

LABORATÓRIO DE INOVAÇÃO TECNOLÓGICA EM SAÚDE- LAIS

RELATÓRIO TÉCNICO: PROJETO PERFIL 1 - INSTRUMENTAÇÃO BIOMÉDICA

GILBERTO MARTINS FILHO

**NATAL - RN
2022**

INTRODUÇÃO

A capnografia é o monitoramento em tempo real da concentração ou pressão parcial de dióxido de carbono nos gases respiratórios. Utilizado quando há uma ventilação pulmonar mecânica, essa ferramenta é um importante instrumento na identificação da insuficiência, falha e até parada respiratória. O CO₂ produzido pela respiração celular é transportado pelo sistema venoso até os pulmões, onde há a troca gasosa para exalar esse gás (GRAVENSTEIN, 2011).

O presente relatório técnico consiste no desenvolvimento de um sistema de aquisição de dados de pressão parcial de CO₂ na respiração em tempo real e apresentar os dados em um terminal, armazená-los, e alertar o usuário caso o algoritmo encontre alguma anormalidade.

PROJETO PROPOSTO

Desenvolvimento do sistema de aquisição

O sistema de aquisição de dados de pressão parcial de CO₂ na respiração descrito neste relatório utilizou um sensor de gás disponível no ThinkerCAD para realizar a montagem e simulação do seu funcionamento. O sensor disposto na plataforma é o de sensor de fumaça, mas com funcionamento parecido com o do CO₂. O modelo do sensor proposto para uso neste projeto é o de MQ-7, um sensor acessível e com modelo similar para prototipagem.

O sensor será utilizado em conjunto com o arduino MEGA 2560. Entretanto, para o sistema demonstrado aqui, o arduino disposto no ThinkerCAD é o UNO e ele que foi utilizado. A Figura 1 mostra a disposição de montagem dos equipamentos, onde também será utilizado um buzzer para alarme sonoro e uma tela LCD para comunicação junto ao usuário. Já a Figura 2 é a vista esquemática do circuito.

