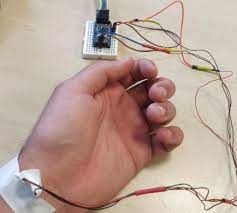
max30102

Diagram

Description automatically generated

measuring the oxygen concentration in the blood (SpO2 percentage)

see [2] to know how to add it in ur arduino

Reading Temperature Code:

#include <Wire.h>

#include "MAX30105.h"

MAX30105 particleSensor;

void setup() {

Serial.begin(9600);

Serial.println("Initializing...");

// Initialize sensor

if (particleSensor.begin(Wire, I2C\_SPEED\_FAST) == false) { //Use default I2C port, 400kHz speed

Serial.println("MAX30102 was not found. Please check wiring/power. ");

while (1);

}

//The LEDs are very low power and won't affect the temp reading much but

//you may want to turn off the LEDs to avoid any local heating

particleSensor.setup(0); //Configure sensor. Turn off LEDs

particleSensor.enableDIETEMPRDY(); //Enable the temp ready interrupt. This is required.

}

void loop() {

float temperature = particleSensor.readTemperature();

Serial.print("temperatureC=");

Serial.print(temperature, 4);

float temperatureF = particleSensor.readTemperatureF();

Serial.print(" temperatureF=");

Serial.print(temperatureF, 4);

Serial.println();

}

Measuring Heart-Rate (BPM) Code:

#include <Wire.h>

#include "MAX30105.h"

#include "heartRate.h"

MAX30105 particleSensor;

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;

void setup() {

Serial.begin(115200);

Serial.println("Initializing...");

// Initialize sensor

if (!particleSensor.begin(Wire, I2C\_SPEED\_FAST)) {

Serial.println("MAX30102 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

}

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;

}

}

Serial.print("IR=");

Serial.print(irValue);

Serial.print(", BPM=");

Serial.print(beatsPerMinute);

Serial.print(", Avg BPM=");

Serial.print(beatAvg);

if (irValue < 50000)

Serial.print(" No finger?");

Serial.println();

}

Measuring Oxygen Saturation (SpO2) code:

#include <Wire.h>

#include "MAX30105.h"

#include "spo2\_algorithm.h"

MAX30105 particleSensor;

#define MAX\_BRIGHTNESS 255

#if defined(\_\_AVR\_ATmega328P\_\_) || defined(\_\_AVR\_ATmega168\_\_)

//Arduino Uno doesn't have enough SRAM to store 100 samples of IR led data and red led data in 32-bit format

//To solve this problem, 16-bit MSB of the sampled data will be truncated. Samples become 16-bit data.

uint16\_t irBuffer[100]; //infrared LED sensor data

uint16\_t redBuffer[100]; //red LED sensor data

#else

uint32\_t irBuffer[100]; //infrared LED sensor data

uint32\_t redBuffer[100]; //red LED sensor data

#endif

int32\_t bufferLength; //data length

int32\_t spo2; //SPO2 value

int8\_t validSPO2; //indicator to show if the SPO2 calculation is valid

int32\_t heartRate; //heart rate value

int8\_t validHeartRate; //indicator to show if the heart rate calculation is valid

byte pulseLED = 11; //Must be on PWM pin

byte readLED = 13; //Blinks with each data read

void setup()

{

Serial.begin(115200); // initialize serial communication at 115200 bits per second:

pinMode(pulseLED, OUTPUT);

pinMode(readLED, OUTPUT);

// Initialize sensor

if (!particleSensor.begin(Wire, I2C\_SPEED\_FAST)) //Use default I2C port, 400kHz speed

{

Serial.println(F("MAX30105 was not found. Please check wiring/power."));

while (1);

}

Serial.println(F("Attach sensor to finger with rubber band. Press any key to start conversion"));

while (Serial.available() == 0) ; //wait until user presses a key

Serial.read();

byte ledBrightness = 60; //Options: 0=Off to 255=50mA

byte sampleAverage = 4; //Options: 1, 2, 4, 8, 16, 32

byte ledMode = 2; //Options: 1 = Red only, 2 = Red + IR, 3 = Red + IR + Green

byte sampleRate = 100; //Options: 50, 100, 200, 400, 800, 1000, 1600, 3200

int pulseWidth = 411; //Options: 69, 118, 215, 411

int adcRange = 4096; //Options: 2048, 4096, 8192, 16384

particleSensor.setup(ledBrightness, sampleAverage, ledMode, sampleRate, pulseWidth, adcRange); //Configure sensor with these settings

}

void loop()

{

bufferLength = 100; //buffer length of 100 stores 4 seconds of samples running at 25sps

//read the first 100 samples, and determine the signal range

for (byte i = 0 ; i < bufferLength ; i++)

{

while (particleSensor.available() == false) //do we have new data?

particleSensor.check(); //Check the sensor for new data

redBuffer[i] = particleSensor.getRed();

irBuffer[i] = particleSensor.getIR();

particleSensor.nextSample(); //We're finished with this sample so move to next sample

Serial.print(F("red="));

Serial.print(redBuffer[i], DEC);

Serial.print(F(", ir="));

Serial.println(irBuffer[i], DEC);

}

//calculate heart rate and SpO2 after first 100 samples (first 4 seconds of samples)

maxim\_heart\_rate\_and\_oxygen\_saturation(irBuffer, bufferLength, redBuffer, &spo2, &validSPO2, &heartRate, &validHeartRate);

//Continuously taking samples from MAX30102. Heart rate and SpO2 are calculated every 1 second

while (1)

{

//dumping the first 25 sets of samples in the memory and shift the last 75 sets of samples to the top

for (byte i = 25; i < 100; i++)

{

redBuffer[i - 25] = redBuffer[i];

irBuffer[i - 25] = irBuffer[i];

}

//take 25 sets of samples before calculating the heart rate.

for (byte i = 75; i < 100; i++)

{

while (particleSensor.available() == false) //do we have new data?

particleSensor.check(); //Check the sensor for new data

digitalWrite(readLED, !digitalRead(readLED)); //Blink onboard LED with every data read

redBuffer[i] = particleSensor.getRed();

irBuffer[i] = particleSensor.getIR();

particleSensor.nextSample(); //We're finished with this sample so move to next sample

//send samples and calculation result to terminal program through UART

Serial.print(F("red="));

Serial.print(redBuffer[i], DEC);

Serial.print(F(", ir="));

Serial.print(irBuffer[i], DEC);

Serial.print(F(", HR="));

Serial.print(heartRate, DEC);

Serial.print(F(", HRvalid="));

Serial.print(validHeartRate, DEC);

Serial.print(F(", SPO2="));

Serial.print(spo2, DEC);

Serial.print(F(", SPO2Valid="));

Serial.println(validSPO2, DEC);

}

//After gathering 25 new samples recalculate HR and SP02

maxim\_heart\_rate\_and\_oxygen\_saturation(irBuffer, bufferLength, redBuffer, &spo2, &validSPO2, &heartRate, &validHeartRate);

}

}

Datasheet

<https://cdn.shopify.com/s/files/1/0672/9409/files/MAX30102_datasheet.pdf?v=1598906487>

Ref

[1]<https://store.fut-electronics.com/products/pulse-oximeter-spo2-heart-rate-sensor-max30100#:~:text=The%20MAX30102%20can%20be%20used,the%20infrared%20light%20is%20needed>.

[2]<https://lastminuteengineers.com/max30102-pulse-oximeter-heart-rate-sensor-arduino-tutorial/>