

CENTRO PAULA SOUZA



**Etec Paulino Botelho
Técnico em Mecatrônica**

José Ronaldo da Silva
Ricardo Bruno Napoleão

FECHADURA INTELIGENTE

**São Carlos
2023**

Etec Paulino Botelho
Técnico em mecatrônica

José Ronaldo da Silva
Ricardo Bruno Napoleão

FECHADURA INTELIGENTE

Trabalho de Conclusão de Curso apresentado ao Técnico em Mecatrônica da Etec Paulino Botelho, orientado pelo docente Claudio Torres Gonçalves, como requisito obrigatório para obtenção do título de Técnico em Mecatrônica.

São Carlos
2023

Fechadura Biométrica

Trabalho de Conclusão de Curso apresentado ao Curso Técnico em Mecatrônica da ETEC Paulino Botelho, como requisito parcial para obtenção do título de Técnico em Mecatrônica.

Monografia aprovada em: ____/____/____.

Orientador: _____

Prof. Cláudio Torres Gonsalves.

1º Examinador (a): _____

Prof. Evandra Maria Raymundo.

2º Examinador (a): _____

Prof. Anderson Angelo Beluco

Coordenador do curso: _____

DEDICATÓRIA

Dedicamos este trabalho em primeiro lugar a Deus, a Etec e aos nossos professores que se esforçaram para nos passar o conhecimento necessário para chegarmos ao término deste curso com êxito.

AGRADECIMENTOS

Aos caros alunos do grupo, pela convivência harmoniosa e saudável que vivenciamos durante o período de curso. Juntos compartilhamos e trocamos experiência para desenvolver nosso TCC.

Agradecemos também a todos os professores pelo empenho, apoio e dedicação.

EPÍGRAFE

“Quando se quer muito é possível alcançar”

(Grupo)

RESUMO

O tema aqui proposto tem por finalidade compreender o funcionamento de um sistema automatizado utilizando um dispositivo de tranca via Arduino controlado por sistema biométrico. Como fase inicial será montado um protótipo de uma tranca, baseado em nossos conhecimentos de mecatrônica, enquanto o software será implementado com base nos conhecimentos adquiridos em aula que foram desenvolvidas a partir de um estudo sobre sistemas biométricos. Esta aplicação tem por objetivo fazer a segurança em alto nível com uso de um dispositivo biométrico acoplado, de forma que somente o dono ou a pessoa cadastrada e autenticada pelo sistema poderá abrir as trancas da residência ou local em que o projeto for instalado. Objetiva desenvolver um sistema bem simples, porém muito seguro assim como de fácil entendimento para o usuário.

Palavras-chave: Fechadura biométrica; Arduino.

SUMMARY

The theme proposed here aims to understand the operation of an automated system using a locking device via Arduino controlled by a biometric system. As an initial phase, a prototype of a lock will be assembled, based on our knowledge of mechatronics, while the software will be implemented based on the knowledge acquired in class that was developed from a study on biometric systems. This application aims to provide high-level security using an attached biometric device, so that only the owner or the person registered and authenticated by the system will be able to open the locks of the residence or place where the project is installed. It aims to develop a very simple but very secure system, as well as easy to understand for the user. Keywords: Biometric lock; arduino.

LISTA DE FIGURAS

Figura 1: Montagem de sensores	14
<i>Figura 2: Sensores Montados.....</i>	<i>14</i>
Figura 3: Case 3D	15
Figura 4: Case 3D Impressa	15
Figura 5: Solenoide	17
Figura 6: Porta	17
Figura 7: Porta e batente.....	17
Figura 8: Porta e Led (Frente).....	18
Figura 9: Porta e Led (Verso).....	18
Figura 10: Placa	18
Figura 11: Porta Finalizada Pintura Verso.....	18
Figura 12: Sensor Biométrico	19
Figura 13: Diagrama Protoboard.....	21
<i>Figura 14: Diagrama e Pinout sensor.....</i>	<i>21</i>

