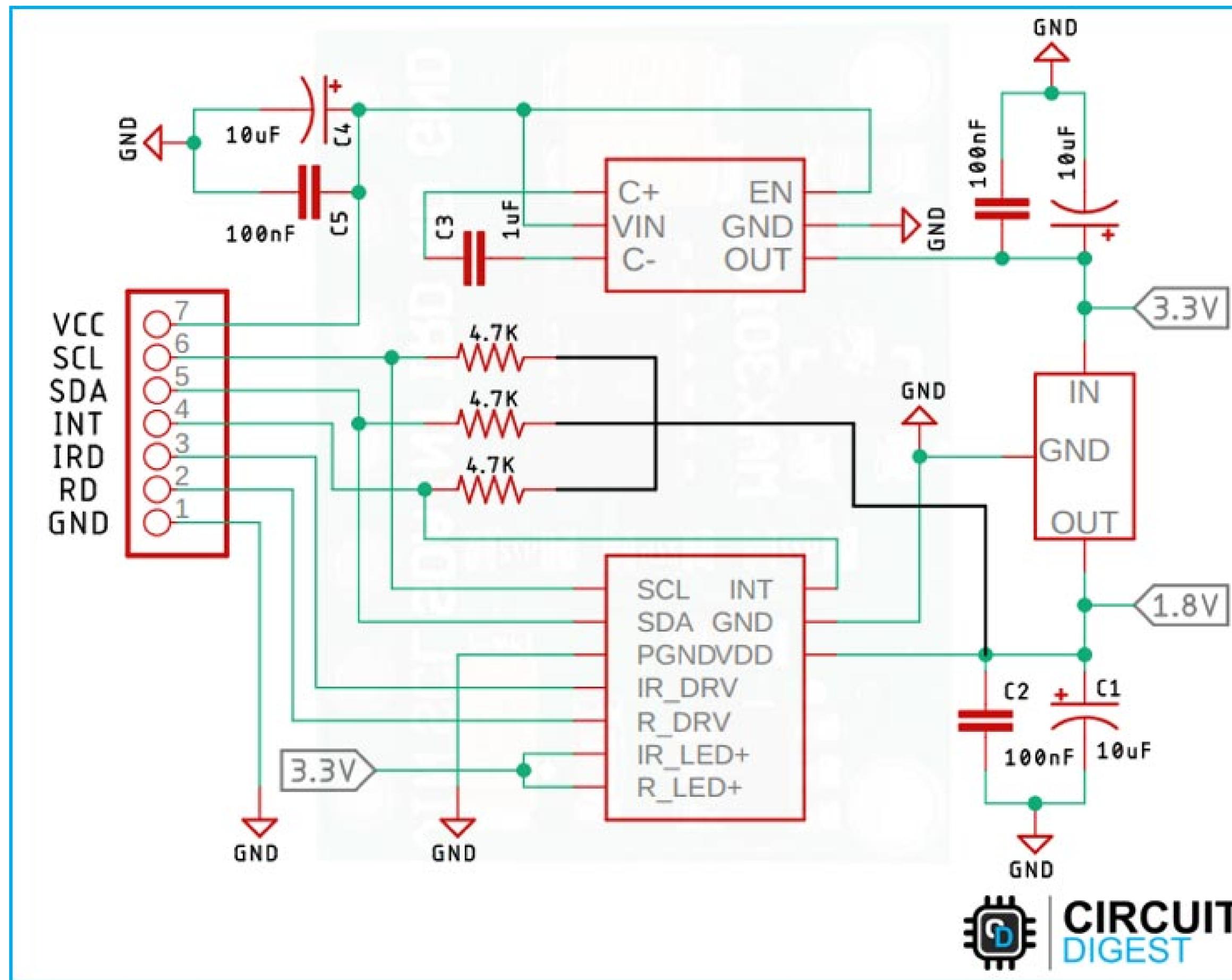
WORKING:

The MAX30102 Sensor shines both the light through the skin and measures the reflection with the photodetector. This method of pulse detection through light is called **Photoplethysmogram**



A study conducted by IJEECS suggests that the accuracy rate of the MAX30102 measurement is 97.11% and 98.84%, for heart rate and oxygen saturation (SpO_2), respectively.

