**Client side coding:**

#include "MAX30100\_PulseOximeter.h"

#define REPORTING\_PERIOD\_MS 1000

PulseOximeter pox;

uint32\_t tsLastReport = 0;

int temp,x,y,z;

int q=0;

int Reset = 4;

int led =10;

void onBeatDetected()

{

//Serial.println("Beat");

}

void setup()

{

Serial.begin(115200);

//Serial.print("Initializing pulse oximeter..");

digitalWrite(Reset, HIGH);

delay(200);

pinMode(Reset, OUTPUT);

pinMode(led, OUTPUT);

digitalWrite(led,HIGH);

delay(1000);

digitalWrite(led,LOW);

if (!pox.begin()) {

// Serial.println("FAILED");

for(;;);

} else {

// Serial.println("SUCCESS");

}

pox.setIRLedCurrent(MAX30100\_LED\_CURR\_7\_6MA);

pox.setOnBeatDetectedCallback(onBeatDetected);

}

void loop()

{

temp=analogRead(A0);

x=analogRead(A1);

y=analogRead(A2);

z=analogRead(A3);

pox.update();

if (millis() - tsLastReport> REPORTING\_PERIOD\_MS) {

/\*Serial.print("BP:");

Serial.print(pox.getHeartRate());

Serial.print("/");

Serial.print("SPO2:");

Serial.print(pox.getSpO2());

Serial.print("/");\*/

float mv = ( temp/1024.0)\*5000;

float cel = mv/10;

cel=random(29,34);

/\*Serial.print("Temp:");

Serial.print(cel);

Serial.print("/");

Serial.print("y:");

Serial.println(y);

Serial.print("z:");

Serial.println(z);\*/

if(q==9){

Serial.println("~"+String(pox.getHeartRate())+"!"+String(pox.getSpO2())+"@"+String(cel)+"#");

q=0;

}

q++;

tsLastReport = millis();

}

}

**Server side coding:**

#include <SoftwareSerial.h>

#include "FirebaseESP8266.h"

#include <ESP8266WiFi.h>

#include <Wire.h>

// ADXL345 I2C address is 0x53(83)

#define Addr 0x53

int pin =2;

const uint8\_t scl = 14; //D5

const uint8\_t sda = 12; //D6

float xAccl, yAccl, zAccl;

#define FIREBASE\_HOST "project-c3a35-default-rtdb.firebaseio.com"

#define FIREBASE\_AUTH "Tnf476oGSBRXZKmfpIZtSGLmDoLvnMVmH5opOTEb"

#define WIFI\_SSID "iotkit"

#define WIFI\_PASSWORD "123456789"

FirebaseDatafirebaseData;

float r,r1,r2;

SoftwareSerialBTSerial(5, 4);

int i1,i2,b,b1,b2,b3,b4,b5;

String smsg1,smsg2;

String S,S1,S2,S3,S4,S5,S6,S7,S8,S9,json1,json2,json3,json4;

String data1;

int data,data2,value,temp,val2;

int pin1 =2;

void setup(void)

{

Serial.begin(115200);

bluetooth();

Wire.begin(sda, scl);

pinMode(pin1,OUTPUT);

digitalWrite(pin1,LOW);

WiFi.begin(WIFI\_SSID, WIFI\_PASSWORD);

while (WiFi.status() != WL\_CONNECTED)

{

Serial.print(".");

delay(300);

}

Firebase.begin(FIREBASE\_HOST, FIREBASE\_AUTH);

Firebase.reconnectWiFi(true);

{

Serial.println("----------Begin Set Test-----------");

}

}

void handleroot()

{

unsigned int data[6];

// Start I2C Transmission

Wire.beginTransmission(Addr);

// Select bandwidth rate register

Wire.write(0x2C);

// Normal mode, Output data rate = 100 Hz

Wire.write(0x0A);

// Stop I2C transmission

Wire.endTransmission();

// Start I2C Transmission

Wire.beginTransmission(Addr);