SUMÁRIO

1. INTRODUÇÃO.....	10
2. OBJETIVO.....	11
3. JUSTIFICATIVA	12
4. METODOLOGIA.....	13
5. DESENVOLVIMENTO	14
5.1. MONTAGEM DA FECHADURA.....	14
5.2. COMPONENTES DA FECHADURA	15
5.3. CASE 3D.....	15
5.4. ESTRUTURA DA FECHADURA INTELIGENTE.....	16
5.5. SOLENÓIDE 12V.....	17
5.6. PORTA.....	17
5.7. SENSOR BIOMETRICO.....	19
6. CONCLUSÃO.....	20
7. DIAGRAMA.....	21
9. REFERÊNCIA.....	22

1. INTRODUÇÃO

As passagens entre ambientes internos ou externos de uma maneira geral possuem sistema mecânico de manuseio de suas trancas, sendo necessário uma pessoa para sua abertura ou fechamento. Foi observado essas passagens que poderiam ficar abertos por um grande período de tempo sem nenhum tipo de controle de acesso aos ambientes, ficando expostos muitos bens de valor econômico e técnico a possíveis avarias e furtos. Alguns desses equipamentos sendo únicos. A partir das problemáticas observadas nesses ambientes do nosso cotidiano, fica claro as pessoas que acessa os ambientes a necessidade de desenvolver um sistema que fizesse o controle de acesso a estes espaços físicos, permitindo a entrada de somente pessoas autorizadas e também a necessidade de um histórico de acessos a estes locais. Foi realizada uma pesquisa de fechaduras eletrônicas existentes no mercado, sendo constatado que há vários tipos de fechadura no mercado cada uma atende-se as necessidades específicas e suas características técnicas e mecânicas variando em valores e tamanhos havendo muitas que atendessem as necessidades específicas em questão. Este trabalho será dividido em três capítulos. No primeiro, será apresentado considerações sobre fechaduras eletrônicas. No segundo capítulo, espõem-se os componentes, hardware, software e protocolos. Por fim, no terceiro capítulo, serão apresentadas as etapas do desenvolvimento da fechadura eletrônica proposta.

2. OBJETIVO

Este trabalho tem como objetivo o desenvolvimento de uma fechadura biométrica para o controle de acesso as mais variadas formas de passagens entrada e saída que exija controle, utilizando um sistema que possa fazer a melhor segurança possível aos ambientes assim acessados. Podemos utilizar senhas como chave de acesso, limitar o acesso apenas para pessoas autorizadas, possibilidade de abertura com chave mecânica em caso de falta de energia, criar um cadastro para pessoas autorizadas e seus históricos de entrada e saída, possuir um site para visualização do histórico de acesso, possibilidade de melhorias futuras e fácil substituição dos componentes.

3. JUSTIFICATIVA

A fechadura inteligente, é uma ótima opção para controle de entradas com segurança, tecnologia de ponta e versátil, podendo ser cadastrada até 123 usuários.

Utilizando duas opções de segurança trabalhando em conjunto, são mais seguras que o método atual atualmente padrão, exemplos fechadura convencional de chaves e segredos, podendo ser violadas ou danificadas.

Com a fechadura Inteligente você tem duas opções de abertura, Leitura Biométrica ou senha PIN digitada, trazendo redundância caso a biometria falhe.

4 METODOLOGIA

Este trabalho foi desenvolvido de acordo com os conhecimentos adquiridos no curso Técnico em Mecatrônica do grupo em relação ao desenvolvimento de uma fechadura inteligente, as pesquisas foram feitas através de livros, sites relacionados com o tema abordado e experiência de campo adquiridas em conhecimentos técnicos realizados no dia a dia.

5. DESENVOLVIMENTO

5.1 MONTAGEM DA PARTE ELETRICA

O primeiro processo de montagem começou com a montagem do esquema elétrico digital e diagrama, definindo cada ligação nas suas portas do Arduino em uma protoboard.