CIRCUIT DIAGRAM FOR MAX30102 PULSE OXIMETER



CODE:

```
#include <Wire.h>
#include "MAX30105.h"
#include "heartRate.h"

#include <ESP8266WiFi.h>;
#include <WiFiClient.h>;
#include <ThingSpeak.h>

MAX30105 particleSensor;

const char* ssid = "Abi's Nord"; //Your Network SSID
const char* password = "abi@123456"; //Your Network Password
WiFiClient client;
unsigned long myChannelNumber = 1951295; //Your Channel Number (Without Brackets)
const char *myWriteAPIKey = "VJ8YE09Y4PPASTFO"; //Your Write API Key

const byte RATE_SIZE = 4; //Increase this for more averaging. 4 is good.
byte rates[RATE_SIZE]; //Array of heart rates
byte rateSpot = 0;
long lastBeat = 0; //Time at which the last beat occurred

float beatsPerMinute;
int beatAvg;

int period = 20000;
unsigned long time_now = 0;

void setup()
{
    Serial.begin(115200);
    Serial.println("Initializing...");

    // Initialize sensor
    if (!particleSensor.begin(Wire, I2C_SPEED_FAST)) //Use default I2C port, 400kHz speed
    {
        Serial.println("MAX30105 was not found. Please check wiring/power. ");
        while (1);
    }
    Serial.println("Place your index finger on the sensor with steady pressure.");

    particleSensor.setup(); //Configure sensor with default settings
    particleSensor.setPulseAmplitudeRed(0x0A); //Turn Red LED to low to indicate sensor is running
    particleSensor.setPulseAmplitudeGreen(0); //Turn off Green LED
    particleSensor.enableDIETEMPRDY();

    WiFi.begin(ssid, password); //connect to wifi network
    ThingSpeak.begin(client);
}
```

CODE:

```
void loop()
{
    long irValue = particleSensor.getIR();

    if (checkForBeat(irValue) == true)
    {
        //We sensed a beat!
        long delta = millis() - lastBeat;
        lastBeat = millis();

        beatsPerMinute = 60 / (delta / 1000.0);

        if (beatsPerMinute < 255 && beatsPerMinute > 20)
        {
            rates[rateSpot++] = (byte)beatsPerMinute; //Store this reading in the array
            rateSpot %= RATE_SIZE; //Wrap variable

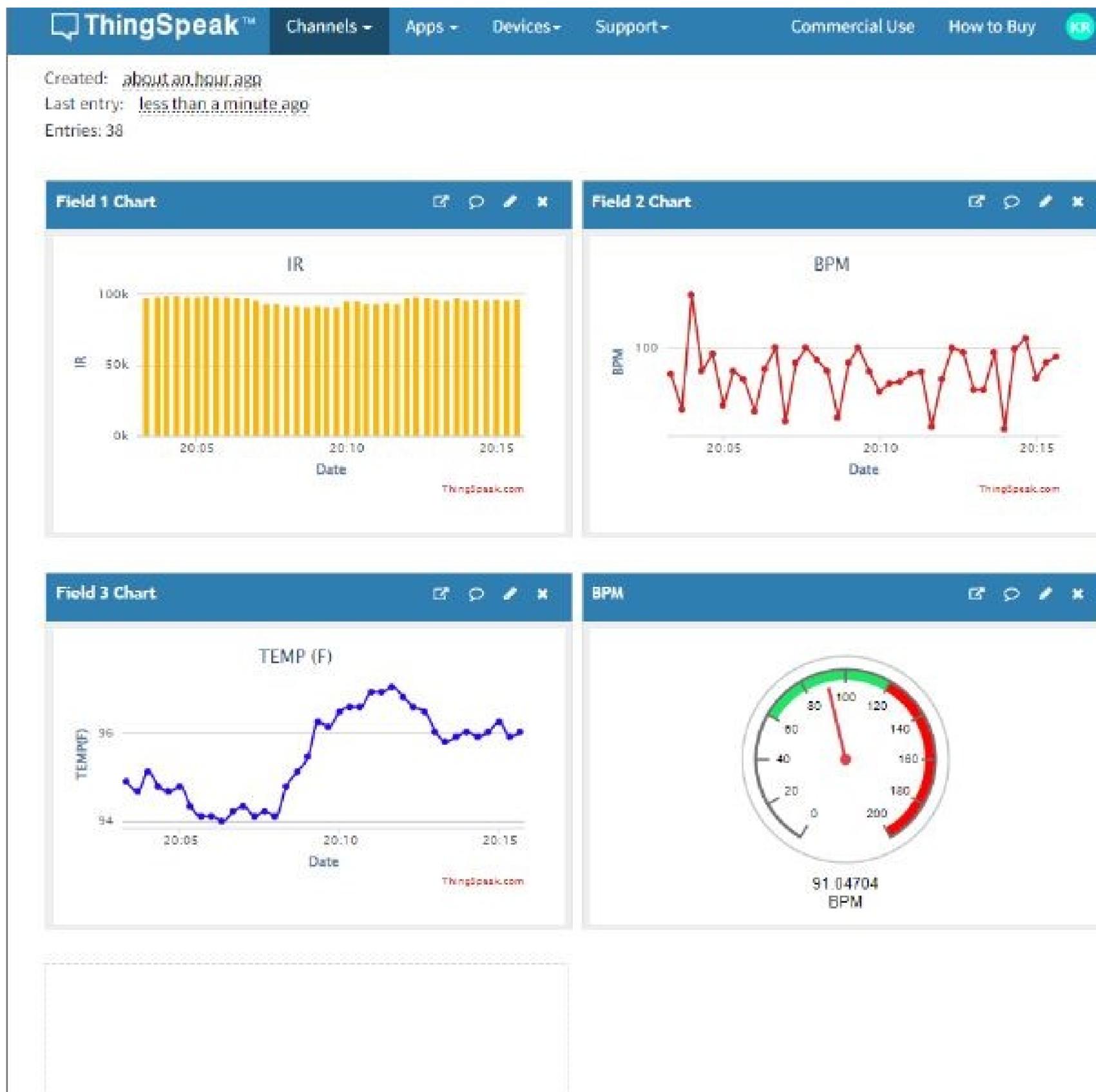
            //Take average of readings
            beatAvg = 0;
            for (byte x = 0 ; x < RATE_SIZE ; x++)
                beatAvg += rates[x];
            beatAvg /= RATE_SIZE;
        }
    }

    float temperatureF = particleSensor.readTemperatureF();

    Serial.print("IR=");
    Serial.print(irValue);
    Serial.print(", BPM=");
    Serial.print(beatsPerMinute);
    Serial.print(", Avg BPM=");
    Serial.print(beatAvg);
    Serial.print(" temperatureF=");
    Serial.print(temperatureF, 2);
    if (irValue < 50000)
        Serial.print(" No finger?");

    Serial.println();
    if(millis() >= time_now + period){
        time_now += period;
        ThingSpeak.setField(1,irValue);
        ThingSpeak.setField(2,beatsPerMinute);
        ThingSpeak.setField(3,temperatureF);
        ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);
    }
}
```

