UNIVERSITY of Karachi

PULSE OXIMETER USING ARDUINO

GROUP MEMBERS:

- M. Faizan Chawla (EB20102074)
- Osama Ali (EB20102106)
- Zaid Khan (EB20102148)

COURSE NAME:

System Design with Microprocessor (402)

Submitted to:

Mr. Abdul Bari

ABSTRACT:

Pulse Oximeter is a noninvasive medical device used for measuring the oxygen saturation and pulse rate of a patient. The system is also smaller, energy efficient, user and cost friendly. This project involved the process to design portable equipment that can be used to measure the oxygen percentage in blood and also to detect the heart rate of humans by using the MAX30100 Sensor with Arduino and displaying it on the LCD display.

Table of Contents

INTRODUCTION:

In this project we are going to make the **PULSE OXIMETER** using arduino. The MAX30100 Sensor is capable of measuring **Blood Oxygen** & **Heart Rate**. We can use any display like a **16x2 LCD Display** to view the value of **SpO2** and **BPM**. The blood Oxygen Concentration termed SpO2 is measured in Percentage and **Heart Beat/Pulse Rate is measured in BPM.** With each beat, the display value is changed in the screen

The main component of this project is the MAX30100 sensor. The **MAX30100** sensor is a simple module that communicates in the **I2C** interface with the microcontroller. It provides the **SpO2** and **pulse information** to the connected microcontroller. In simple terms, this sensor is used for **identifying oxygen saturation**. Hence, this module can be used for monitoring the pulse rate and oxygen saturation level of the blood in a non-invasive form. For the display circuit, Arduino Uno is the type of microcontroller that has been used as the interface between the overall circuit and LCD Display.

HARDWARE COMPONENTS:

- Arduino Uno Board
- MAX30100 Pulse Oximeter Sensor
- 16 x 2 LCD Display with I2C Module.
- Connecting Wires

ARDUINO UNO BOARD:

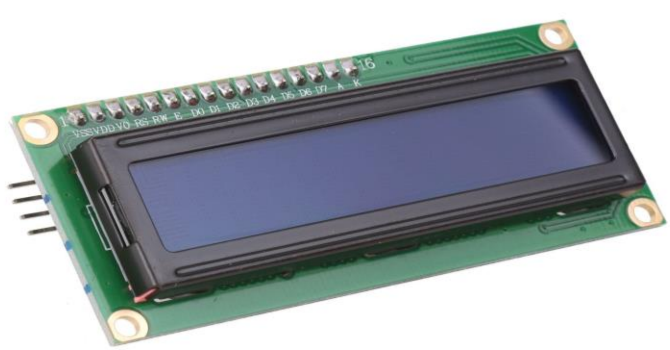
This board is a microcontroller based on ATmega328. This board contains 14 incoming and outgoing digital pins, six incoming analog pins, one 16 megahertz Ceramic Resonator, one USB port, one power jack (for power inlet), one ICSP header, one reset button, and some other small chips. The required voltage for the Arduino Uno board can be supplied through either the USB connection or an external power supply; such as a battery or an AC/DC adaptor.

MAX30100 Pulse Oximeter Sensor:

The sensor is an integrated pulse oximetry and heart-rate monitor sensor solution. It combines two LED's, a photo detector, optimized optics, and low-noise analog signal processing to detect pulse and heart-rate signals. It operates from 1.8V and 3.3V power supplies and can be powered down through software with negligible standby current, permitting the power supply to remain connected at all times.

16 x 2 LCD DISPLAY WITH I2C MODULE:

An LCD (Liquid Crystal Display) monitor is an electronic display module having a wide range of applications. A 16x2 LCD is a basic module and is widely used in various devices and circuits. A 16x2 LCD has two lines on it and sixteen characters can be displayed on each line. The I2C converts the 16 pins into 4 pins.



Working of MAX30100 Pulse Oximeter and Heart-Rate Sensor:

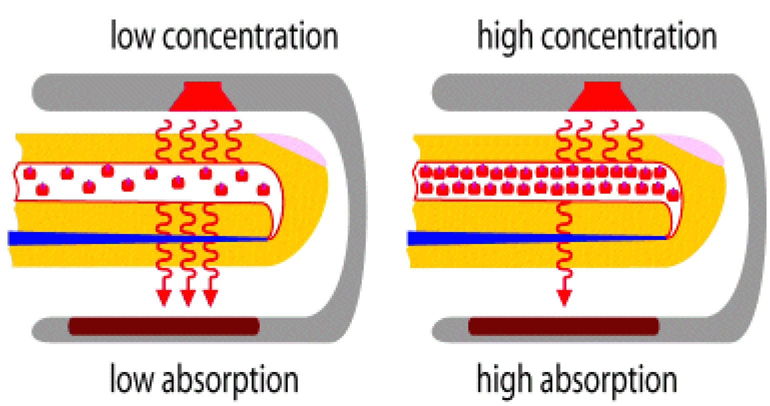
The device has two LEDs, one emitting red light, another emitting infrared light. For pulse rate, only the infrared light is needed. Both the red light and infrared light is used to measure oxygen levels in the blood.