Após testes virtuais, iniciamos os segundos testes em bancadas com protoboard e seus componentes.

A terceira etapa se resultou em integrar todos os sensores.

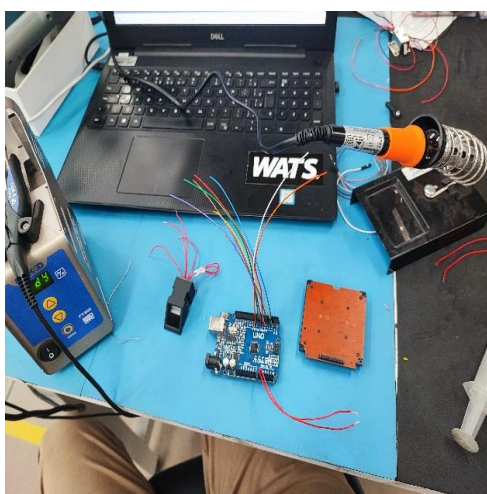


Figura 1: Montagem de sensores

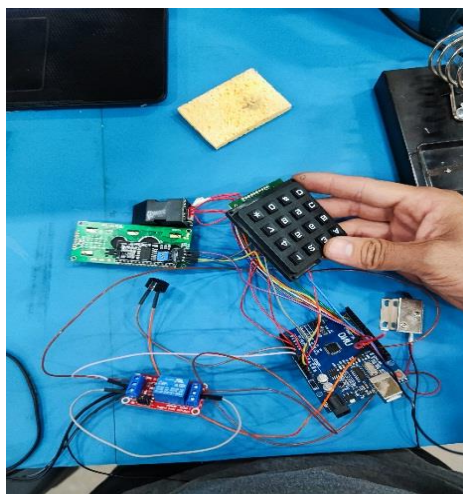


Figura 2: Sensores Montados

5.2 COMPONENTES DA FECHADURA INTELIGENTE

A Fechadura inteligente contou com os seguintes componentes, Arduino Uno, solenoide 12v, relê, bateria 9v, Teclado 16x16 matricial, sensor biométrico, display com i2c, buzzer, leds.

5.3 CASE 3D

A primeira etapa da case 3d foi desenvolver utilizando o software Solid Works 2020, utilizando técnicas que foram absorvidas em sala de aula.