OUTPUT:



Example5_HeartRate § License

```
WiFi.begin(ssid, password); //connect to wifi network
ThingSpeak.begin(client);
}

void loop()
{
    long irValue = particleSensor.getIR();

    if (checkForBeat(irValue) == true)
    {
        //We sensed a beat!
        long delta = millis() - lastBeat;
        lastBeat = millis();

        beatsPerMinute = 60 * COMS
    }

    if (beatsPerMinute
    {
        rates[rateSpot++]
        rateSpot %= RATE_
        IR=97365, BPM=99.50, Avg BPM=91 temperatureF=95.90
        IR=97576, BPM=99.50, Avg BPM=91 temperatureF=95.90
        IR=97659, BPM=99.50, Avg BPM=91 temperatureF=95.68
        IR=97741, BPM=99.50, Avg BPM=91 temperatureF=95.90
        IR=97850, BPM=99.50, Avg BPM=91 temperatureF=95.90
        IR=97824, BPM=99.50, Avg BPM=91 temperatureF=95.79
        IR=97727, BPM=99.50, Avg BPM=91 temperatureF=95.90
        IR=96802, BPM=99.50, Avg BPM=91 temperatureF=95.68
        IR=96674, BPM=95.09, Avg BPM=96 temperatureF=95.79
        IR=96673, BPM=95.09, Avg BPM=96 temperatureF=95.68
        IR=96895, BPM=95.09, Avg BPM=96 temperatureF=95.56
        IR=96980, BPM=95.09, Avg BPM=96 temperatureF=96.01
    }
}
```

Done uploading

Leaving... Hard resetting via RTS pin...

Autoscroll Show timestamp Newline 115200 baud Clear output

References

<https://datasheets.maximintegrated.com/en/ds/MAX30102.pdf>