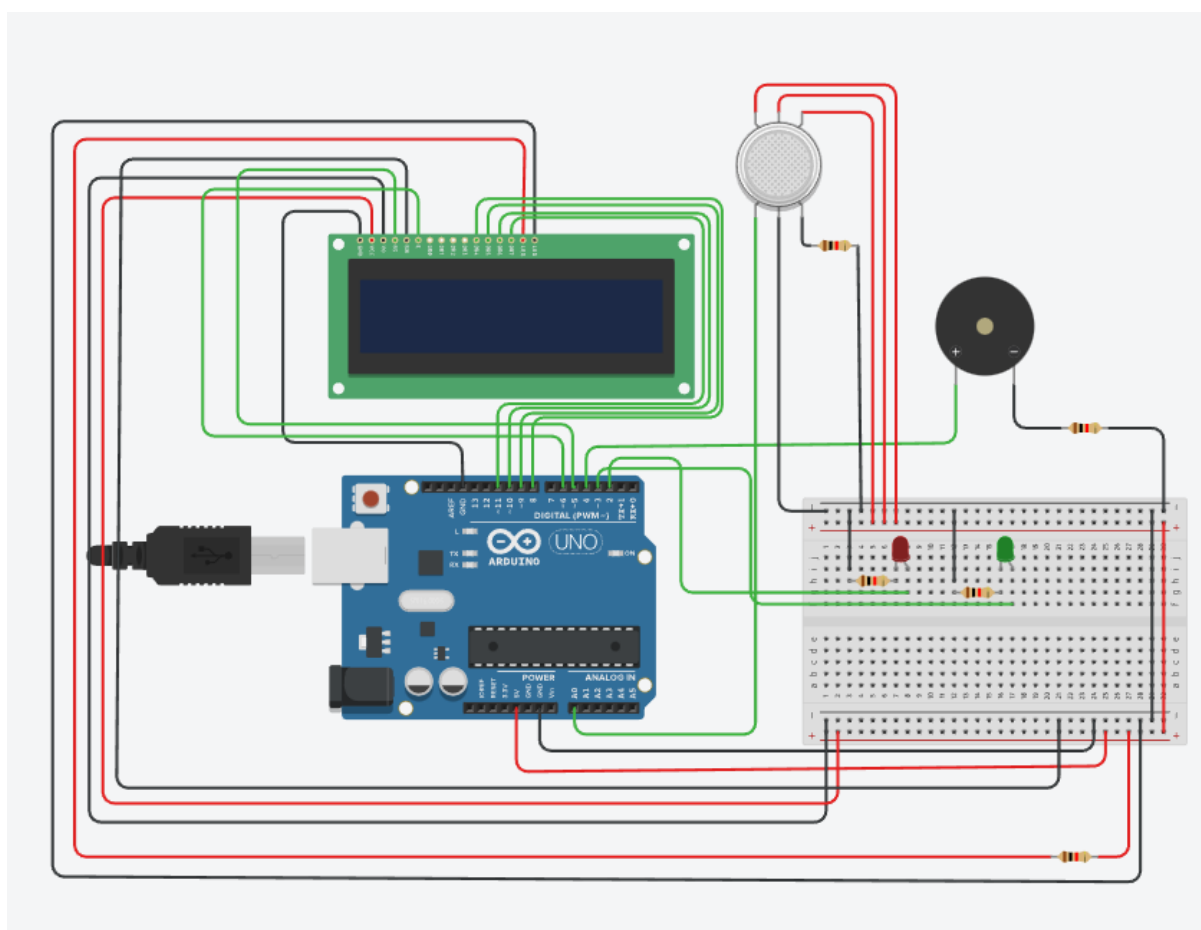


Figura 1 - Sistema embarcado produzido no ThinkerCAD para exemplificação da aquisição da pressão parcial de CO₂. (Fonte: Autor, 2022)