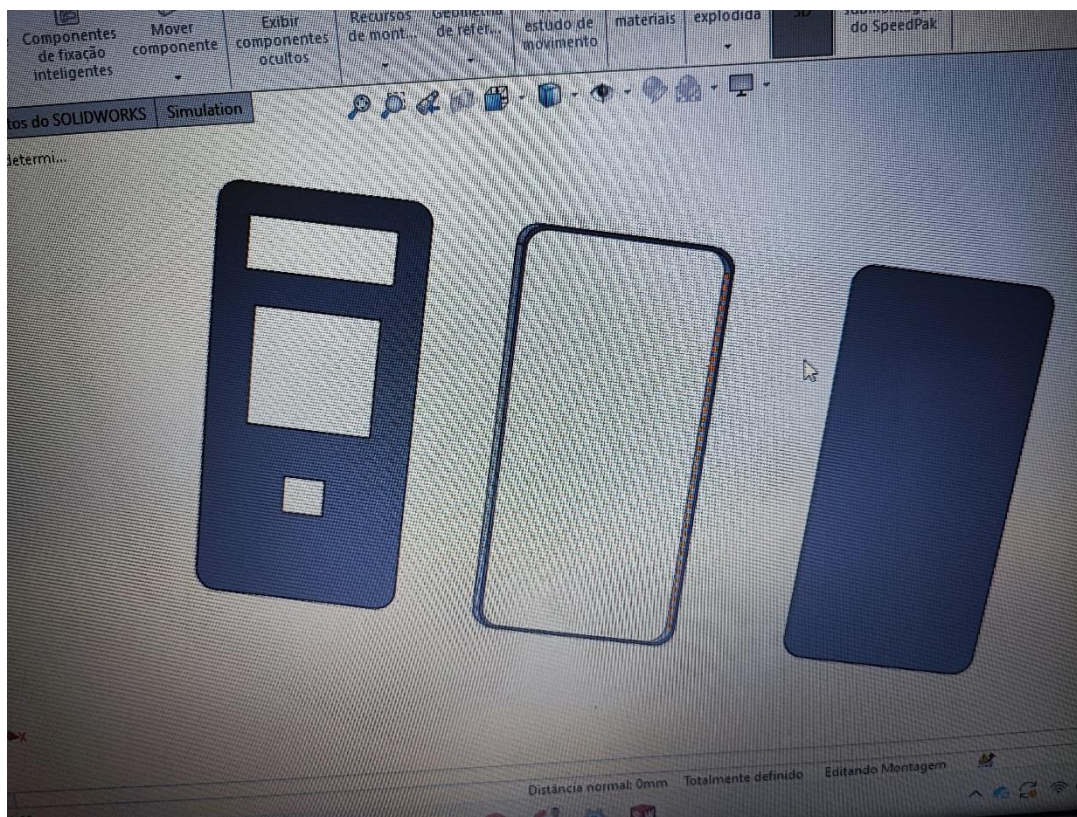


Figura 3: Case 3D

5.4 ESTRUTURA DA FECHADURA INTELIGENTE



Figura 4: Case 3D Impressa

A case foi desenvolvida em material filamento PLA.

5.5 SOLENOIDE 12V

A mini fechadura Solenoide é ideal para a realização de controle de acesso a pequenos ambientes, ela é acionada quando recebe um sinal 12 V assim recolhendo a trava.

Recomendado o acionamento da trava por no máximo 10 segundos para evitar um sobreaquecimento da válvula

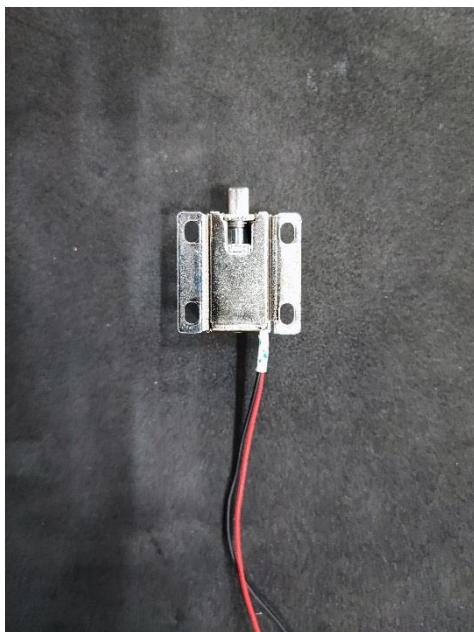


Figura 2: Solenoide

5.6 PORTA

A porta para demonstração da fechadura foi desenvolvida reaproveitando materiais já prontos pelo mercado, no caso foi escolhida uma porta de quadro elétrico.

Para dar consistência a porta, foi utilizado batentes de madeira desenvolvida manualmente.

Utilizamos como suporte para a porta uma base de luminária de led para dar apoio e fixação a porta



Figura 6: Porta



Figura 7: Porta e batente



Figura 8: Porta e Led (frente)



Figura 9: Porta e Led (Verso)



Figura 3: Porta Finalizada Pintura



Figura 11: Porta Finalizada Pintura Verso

5.7 SENSOR BIOMETRICO

O Leitor Biométrico Arduino é um produto que pode agregar além de funcionalidade e praticidade, bastante qualidade aos mais diversificados projetos. Utilizando em instalações residenciais, comerciais e industriais, este pequeno equipamento possui uma facilidade extremamente grande de se encaixar em diversas situações.

