

# **Projeto de um sistema eletrônico de controle de acesso**

Arduino UNO

Ivander Gomes Ferreira Valim

Engenheiro de Telecomunicações - FT Unicamp

Projeto de um sistema embarcado para controle de acesso automatizado, integrando teclado matricial para autenticação por senha, display LCD para interação com o usuário e servomotor como atuador de abertura/fechamento da fechadura. Sensores de presença e LDR são utilizados para monitorar as condições do ambiente, permitindo ajuste dinâmico da iluminação quando necessário. O sistema foi desenvolvido com foco em segurança, usabilidade e automação inteligente.

## SUMÁRIO

<b>1. Objetivo</b>	<b>4</b>
<b>2. Introdução</b>	<b>4</b>
<b>3. Metodologia</b>	<b>4</b>
<b>a. Sistema eletrônico de controle de acesso</b>	<b>4</b>
<b>i. Desenvolvimento Lógico</b>	<b>4</b>
<b>ii. Código em linguagem C++</b>	<b>6</b>
<b>Discussão dos Resultados</b>	<b>9</b>
<b>4. Considerações</b>	<b>9</b>

## 1. OBJETIVO

Desenvolver um sistema de automação de fechadura integrado à iluminação do ambiente.

## 2. INTRODUÇÃO

Foi utilizado um servomotor como atuador para controlar a abertura e o fechamento da fechadura. A interface com o usuário foi implementada por meio de um display LCD e um teclado matricial 4x4, utilizado para inserção de senha. Além disso, foi integrado um sensor de presença em conjunto com um sensor LDR, permitindo a detecção de movimento e a medição da luminosidade do ambiente, possibilitando o ajuste automático da iluminação quando necessário.

## 3. METODOLOGIA

### a. Sistema eletrônico de controle de acesso

### MATERIAIS:

- 1 Arduino UNO.
- 1 Protoboard.
- 1 Potenciômetro.
- 1 Resistor de 220Ω.
- 1 Resistor de 10KΩ.
- 1 Lâmpada.
- 1 Fotorresistor (LDR).
- 1 Sensor de movimento (PIR).
- 1 LCD 16 x 2.
- 1 Servomotor.
- 1 Teclado matricial 4x4.

### CONFIGURAÇÕES DE PINAGEM:

**configuração de pinagem LCD:** pino GND <solo>, <pino VCC 5V <Potência>, pino VD no limpador do potenciômetro <Contraste>, pino RS na porta digital D13 <Seleção de Registro>, pino RW no GND <Leitura/Gravação>, pino E na porta analógica A1 <ativar>, pino D4 na porta analógica A2 <dado>, pino D5 na porta analógica A3 <dado>, pino D6 na porta analógica A4 <dado>, pino D7 na porta analógica A5 <dado>, pino LED ânodo no VCC 5V <potência leds>, pino LED catodo no GND com um resistor de 220Ω <solo leds>.

**configuração de pinagem Teclado matricial 4x4:** pino L1 <D9>, pino L2 <D8>, pino L3 <D7>, pino L4 <D6>, pino C1 <D5>, pino C2 <D4>, pino C3 <D3>, pino C4 <D2>.

**configuração de pinagem servomotor:** potência conectada ao 5V, o solo ao GND e o sinal a porta digital D10 do arduino uno.

**configuração de pinagem sensor PIR:** potência conectada ao 5V, o solo ao GND com um resistor de 10KΩ e o sinal a porta digital D11 do arduino uno.

**configuração de pinagem sensor LDR:** potência conectada ao 5V, o solo ao GND e o sinal a porta digital A0 do arduino uno.