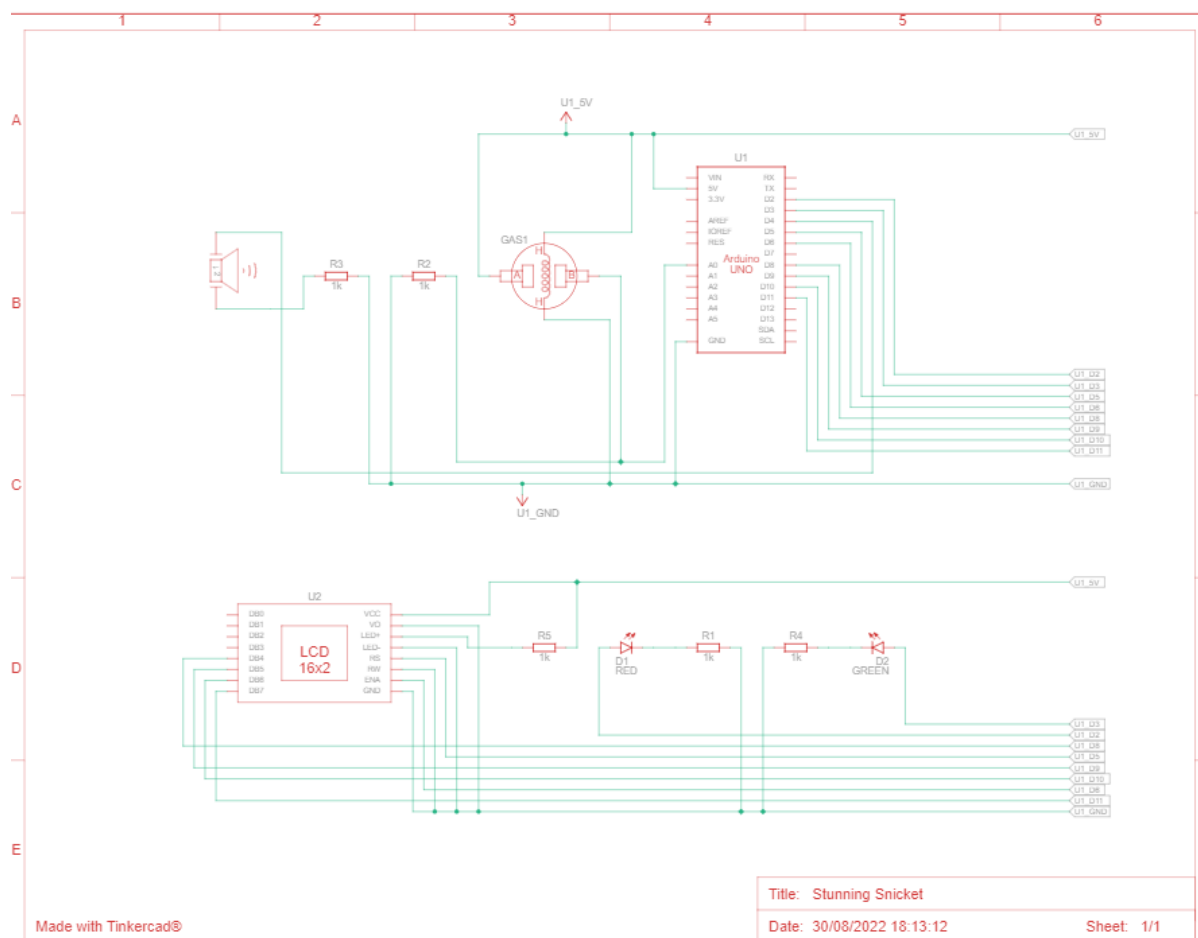


Figura 1 - Vista esquemática produzida no ThinkerCAD para exemplificação da aquisição da pressão parcial de CO₂. (Fonte: Autor, 2022)

Desenvolvimento do firmware

O firmware foi desenvolvido em C++ na interface do ThinkerCAD, que simula o arduino IDE, e suas principais tarefas estão demonstradas na Figura 2. Inicialmente há a aquisição do sinal e a comunicação com os sensores, onde o arduino passa a receber informações e enviá-las via porta serial.

O sistema de notificações ao usuário também foi implementado neste algoritmo. Valores de entre 35-45mmHg são considerados valores de indivíduos saudáveis (GRAVENSTEIN, 2011), onde valores acima de 45mmHg mostram que há maior concentração de CO₂ no sangue demonstrando uma deficiência na troca gasosa. Valores obtidos a partir de 45mmHg é notificado como um valor perigoso e gerando um aviso sonoro com estes valores e o LED vermelho será acionado. Caso estejam em funcionamento normal, o LED verde estará ligado e mensagens aparecerão no LCD.

A Figura 3 mostra o algoritmo para realizar o armazenamento dos dados coletados pelo sensor. Além dos valores de pressão parcial de CO₂, há o armazenamento do tempo de aquisição, data e hora exata de cada aquisição. O arquivo gerado é um .txt e acoplado diretamente no sistema embarcado. Já a Figura 4 mostra um código em python para realizar a comunicação com a porta serial do arduino e assim implementar posteriormente uma interface gráfica.

```

1  #include <LiquidCrystal.h>
2  LiquidCrystal lcd(5,6,8,9,10,11);
3
4  int redled = 3;
5  int greenled = 2;
6  int buzzer = 4;
7  int sensor = A0;
8  int sensorThresh = 30;
9  // C++ code
10 //
11 void setup()
12 {
13     pinMode(redled, OUTPUT);
14     pinMode(greenled, OUTPUT);
15     pinMode(buzzer, OUTPUT);
16     pinMode(sensor, INPUT);
17     Serial.begin(9600);
18     lcd.begin(16,2);
19
20 }
21
22 void loop()
23 {
24     int analogValue = analogRead(A0);
25     int b = map(analogValue,0,1023,0,255);
26     Serial.println(b);
27     if(b>sensorThresh)
28     {
29         digitalWrite(redled, HIGH);
30         digitalWrite(greenled, LOW);
31         tone(buzzer, 1000, 10000);
32         lcd.clear();
33         lcd.setCursor(0,1);
34         lcd.print("ALERTA");
35         delay(700);
36         lcd.setCursor(0,1);
37         lcd.print("NIVEL ALTO");
38         delay(700);
39     }
40     else
41     {
42         digitalWrite(greenled, HIGH);
43         digitalWrite(redled, LOW);
44         noTone(buzzer);
45         lcd.clear();
46         lcd.setCursor(0,0);
47         lcd.print("SEGURO");
48         delay(700);
49         lcd.clear();
50         lcd.setCursor(0,1);
51         lcd.print("NIVEL CORRETO");
52         delay(700);
53     }
54 }