Quem nunca saiu de casa e se esqueceu das chaves e depois teve que ficar para o lado de fora até que alguém chegasse, ou que tenha pensado em um modo seguro e profissional de controlar o tráfego de pessoas em sua empresa? Visando lhe proporcionar um projeto simples e prático para lhe sanar esta dificuldade.



Figura 12: Sensor Biométrico

Planilha de gastos

Custo do Projeto

Total

R\$ 352.10

Materials	Preço
Relê	R\$ 22.00
Display LCD	R\$ 22.75
Teclado Matrixial 16x16	R\$ 36.63
Conversor I2C	R\$ 11.30
Sensor Biometrico	R\$ 102.00
Solenóide 12V	R\$ 36.62
Arduino Uno	R\$ 43.90
Porta quadro elétrico	R\$ 39.90
Batentes Madeira	R\$ 22.00
Case Stl	R\$ 15.00

6. CONCLUSÃO

Este trabalho foi desenvolvido para apresentar um dispositivo de segurança, moderno e confiável, voltado para ambientes com uso interno e externo, aplica-se em portas de escritórios, recepções, hotéis, portões, casas residenciais, com o intuito de oferecer tranquilidade e comodidade.

Estamos vivendo em um mundo tecnológico que está em constantes mudanças e inovações, sendo assim se faz necessário novos métodos e conhecimentos que auxiliem o dia a dia das pessoas.