**configuração de pinagem Lâmpada:**, o solo ao GND e o sinal a porta digital D12 do arduino uno.

**configuração de pinagem Potenciômetro:** potência conectada ao 5V, o solo ao GND e o sinal a porta VD do LCD.

### i. Desenvolvimento lógico do projeto:

O LCD 16 x 2 apresenta uma mensagem solicitando ao usuário para pressionar uma tecla qualquer para liberar o teclado matricial 4x4, que acompanhado da biblioteca <Keypad.h>, promove abstração de hardware, melhora a legibilidade do código, oculta as chamadas pinMode e digitalRead, fornecendo praticidade e interação para o usuário. [2]

Após pressionar uma tecla qualquer o sistema solicitará uma senha de 4 dígitos, apresentando uma mensagem de “aberto” se a senha digitada estiver correta, liberando acesso a fechadura, se a senha digitada estiver incorreta, não é liberado acesso a fechadura, apresentando uma mensagem de “fechado” no display através da biblioteca <LiquidCrystal.h> que permite a comunicação com telas alfanuméricas de cristal líquido (LCDs)[3]. A autenticidade da senha é verificada na EEPROM, através da biblioteca EEPROM.h, que permite ler e gravar dados eletricamente na memória EEPROM do arduino.[4]

Aplicamos um servomotor como atuador, para abrir e fechar a fechadura através da biblioteca <servo.h>, que concede o controle com precisão em graus da rotação de servomotores RC[1], incorporando um sensor de presença para identificar a existência de um usuário, integrado a um fotoresistor (LDR) que verifica se há ou não luminosidade no ambiente, para acender a lâmpada se houver presença de usuário sem iluminação e para não acender a lâmpada se houver presença de usuário com iluminação.

