

**Video Link -** [https://drive.google.com/file/d/1smJLNS9XI7ca520TA-TeXJGtV7Y\\_JEQu/view?usp=sharing](https://drive.google.com/file/d/1smJLNS9XI7ca520TA-TeXJGtV7Y_JEQu/view?usp=sharing)

## MAX30100 with thingspeak

```
#include <ThingSpeak.h>
#include <Wire.h>
#include "MAX30100_PulseOximeter.h"
#define BLYNK_PRINT Serial
#include <Blynk.h>
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

#include "Wire.h"
#include "Adafruit_GFX.h"
#include "OakOLED.h"

OakOLED oled;

WiFiClient client;

long myChannelNumber = 1578852;
const char myWriteAPIKey[] = "XJZ5M3UQC8NM2NVN";

char auth[] = "puf-_b9vZ7axCa4QBSShtXE4gJilzKTy"; // You should get Auth Token in
the Blynk App.

char ssid[] = "TP-Link_A30C"; // Your WiFi credentials.
char pass[] = "23779266";

// Connections : SCL PIN - D1 , SDA PIN - D2 , INT PIN - D0
PulseOximeter pox;
```

```
float BPM, SpO2;
```

```
uint32_t tsLastReport = 0;
```

```
String myStatus = "";
```

```
const unsigned char bitmap [] PROGMEM=
```

```
{
```

```
0x00, 0x00, 0x00, 0x00, 0x01, 0x80, 0x18, 0x00, 0x0f, 0xe0, 0x7f, 0x00, 0x3f, 0xf9, 0xff, 0xc0,  
0x7f, 0xf9, 0xff, 0xc0, 0x7f, 0xff, 0xff, 0xe0, 0x7f, 0xff, 0xff, 0xe0, 0xff, 0xff, 0xff, 0xf0,  
0xff, 0xf7, 0xff, 0xf0, 0xff, 0xe7, 0xff, 0xf0, 0xff, 0xe7, 0xff, 0xf0, 0x7f, 0xdb, 0xff, 0xe0,  
0x7f, 0x9b, 0xff, 0xe0, 0x00, 0x3b, 0xc0, 0x00, 0x3f, 0xf9, 0x9f, 0xc0, 0x3f, 0xfd, 0xbf, 0xc0,  
0x1f, 0xfd, 0xbf, 0x80, 0x0f, 0xfd, 0x7f, 0x00, 0x07, 0xfe, 0x7e, 0x00, 0x03, 0xfe, 0xfc, 0x00,  
0x01, 0xff, 0xf8, 0x00, 0x00, 0xff, 0xf0, 0x00, 0x00, 0x7f, 0xe0, 0x00, 0x00, 0x3f, 0xc0, 0x00,  
0x00, 0x0f, 0x00, 0x00, 0x00, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
```

```
};
```

```
void onBeatDetected()
```

```
{
```

```
    Serial.println("Beat Detected!");  
    oled.drawBitmap( 60, 20, bitmap, 28, 28, 1);  
    oled.display();
```

```
}
```

```
void setup()
```

```
{
```

```
    Serial.begin(115200);
```

```
    ThingSpeak.begin(client);
```

```
oled.begin();  
oled.clearDisplay();  
oled.setTextSize(1);  
oled.setTextColor(1);  
oled.setCursor(0, 0);  
oled.println("Initializing pulse oximeter..");  
oled.display();
```

```
pinMode(16, OUTPUT);  
Blynk.begin(auth, ssid, pass);
```

```
Serial.print("Initializing Pulse Oximeter..");
```

```
if (!pox.begin())  
{  
    Serial.println("FAILED");  
    oled.clearDisplay();  
    oled.setTextSize(1);  
    oled.setTextColor(1);  
    oled.setCursor(0, 0);  
    oled.println("FAILED");  
    oled.display();  
    for(;;);  
}  
else  
{  
    oled.clearDisplay();  
    oled.setTextSize(1);  
    oled.setTextColor(1);
```

```
oled.setCursor(0, 0);  
oled.println("SUCCESS");  
oled.display();  
Serial.println("SUCCESS");  
pox.setOnBeatDetectedCallback(onBeatDetected);  
}
```

// The default current for the IR LED is 50mA and it could be changed by uncommenting the following line.

```
pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);  
  
}
```

```
void loop()
```

```
{  
  pox.update();  
  Blynk.run();
```

```
  BPM = pox.getHeartRate();
```

```
  SpO2 = pox.getSpO2();
```

```
  Serial.print(" ❤️ Heart rate:");
```

```
  Serial.print(BPM);
```

```
  Serial.print(" bpm / SpO2:");
```

```
  Serial.print(SpO2);
```

```
  Serial.println(" %");
```

```
  Blynk.virtualWrite(V7, BPM);
```

```
  Blynk.virtualWrite(V8, SpO2);
```

```
ThingSpeak.setField(2, BPM);
```

```
ThingSpeak.setField(3, SpO2);
```

```
ThingSpeak.setStatus(myStatus);
```

```
int x = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);
```

```
if(x == 200){
```

```
    Serial.println("Channel update successful.");
```

```
}
```

```
else{
```

```
    Serial.println("Problem updating channel. HTTP error code " + String(x));
```

```
}
```

```
oled.clearDisplay();
```

```
oled.setTextSize(1);
```

```
oled.setTextColor(1);
```

```
oled.setCursor(0,16);
```

```
oled.println(pox.getHeartRate());
```

```
oled.setTextSize(1);
```

```
oled.setTextColor(1);
```

```
oled.setCursor(0, 0);
```

```
oled.println("Heart BPM");
```

```
oled.setTextSize(1);
```

```
oled.setTextColor(1);
```

```
oled.setCursor(0, 30);
```

```
oled.println("Spo2");
```

```
oled.setTextSize(1);  
oled.setTextColor(1);  
oled.setCursor(0,45);  
oled.println(pox.getSpO2());  
oled.display();  
  
delay(10);  
}
```