When the heart pumps blood, there is an increase in oxygenated blood as a result of having more blood. As the heart relaxes, the volume of oxygenated blood also decreases. By knowing the time between the increase and decrease of oxygenated blood, the pulse rate is determined.

It turns out, oxygenated blood absorbs more infrared light and passes more red light while deoxygenated blood absorbs red light and passes more infrared light. This is the main function of the MAX30100: it reads the absorption levels for both light sources and stored them in a buffer that can be read via I2C.

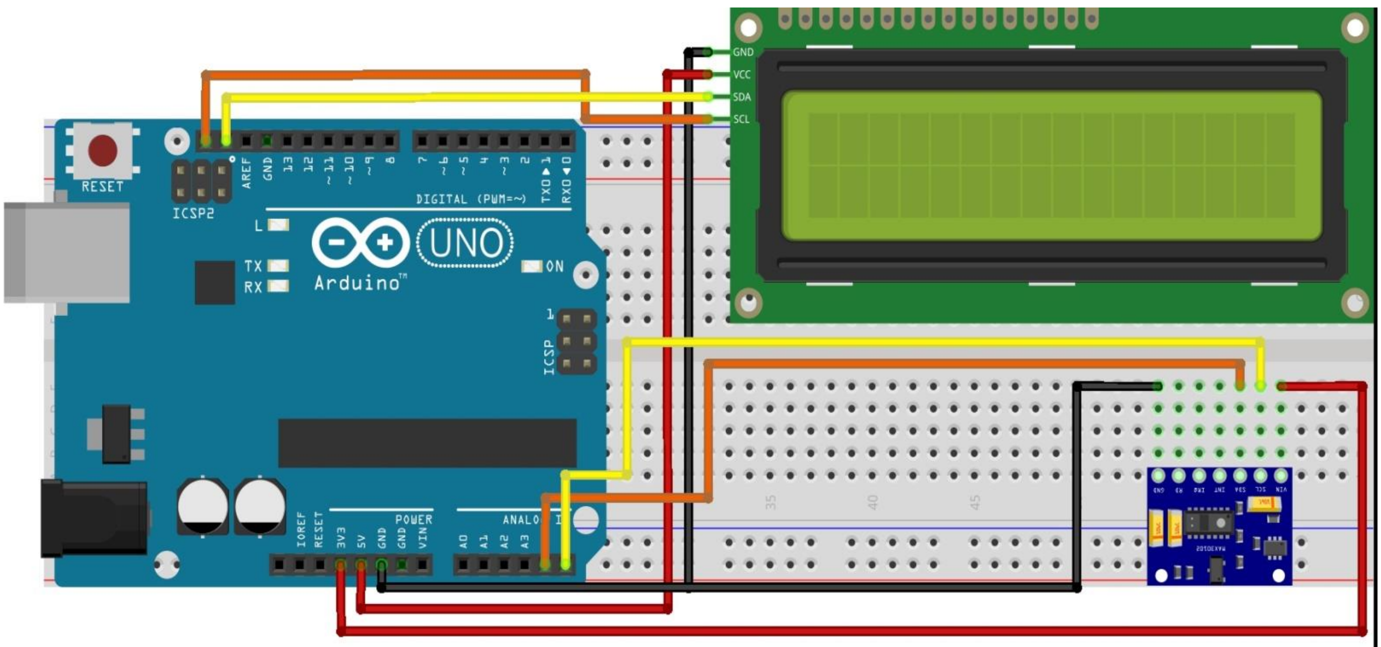
How Does Pulse Oximeter Works?

Oxygen enters the lungs and then is passed on into blood. The blood carries oxygen to the various organs in our body. The main way oxygen is carried in our blood is by means of hemoglobin. By placing the finger on the sensor, small beams of light pass through the blood in the finger, measuring the amount of oxygen. It does this by measuring changes in light absorption in oxygenated or deoxygenated blood.



Circuit Diagram & Connections:

The connections of pulse oximeter are very simple. As MAX30100 works on I2C Communication Protocol, so it's SDA & SCL pin is connected to I2C pin of Arduino, i.e. A4 & A5. And the ground pin and VCC pin of the sensor is connected to the ground and 3.3V of arduino respectively. While the SCL & SDA pin of LCD is connected to the SCL & SDA pins of arduino and the ground and VCC is connected to the ground and 5V of arduino respectively as shown below.