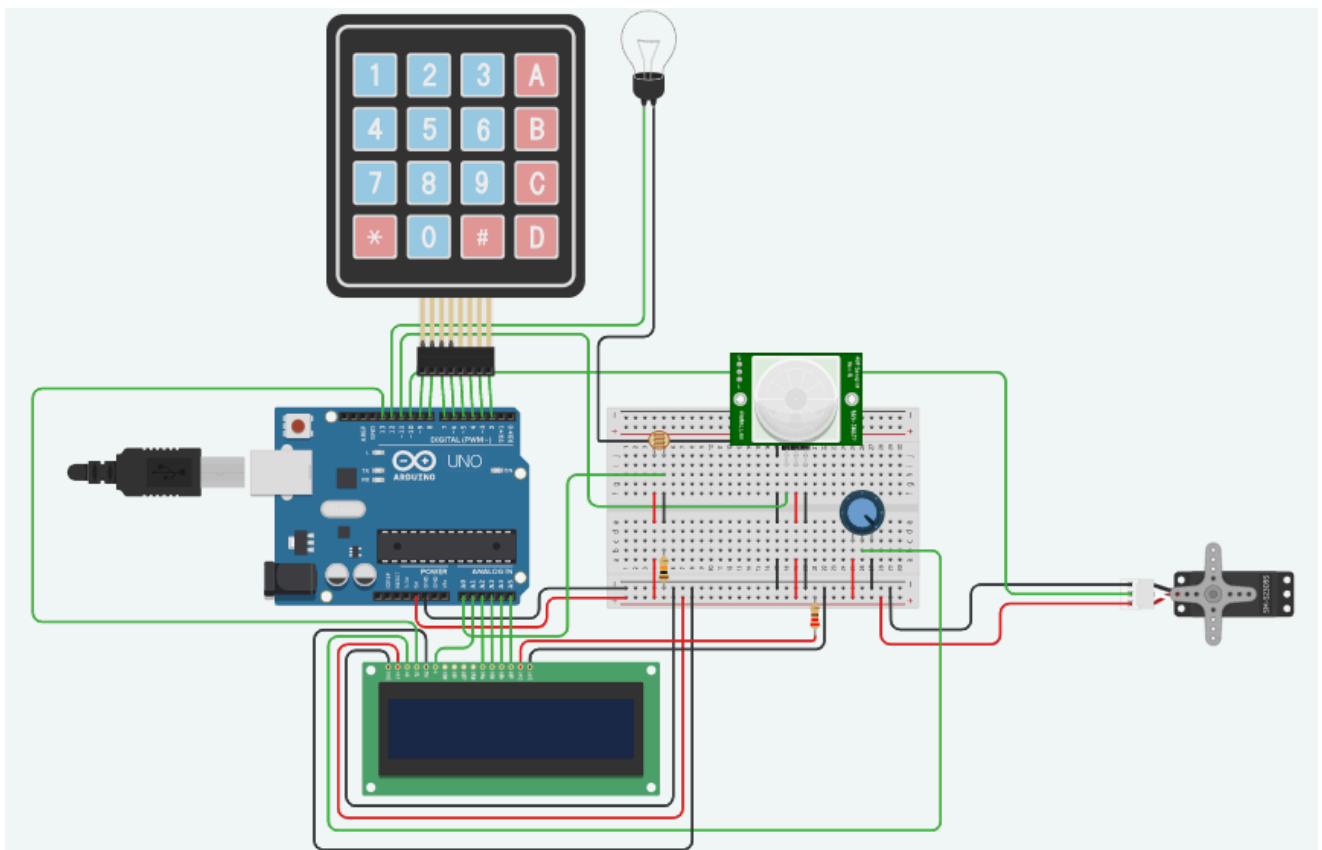


Figura 1 - Circuito automação de fechadura

## ii. Código em Linguagem C++:

```
#include <LiquidCrystal.h>           // inclusão da biblioteca do LCD.
#include <Servo.h>                   // inclusão da biblioteca do Servo motor.
#include <Keypad.h>                   // Biblioteca do código.
#include <EEPROM.h>                   // inclusão da biblioteca da memória externa.

// Declaração da variável senha.

int senha, senhaCorreta = 1234;

// Declaração dos Sensores

int pir      = 11;           //Nomeação do sensor PIR na porta digital D11.
int ldr      = A0;           //Nomeação do sensor LDR na porta analógica A0.

// Declaração das linhas e colunas do teclado matricial.

const byte linhas = 4;       //Declaração da quantidade de linhas.
const byte colunas = 4;      //Declaração da quantidade de colunas.

// Mapeamento da matriz do teclado.

char matriz_teclado[linhas][colunas] =
{
    {'1','2','3','A'},
    {'4','5','6','B'},
    {'7','8','9','C'},
    {'*','0','#','D'}
};

// Nomeação da pinagem do teclado matricial

byte Pinoslinhas[linhas] = {9, 8, 7, 6}; //Nomeação dos pinos linhas do teclado.
byte Pinoscolunas[colunas] = {5, 4, 3, 2}; //Nomeação dos pinos colunas do teclado.

// Instanciando um objeto teclado que define mapeamento, pinagem, linhas e colunas.

Keypad teclado = Keypad(makeKeymap(matriz_teclado), Pinoslinhas, Pinoscolunas, linhas, colunas);

// Declaração dos atuadores

int lampada = 12;           //Nomeação da lâmpada na porta digital D12.
Servo fechadura;           //Instanciando um objeto fechadura no servomotor.
LiquidCrystal lcd(13,A1,A2,A3,A4,A5); //Declaração de pinagem do LCD.
```

```

void setup()
{

    pinMode(lampada,OUTPUT);    // Lâmpada é nomeada como saída.
    pinMode(pir,INPUT);        // Sensor de presença PIR é nomeado como entrada.
    pinMode(A0,INPUT);         // Sensor de luminosidade LDR é nomeado como entrada.
    lcd.begin(16,2);           // Inicialização do LCD 16 x 2.
    fechadura.attach(10);      // Anexa a variável servo a porta digital D10.
    teclado.setHoldTime(1000); // Estabelece 1s para a espera de uma nova tecla.
    EEPROM.put(0, senhaCorreta); // Insere a senha na posição 0 da memória EEPROM.
    Serial.begin(9600);

}

void loop()
{
    int senhaCadastrada = EEPROM.get(0,senhaCorreta); // Lê na memória EEPROM a senha cadastrada.

    ldr          = analogRead(A0);          // Verifica luminosidade do ambiente.
    bool presenca = sensor Presenca();      // Verifica presença no ambiente.

    //Escreve "PRESSIONE QUALQUER TECLA" no LCD.
    lcd.setCursor(0,0);          /
    lcd.print("  PRESSIONE  ");
    lcd.setCursor(0,1);
    lcd.print(" QUALQUER TECLA");

    //Se houver presença de usuário e a luminosidade for menor igual 900 a lâmpada acende
    if ( presenca == true && ldr <= 900 )
    {

        digitalWrite(lampada, HIGH);

    }else // Se não ela não acende
    {

        digitalWrite(lampada, LOW);

    }

    char tecla_preSSIONada = teclado.getKey();    // Verifica se o teclado foi pressionado

    if( tecla_preSSIONada) //Se alguma tecla for pressionada solicita para digitar uma senha de 4 dígitos
    {

        senha = solicitacaoSenha();
    }
}