## MAX30100 without thingspeak

```
//#include <ThingSpeak.h>  
  
#include <Wire.h>  
#include "MAX30100_PulseOximeter.h"  
#define BLYNK_PRINT Serial  
#include <Blynk.h>  
#include <ESP8266WiFi.h>  
#include <BlynkSimpleEsp8266.h>  
  
#include "Wire.h"  
#include "Adafruit_GFX.h"  
#include "OakOLED.h"  
  
#define REPORTING_PERIOD_MS 1000  
  
OakOLED oled;  
  
//WiFiClient client;
```

[illegible]

```
Serial.println("Beat Detected!");  
oled.drawBitmap( 60, 20, bitmap, 28, 28, 1);  
oled.display();  
}
```

```
void setup()  
{  
  Serial.begin(115200);
```

```
//  ThingSpeak.begin(client);
```

```
  oled.begin();  
  oled.clearDisplay();  
  oled.setTextSize(1);  
  oled.setTextColor(1);  
  oled.setCursor(0, 0);
```

```
  oled.println("Initializing pulse oximeter..");  
  oled.display();
```

```
  pinMode(16, OUTPUT);  
  Blynk.begin(auth, ssid, pass);
```

```
  Serial.print("Initializing Pulse Oximeter..");
```

```
  if (!pox.begin())  
  {  
    Serial.println("FAILED");
```



```

    oled.clearDisplay();
    oled.setTextSize(1);
    oled.setTextColor(1);
    oled.setCursor(0, 0);
    oled.println("FAILED");
    oled.display();

    for(;;);
}
else
{
    oled.clearDisplay();
    oled.setTextSize(1);
    oled.setTextColor(1);
    oled.setCursor(0, 0);
    oled.println("SUCCESS");
    oled.display();
    Serial.println("SUCCESS");
    pox.setOnBeatDetectedCallback(onBeatDetected);
}

```

// The default current for the IR LED is 50mA and it could be changed by uncommenting the following line.

```

    pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);

}

```

```

void loop()
{
    pox.update();
    Blynk.run();
}

```

```
BPM = pox.getHeartRate();
SpO2 = pox.getSpO2();

if (millis() - tsLastReport > REPORTING_PERIOD_MS)
{
    Serial.print(" ❤️ Heart rate:");
    Serial.print(BPM);
    Serial.print(" bpm / SpO2:");
    Serial.print(SpO2);
    Serial.println(" %");

    Blynk.virtualWrite(V7, BPM);
    Blynk.virtualWrite(V8, SpO2);

    // ThingSpeak.writeField(myChannelNumber, 2, pox.getHeartRate(), myWriteAPIKey);
    // ThingSpeak.writeField(myChannelNumber, 3, pox.getSpO2(), myWriteAPIKey);

    oled.clearDisplay();
    oled.setTextSize(1);
    oled.setTextColor(1);
    oled.setCursor(0,16);
    oled.println(pox.getHeartRate());

    oled.setTextSize(1);
    oled.setTextColor(1);
    oled.setCursor(0, 0);
    oled.println("Heart BPM");
```

```

oled.setTextSize(1);
oled.setTextColor(1);
oled.setCursor(0, 30);
oled.println("Spo2");

oled.setTextSize(1);
oled.setTextColor(1);
oled.setCursor(0,45);
oled.println(pox.getSpO2());
oled.display();

// set the status
//   ThingSpeak.setStatus(myStatus);

//   // write to the ThingSpeak channel
//   int x = ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);
//   if(x == 200){
//       Serial.println("Channel update successful.");
//   }
//   else{
//       Serial.println("Problem updating channel. HTTP error code " + String(x));
//   }

    tsLastReport = millis();
}
// delay(10);
}

```

# MLX90614 code

```
#include <ThingSpeak.h>

#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>

//char auth[] = "SfE1100B-cC4FzhWg8Ef93HSE8le2WEH";
//char ssid[] = "Airtel_7042615199";
//char pass[] = "air46552";

WiFiClient client;

long myChannelNumber = 1578852;
const char myWriteAPIKey[] = "XJZ5M3UQC8NM2NVN";

char auth[] = "puf-_b9vZ7axCa4QBSShtXE4gJilzKTy";
char ssid[] = "TP-Link_A30C";
char pass[] = "23779266";

#include <Wire.h>
#include <Adafruit_MLX90614.h>
Adafruit_MLX90614 mlx = Adafruit_MLX90614();

float ambient_temp = 0.0;
float object_temp = 0.0;

BlynkTimer timer;
```

```

void sensor1(){
    ambient_temp = mlx.readAmbientTempC();
    Serial.print("Ambient Temp in °C = ");
    Serial.println(ambient_temp);
    Blynk.virtualWrite(V4,ambient_temp);
    // ThingSpeak.writeField(myChannelNumber, 4, ambient_temp, myWriteAPIKey);

    object_temp = mlx.readObjectTempC();
    Serial.print("Object Temp in °C = ");
    Serial.println(object_temp);
    Blynk.virtualWrite(V5,object_temp);
    ThingSpeak.writeField(myChannelNumber, 1, object_temp, myWriteAPIKey);
}

void setup() {
    Serial.begin(9600);
    Serial.println("Adafruit MLX90614 test");
    Blynk.begin(auth, ssid, pass, "blynk-cloud.com", 80);
    mlx.begin();
    ThingSpeak.begin(client);
    timer.setInterval(1000L,sensor1);
}

void loop() {
    Blynk.run();
    timer.run();
}

```