7. Diagrama

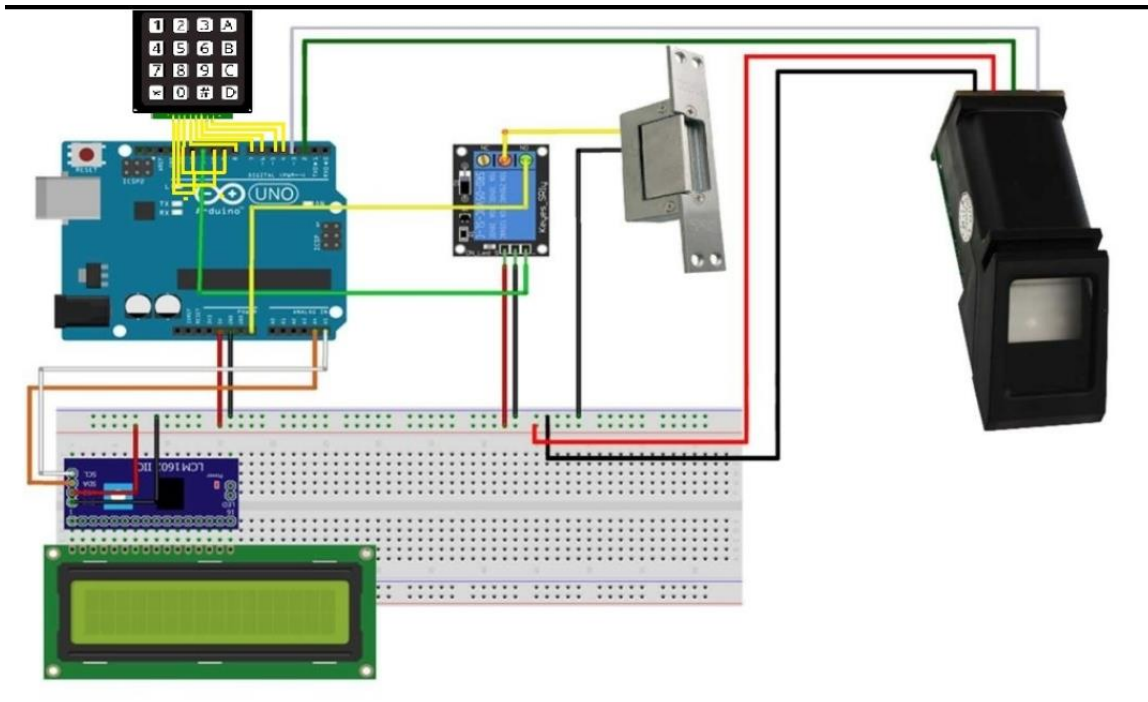


Figura 13: Diagrama Protoboard

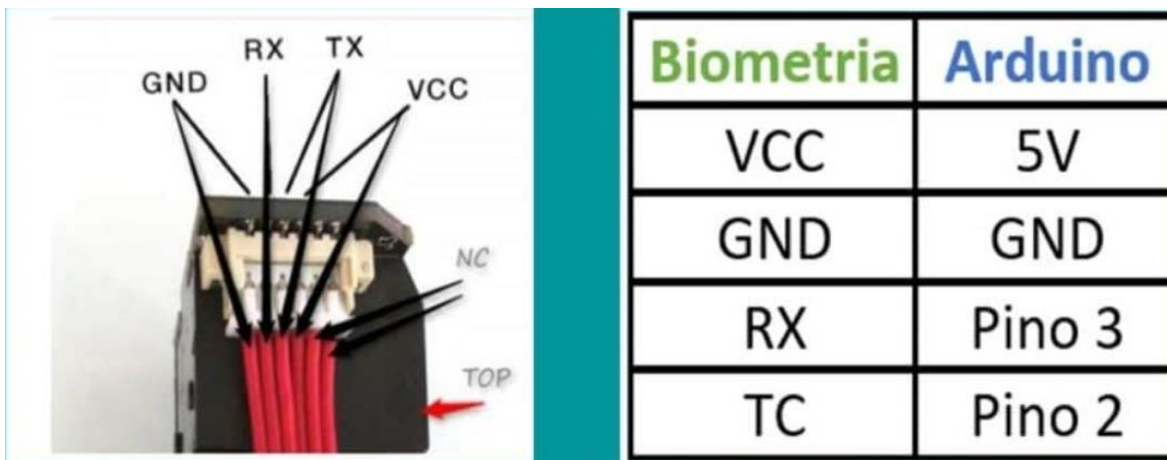


Figura 14: Diagrama e Pinout sensor

Código de programação Arduino

```
//biblioteca responsavel pela comunicação com o display
#include <LiquidCrystal_I2C.h>
// biblioteca para comunicação I2C
#include <wire.h>
//biblioteca para manipular objeto sensor biometrico
```

```

#include <Adafruit_Fingerprint.h>
// biblioteca para comunicação serial com sensor biometrico
#include <SoftwareSerial.h>
//biblioteca responsável por capturar a tecla que foi pressionada no
teclado
#include <Keypad.h>

int getFingerprintIDin();
int getFingerprintIDez();
SoftwareSerial mySerial(A3,A0);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
LiquidCrystal_I2C lcd(0x3F,16,2);

int autofalante=12;
int freq=500; //frequencia para o bip
int dur=120; //duração do bip
const byte LINHAS = 4; //número de linhas do teclado
const byte COLUNAS = 4; //número de colunas do teclado

const int PINO_LED_VERMELHO = 10; // LED vermelho conectado ao pino
11
const int PINO_LED_VERDE = 11; // LED verde conectado ao pino 10
const int rele = 13;
//define uma matriz com os símbolos que deseja ser lido do teclado
char SIMBOLOS[LINHAS][COLUNAS] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};

byte PINOS_LINHA[LINHAS] = {5, 4, 3, 2}; //pinos que indicam as linhas do
teclado
byte PINOS_COLUNA[COLUNAS] = {9, 8, 7, 6}; //pinos que indicam as
colunas do teclado

//instancia de Keypad, responsável por capturar a tecla pressionada
Keypad customKeypad = Keypad( makeKeymap(SIMBOLOS),
PINOS_LINHA, PINOS_COLUNA, LINHAS, COLUNAS);

//variáveis responsáveis por armazenar as senhas
const String SENHA_ESPERADA = "0506";
String SENHA_DIGITADA = "";

void setup(){

  lcd.init();// inicia display