CODE:

In this project, we are using two libraries of LCD and the sensor. For LCD, we use LiquidCrystal_I2C.h and for the sensor we use MAX30100_PulseOximeter.h that is:

```
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include "MAX30100_PulseOximeter.h"
```

For communicating the LCD with arduino, we provide the address that is 0x25 and 16, 2 is the number of columns and rows respectively.

```
LiquidCrystal_I2C lcd(0x25, 16, 2);
```

We use the function of smile, sad and moderate face emoji according to the result of SpO2.

```
byte smile[] = {
  B00000,
  B00000,
  B01010,
  B00000,
  B10001,
  B01110,
  B00000,
  B00000
};
byte mod[] = {
  B00000,
  B00000,
  B01010,
  B00000,
  B11111,
  B00000,
  B00000,
  B00000
};
byte sad[] = {
  B00000,
  B00000,
  B01010,
  B00000,
  B01110,
  B10001,
  B00000,
  B00000
};
```

If the SpO2 is greater than or equal to 96, it smiles. And If the SpO2 is less than or equal to 95 and greater than or equal to 91, the moderate/average emoji arises. And if SpO2 is less than 91, the sad emoji arrives which is given below in the following section of loop.

```
if (pox.getSpO2() >= 96) {
  lcd.setCursor(15 , 1);
  lcd.write(1);
}
else if (pox.getSpO2() <= 95 && pox.getSpO2() >= 91) {
  lcd.setCursor(15 , 1);
  lcd.write(2);
}
else if (pox.getSpO2() <= 90) {
  lcd.setCursor(15 , 1);
  lcd.write(3);
}
```

In the setup section, we initialize the Pulse Oximeter and check if it is working or not. After the successful initialization, the first screen appears is of the “Pulse Oximeter”.

```
void setup()
{
  Serial.begin(9600);

  lcd.init();
  lcd.backlight();
  lcd.createChar(1 , smile);
  lcd.createChar(2 , mod);
  lcd.createChar(3 , sad);
  lcd.setCursor(0, 0);
  lcd.print("      Pulse");
  lcd.setCursor(0, 1);
  lcd.print("    Oximeter");
  delay(1000);

  if (!pox.begin()) {
    Serial.println("FAILED");
    for (;;);
  } else {
    Serial.println("SUCCESS");
  }
  pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);

  pox.setOnBeatDetectedCallback(onBeatDetected);
}
```

And in the loop section, we set the BPM and SpO2 result which changes after every beat.

```
lcd.clear();
lcd.setCursor(0 , 0);
lcd.print("BPM : ");
lcd.print(pox.getHeartRate());
lcd.setCursor(0 , 1);
lcd.print("Sp02: ");
lcd.print(pox.getSpO2());
lcd.print("%");
tsLastReport = millis();
```

FOLLOWING IS THE FULL CODE OF THE SYSTEM:

```
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include "MAX30100_PulseOximeter.h"

#define REPORTING_PERIOD_MS     1000

LiquidCrystal_I2C lcd(0x25, 16, 2);


byte smile[] = {
  B00000,
  B00000,
  B01010,
  B00000,
  B10001,
  B01110,
  B00000,
  B00000
};
byte mod[] = {
  B00000,
  B00000,
  B01010,
  B00000,
  B11111,
  B00000,
  B00000,
  B00000
};


byte sad[] = {
  B00000,
  B00000,
  B01010,
  B00000,
  B01110,
  B10001,
  B00000,
  B00000
};

PulseOximeter pox;
uint32_t tsLastReport = 0;

void onBeatDetected()
{

  Serial.println("Beat!!!");

}

void setup()
{
  Serial.begin(9600);

  lcd.init();
  lcd.backlight();
  lcd.createChar(1 , smile);
  lcd.createChar(2 , mod);
  lcd.createChar(3 , sad);
  lcd.setCursor(0, 0);
```

```
  lcd.print("       Pulse");
  lcd.setCursor(0, 1);
  lcd.print("     Oximeter");
  delay(1000);

  if (!pox.begin()) {
    Serial.println("FAILED");
    for (;;);
  } else {
    Serial.println("SUCCESS");
  }
  pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);

  pox.setOnBeatDetectedCallback(onBeatDetected);


void loop()

  pox.update();
  if (millis() - tsLastReport > REPORTING_PERIOD_MS) {

    lcd.clear();
    lcd.setCursor(0 , 0);
    lcd.print("BPM : ");
    lcd.print(pox.getHeartRate());
    lcd.setCursor(0 , 1);
    lcd.print("Sp02: ");
    lcd.print(pox.getSpO2());
    lcd.print("%");
    tsLastReport = millis();


    if (pox.getSpO2() >= 96) {
      lcd.setCursor(15 , 1);
      lcd.write(1);
    }
    else if (pox.getSpO2() <= 95 && pox.getSpO2() >= 91) {
      lcd.setCursor(15 , 1);
      lcd.write(2);
    }
    else if (pox.getSpO2() <= 90) {
      lcd.setCursor(15 , 1);
      lcd.write(3);
    }
  }
}
```

ADVANTAGES:

- Simple to use
- Non-invasive
- Require no warm up time.
- Cost and user-friendly.
- Operating voltage is only 1.8V – 3.3V.

DISADVANTAGES:

- Oxygen sensor for blood SpO2 measurement does not give accurate readings under certain situations which include interference from external light, poor blood circulation or cold hands

CONCLUSION:

Pulse oximeters have been developed and widely used since the 1930s. The designs and products have evolved from large sizes to small ones. Hence, this project is effective in all medical fields to check the oxygen of the patients and very easy to use.