```

Figura 2 - Firmware de aquisição dos dados do sensor e notificação caso.

```

1  import processing.serial.*;
2  Serial mySerial;
3  PrintWriter output;
4
5  void setup() {
6      String portName = Serial.list()[0];
7      mySerial = new Serial(this, portName, 9600);
8      output = createWriter( "data.txt" );
9  }
10
11 void draw() {
12     if (mySerial.available() > 0 ) {
13         String value = mySerial.readStringUntil('\n'); //Le o valor recebido
14         if ( value != null ) {
15             output.print(hour()); //Escreve no arquivo as horas e os minutos atuais seguido do valor lido pelo sensor
16             output.print(":");
17             output.print(minute());
18             output.print("->");
19             output.println(value);
20             output.flush();
21         }
22     }
23 }
24
25 void keyPressed() {
26     output.flush(); // Termina de escrever os dados pro arquivo
27     output.close(); // Fecha o arquivo
28     exit(); // Para o programa
29 }

```

Figura 3 - Algoritmo para salvar em arquivo .txt o sinal de pressão parcial de CO₂.

Projeto 1 - Fase 2 - LAIS/UFRN

Nome: Gilberto Martins Filho

E-mail: gilberto.filho@edu.isd.org.br

Instalando as bibliotecas necessárias

```
[1]: !pip install pyserial

Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting pyserial
  Downloading pyserial-3.5-py2.py3-none-any.whl (90 kB)
    |#####| 90 kB 3.8 MB/s
Installing collected packages: pyserial
Successfully installed pyserial-3.5
```

Importando as bibliotecas

```
[6]: import serial
import matplotlib.pyplot as plt
import matplotlib.animation as animation
import numpy as np
import pandas as pd
import time
```

Iniciando a conexão Serial

Aqui será iniciada a conexão com a porta serial onde o arduino está conectado e realizando a leitura dos dados.

```
[ ]: arduino = serial.Serial('/caminho_da_porta_serial', 9600, timeout=1)

[ ]: data = float(arduino.readline())
```

Transformando o sinal recebido

O sinal recebido está em PPM, onde deve ser convertido para pressão parcial de CO₂. Assim, realizaremos a seguinte operação:

```
[ ]: #PPM to volume
data_volume = data/10000

[ ]: #Pressão parcial do CO2 - usaremos a pressão atmosférica
pressure = 760
PPCO2 = (data_volume/pressure)/100 #Esse resultado dá a pressão parcial de CO2 em mmHg
```

Realizando a coleta dos dados e fazendo o processamento desses dados

```
[ ]: df = pd.DataFrame()

while True:
    data = float(arduino.readline())
    data_volume = data/10000
    pressure = 760
    PPCO2 = (data_volume/pressure)/100
    t = time

    #plotando o gráfico
    plt.plot(PPCO2, t)
    plt.title('Pressão parcial de CO2')
    plt.xlabel('PPCO2 mmHg')
    plt.ylabel('Tempo (s)')

    #salvando os dados
    caminho = 'caminho para salvar o arquivo'
    df = df.append([PPCO2, time], ignore_index=True)

print(df.shape)
df.to_excel('caminho_do_arquivo.xls', index=False)
```

Figura 4 - Comunicação com arduino em python.

CONCLUSÕES

O projeto proposto teve o objetivo de desenvolver um sistema de aquisição de pressão parcial de CO₂ durante um ciclo respiratório, além de elaborar ferramentas para comunicação com o sistema. Nesse contexto, o projeto buscou satisfazer as condições e necessidades propostas no edital.

REFERÊNCIAS

GRAVENSTEIN, Joachim S. et al. (Ed.). **Capnography**. Cambridge University Press, 2011.

DO AMARAL, José Luiz Gomes; FERREZ, Antonio Carlos Pires Ferreira David; GERETTO, Pedro. **Monitorização da respiração: oximetria e capnografia**. Brazilian Journal of Anesthesiology, v. 42, n. 1, p. 51-58, 2020.

CHAN, Hiang Ping; LEWIS, Craig; THOMAS, Paul S. **Exhaled breath analysis: novel approach for early detection of lung cancer**. Lung Cancer, v. 63, n. 2, p. 164-168, 2009.

HANWEI. **Data Sheet MQ-7 GAS SENSOR**. 2014. Disponível em: <<https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-7.pdf>>