```



```

lcd.backlight();//liga backlight do display
inicio();
finger.begin(57600);//inicia sensor biometrico

digitalWrite(PINO_LED_VERDE, LOW); // LED Verde apagado
digitalWrite(PINO_LED_VERMELHO, LOW); // LED Vermelho apagado
pinMode(rele,OUTPUT);
}

void loop(){

//captura a tecla pressionada do teclado
char customKey = customKeypad.getKey();
char numteclado = customKey;

//caso alguma tecla foi pressionada
if (customKey){

; switch(customKey)
{
//caso alguma das teclas imprimíveis foi pressionada
case 'A':
case 'B':

case 'D':
case '0':
case '1':
case '2':
case '3':
case '4':
case '5':
case '6':
case '7':
case '8':
case '9':

//concatena o novo símbolo a senha que estamos digitando
SENHA_DIGITADA+=customKey;
Serial.println(SENHA_DIGITADA);
//imrpime na tela o símbolo pressionado
lcd.print(customKey);
break;

```

```

//caso a tecla CLEAR tenha sido pressionada
case '*':
    //limpa a variável que guarda a senha que está sendo digitada
    SENHA_DIGITADA = "";
    //chama o comando para limpar a tela
    lcd.clear();
    //configura a mensagem para digitar a senha
    inicio();
    break;
//caso a tecla ENTER seja pressionada, devemos comparar as senhas
case 'C':
    //limpa a tela
    lcd.clear();

    //se a senha digitada foi igual a ESPERADA
    if(SENHA_ESPERADA==SENHA_DIGITADA)
    {
        lcd.blink_off();
        lcd.setCursor(1,0);
        lcd.print("Senha Correta");
        lcd.setCursor(3,1);
        lcd.print("BEM VINDO");

        //rotina de perifericos
        tone(autofalante,1000);
        delay(1000);
        noTone(autofalante);
        digitalWrite(rele,HIGH);
        delay(1000);
        digitalWrite(rele,LOW);

        for(int i = 0; i < 5; i++){ // Pisca o LED 5 vezes rapidamente
            digitalWrite(PINO_LED_VERDE, HIGH);
            delay(50);
            digitalWrite(PINO_LED_VERDE, LOW);
            delay(50);
        }
    }
    //caso senha esteja errada
    else{

        lcd.setCursor(1,0);
        lcd.print("Senha Incorreta!");
        lcd.setCursor(1,1);
        lcd.print("Tente Novamente");

        //rotina de perifericos
        for(int i = 0; i < 5; i++){ // Pisca o LED 5 vezes rapidamente
            tone(autofalante,600);

```

```

    delay(50);
    tone(autofalante,600);
    delay(50);
    tone(autofalante,600);
    delay(50);
    noTone(autofalante);
    digitalWrite(PINO_LED_VERMELHO, HIGH);
    delay(50);
    digitalWrite(PINO_LED_VERMELHO, LOW);
    delay(50);
  }
}
//aguarda 2 segundos para limpar a tela novamente e esperar uma
nova senha ser digitada
delay(5000);
lcd.clear();
SENHA_DIGITADA = "";
inicio();
break;
default: break;

}

}
else{

  getFingerprintIDez();

}

//rotina para mudar o som das teclas
if (numteclado == '1'){
  tone(autofalante,freq,dur);

}

if (numteclado == '2'){
  tone(autofalante,freq*1.25,dur);

}

if (numteclado == '3'){