```

```

if ( senha == senhaCadastrada )    // Se senha for correta abre a porta e printa ABERTO no LCD.
{

    senha = 0;
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("  ABERTO  ");
    abrirPorta();
    delay(1000);

} else{                             // Se senha for incorreta printa FECHADO no LCD.

    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("  FECHADO  ");
    delay(1000);

    }

}

```

```

//Solicita ao usuário para digitar uma senha no teclado e retorna seu valor em inteiro.
int solicitacaoSenha(void)
{

    int senhaDigitada, i = 0;        // Declara variável local senhaDigitada como inteiro.
    String palavraChave;             // Declara variável local palavraChave como string.

    // Escreve no LCD "Digite a senha:".
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Digite a senha:");
    lcd.setCursor(0,1);

    // Solicita para o usuário digitar uma string de 4 dígitos no teclado e a converte em inteiro.
    while ( i < 4 )
    {
        char    tecla_preSSIONada = teclado.getKey();

        if(tecla_preSSIONada)
        {
            lcd.print("*");
            palavraChave += tecla_preSSIONada;
            senhaDigitada = palavraChave.toInt();
            i++;
        }
    }
}

```

```

    }

}

//retorna senha de 4 dígitos inteiro.
return (senhaDigitada);

}

//Verifica se há presença no ambiente e retorna se é verdadeiro ou falso.
bool sensorPresenca(void)
{

bool presenca;

if (digitalRead(pir) == HIGH)
{
    presenca = true;
} else {
    presenca = false;
}

return (presenca);
}

//Rotaciona a tranca (servomotor) 180° e o retorna a posição inicial.
void abrirPorta()
{
    for (int i = 0; i<180; i++)
    {
        fechadura.write(i);
        delay(10);
    }
    for (int i = 180; i>0; i--)
    {
        fechadura.write(i);
        delay(10);
    }
}

```



#### 4. DISCUSSÃO DOS RESULTADOS

Utilizamos uma placa arduino uno e uma protoboard conectadas ao 5V e ao GND, para construir um sistema de automação de fechadura, o LCD apresenta uma mensagem solicitando para o usuário pressionar uma tecla qualquer, para liberar o teclado matricial 4x4 para digitar uma senha de 4 dígitos; incorporamos um servomotor para abrir e fechar a tranca, um sensor de presença para identificar a existência de um usuário, integrado a um fotoresistor (LDR) que verifica se há ou não luminosidade no ambiente, para não acender a lâmpada se houver iluminação e para acender a lâmpada se não houver iluminação.

#### CONSIDERAÇÕES

Fazendo uso da simplicidade da linguagem C++, utilizou-se 3 sensores e 2 atuadores para desenvolver um sistema de automação de fechadura que permite controlar o acesso a uma porta com praticidade, visibilidade, segurança, precisão e interação com o usuário.

#### REFERÊNCIAS

- [1] Biblioteca Servo.h. **Arduino**, <https://www.arduino.cc/reference/en/libraries/servo/>, 26 de junho de 2021.
- [2] Biblioteca Keypad.h. **Arduino**, <https://www.arduino.cc/reference/en/libraries/keypad/>, 26 de junho de 2021.
- [3] Biblioteca LiquidCrystal.h. **Arduino**, <https://www.arduino.cc/reference/en/libraries/liquidcrystal/>, 26 de junho de 2021.
- [4] Biblioteca EEPROM.h. **Arduino**, <https://www.arduino.cc/reference/en/libraries/EEPROM/>, 26 de junho de 2021.