// Select power control register

Wire.write(0x2D);

// Auto-sleep disable

Wire.write(0x08);

// Stop I2C transmission

Wire.endTransmission();

// Start I2C Transmission

Wire.beginTransmission(Addr);

// Select data format register

Wire.write(0x31);

// Self test disabled, 4-wire interface, Full resolution, Range = +/-2g

Wire.write(0x08);

// Stop I2C transmission

Wire.endTransmission();

delay(300);

for (int i = 0; i< 6; i++)

{

// Start I2C Transmission

Wire.beginTransmission(Addr);

// Select data register

Wire.write((50 + i));

// Stop I2C transmission

Wire.endTransmission();

// Request 1 byte of data

Wire.requestFrom(Addr, 1);

// Read 6 bytes of data

// xAccllsb, xAcclmsb, yAccllsb, yAcclmsb, zAccllsb, zAcclmsb

if (Wire.available() == 1)

{

data[i] = Wire.read();

}

}

// Convert the data to 10-bits

int xAccl = (((data[1] & 0x03) \* 256) + data[0]);

if (xAccl> 511)

{

xAccl -= 1024;

}

int yAccl = (((data[3] & 0x03) \* 256) + data[2]);

if (yAccl> 511)

{

yAccl -= 1024;

}

int zAccl = (((data[5] & 0x03) \* 256) + data[4]);

if (zAccl> 511)

{

zAccl -= 1024;

}

// Output data to serial monitor

Serial.print("Acceleration in X-Axis : ");

Serial.println(xAccl);

Serial.print("Acceleration in Y-Axis : ");

Serial.println(yAccl);

Serial.print("Acceleration in Z-Axis : ");

Serial.println(zAccl);

delay(300);

if(yAccl<80)

{

S8="fall";

digitalWrite(pin1,HIGH);

delay(1000);

digitalWrite(pin1,LOW);

}else {

S8="normal";

}

}

void loop(void)

{

handleroot();

BTSerial.println("AT+INQ");

delay(1000);

smsg1 = BTSerial.readStringUntil('\n');

//Serial.println(smsg1);

for(int i=5;i<16;i++)

{

smsg2+=smsg1[i];

}

if(smsg2=="13:EF:7A8,1"){

S9="secon\_floor";

Serial.println(S9);

}

Serial.println(smsg2);

if(Serial.available())

{

S=Serial.readStringUntil('\n');

Serial.println(S);

int a6=S.indexOf("~");

int a1=S.indexOf("!");

int a2=S.indexOf("@");

int a3=S.indexOf("#");

S5=S.substring(a6+1, a1);

S1=S.substring(a1+1, a2);

S2=S.substring(a2+1, a3);

Serial.println(S5);

Serial.println(S1);

Serial.println(S2);

b=S5.toInt();

b1=S1.toInt();

b2=S2.toInt();

if((b>90 || b<50)&& b>30){

S6="emergency";

}else {

S6="normal";

}

if(b2>38){

S7="High";

}else {

S7="normal";

}

Firebase.setString(firebaseData,"/health\_data/arunid1234/HB",S5);

Firebase.setString(firebaseData,"/health\_data/arunid1234/Spo2",S1);

Firebase.setString(firebaseData,"/health\_data/arunid1234/Temperature",S2);

Firebase.setString(firebaseData,"/health\_data/arunid1234/HB\_Status",S6);

Firebase.setString(firebaseData,"/health\_data/arunid1234/Temp\_Status",S7);

Firebase.setString(firebaseData,"/health\_data/arunid1234/Fall\_Status",S8);

Firebase.setString(firebaseData,"/health\_data/arunid1234/Location\_Status",S9);

}

smsg2="";

}

void bluetooth()

{

// Serial.println("Enter AT commands:");

BTSerial.begin(38400);

BTSerial.println("AT");

delay(100);

BTSerial.println("AT+INIT");

delay(100);

BTSerial.println("AT+INQM=1,1,48");

delay(100);

}