```

```
tone(autofalante,freq*1.50,dur);  
}  
  
if (numteclado == '4'){  
tone(autofalante,freq*1.75,dur);  
}  
  
if (numteclado == '5'){  
tone(autofalante,freq*2,dur);  
}  
  
if (numteclado == '6'){  
tone(autofalante,freq*2.25,dur);  
}  
  
if (numteclado == '7'){  
tone(autofalante,freq*2.50,dur);  
}  
  
if (numteclado == '8'){  
tone(autofalante,freq*2.75,dur);  
}  
if (numteclado == '9'){  
tone(autofalante,freq*3,dur);  
}  
  
if (numteclado == '0'){  
tone(autofalante,freq*3.25,dur);  
}  
if (numteclado == '*'){  
tone(autofalante,freq*3.50,dur);  
}  
if (numteclado == '#'){  
tone(autofalante,freq*3.75,dur);  
}  
  
delay(100);
```

```
}

```

```
//função responsável por imprimir na tela a mensagem para digitar a
senha
//é chamada toda vez q a senha foi digitada e comparada, também quando
//a tecla limpar display foi pressionada.
void inicio(){
```

```
    lcd.setCursor(0,0);
    lcd.print("Senha ou Digital");
    lcd.setCursor(3,1);
    lcd.print("Senha:");
    lcd.blink();
}
```

```
uint8_t getFingerprintID() {
    uint8_t p = finger.getImage();
    switch (p) {
        case FINGERPRINT_OK:

            break;
        case FINGERPRINT_NOFINGER:

            return p;
        case FINGERPRINT_PACKETRECEIVEERR:
            return p;
        case FINGERPRINT_IMAGEFAIL:

            return p;
        default:

            return p;
    }
}
```

```
p = finger.image2Tz();
switch (p) {
    case FINGERPRINT_OK:

        break;
    case FINGERPRINT_IMAGEMESS:
```

```

    return p;
case FINGERPRINT_PACKETRECIEVEERR:

    return p;
case FINGERPRINT_FEATUREFAIL:

    return p;
case FINGERPRINT_INVALIDIMAGE:

    return p;
default:

    return p;
}

p = finger.fingerFastSearch();
if (p == FINGERPRINT_OK) {

} else if (p == FINGERPRINT_PACKETRECIEVEERR) {

    return p;
} else if (p == FINGERPRINT_NOTFOUND) {

    return p;
}
else {

    return p;
}

}

int getFingerprintIDez() {
    uint8_t p = finger.getImage();
    if (p != FINGERPRINT_OK) return -1;

    p = finger.image2Tz();
    if (p != FINGERPRINT_OK) return -1;

    p = finger.fingerFastSearch();
    if (p != FINGERPRINT_OK) return -1;

    lcd.blink_off();
    lcd.clear();
    lcd.setCursor(1,0);

```

```
lcd.print("Digital Correta");  
lcd.setCursor(3,1);  
lcd.print("BEM VINDO");  
delay(500);  
  
//rotina de perifericos  
tone(autofalante,1000);  
delay(1000);  
noTone(autofalante);  
digitalWrite(rele,HIGH);  
delay(1000);  
digitalWrite(rele,LOW);  
  
for(int i = 0; i < 5; i++){ // Pisca o LED 5 vezes rapidamente  
digitalWrite(PINO_LED_VERDE, HIGH);  
delay(50);  
digitalWrite(PINO_LED_VERDE, LOW);  
delay(50);  
  
}  
  
delay(2000);  
lcd.clear();  
inicio();  
  
return finger.fingerID;  
  
}
```

8. REFERÊNCIA

<https://www.ferramentaskennedy.com.br/blog/o-que-e-um-torno-mecanico#:~:text=O%20torno%20mec%C3%A2nico%20%C3%A9%20um,mec%C3%A2nicos%20%C3%A9%20fixa%20no%20equipamento>

<https://blogmasterwalkershop.com.br/arduino/arduino-simulando-fechadura-controlada-por-senha>

<https://www.usinainfo.com.br/mini-fechadura-eletrica-solenoides-563>

<https://cepein.femanet.com.br/BDigital/arqTccs/1011320049.pdf>

<https://capsistema.com.br/index.php/2021/03/25/sistema-de-travamento-de-porta-baseado-em-senha-usando-arduino-com-teclado/>

