

CÓDIGO DO CIRCUITO FECHADURA COM SENHA

Feito no tinkercad

```
#include <Keypad.h>
```

```
#include <Servo.h>
```

```
Servo servoMotor;
```

```
char password[] = "321";  
int position = 0;
```

```
//informando ao arduino quantas linhas e colunas tem o keypad
```

```
const byte ROWS = 4;
```

```
const byte COLS = 4;
```

```
//informando ao arduino o que tem no teclado
```

```
char keys [ROWS] [COLS] = {
```

```
    {'1','2','3','A'},
```

```
    {'4','5','6','B'},
```

```
    {'7','8','9','C'},
```

```
    {'*','0','#','D'}
```

```
};
```

```

// arduino vai identificar onde os pinos form ligados
//pinos das colunas e das linhas
byte rowPins[ROWS] = {9,8,7,6};
byte colPins [COLS] = {5,4,3,2};

//as variáveis receberão o valor da leitura dos pinos das linhas e das colunas
Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

//variáveis do led's
const int ledvermelho =12;
const int ledverde =11;

void setup() {
  pinMode (ledvermelho, OUTPUT);
  pinMode (ledverde, OUTPUT);

  servoMotor.attach(13);
  setLocked (true);
}

//Guarda as informações
void loop(){
  char key = keypad.getKey();
  if (key) {

    if (key == password[position]) {
      position++;

      if (position == 3){
setLocked (false);
      delay(100);
    }
    } else {
      position = 0;
    }
  }
}

//função de comando
void setLocked(bool Locked) {
  if (Locked) {
    digitalWrite(ledvermelho, HIGH);
  }
}

```

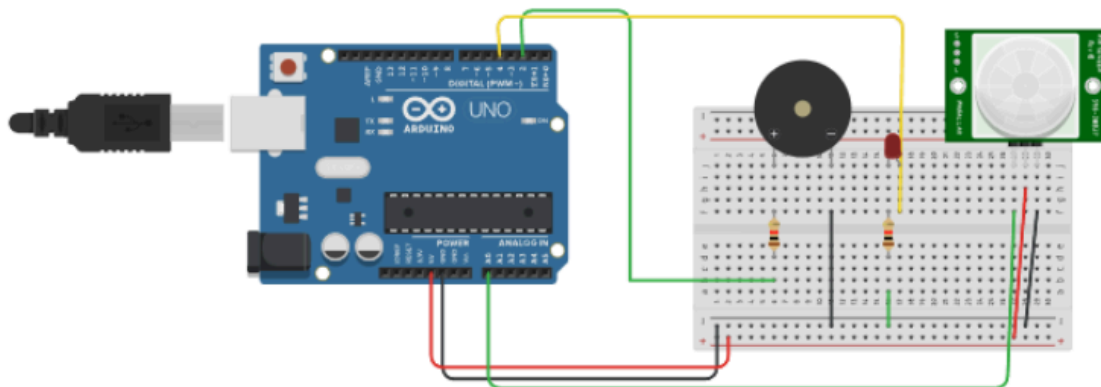
```

digitalWrite (ledverde, LOW);
servoMotor.write(0);
} else {
digitalWrite (ledverde, HIGH);
digitalWrite (ledvermelho, LOW);
servoMotor.write(82);
}
}

```

Sua finalidade é um Controle à Área de Restrito: Limitar o acesso a áreas como salas de servidores, laboratórios ou salas de arquivos confidenciais apenas a pessoal autorizado.

▶ Simular



CÓDIGO DO SISTEMA DE ALARME COM SENSOR DE MOVIMENTO E ALERTA SONORO

Feito no tinkercad

```

int time = 0;
void setup()
{

```

```

    Serial.begin(9600);
}

void loop()
{
    int valorRecebido = analogRead(A0);

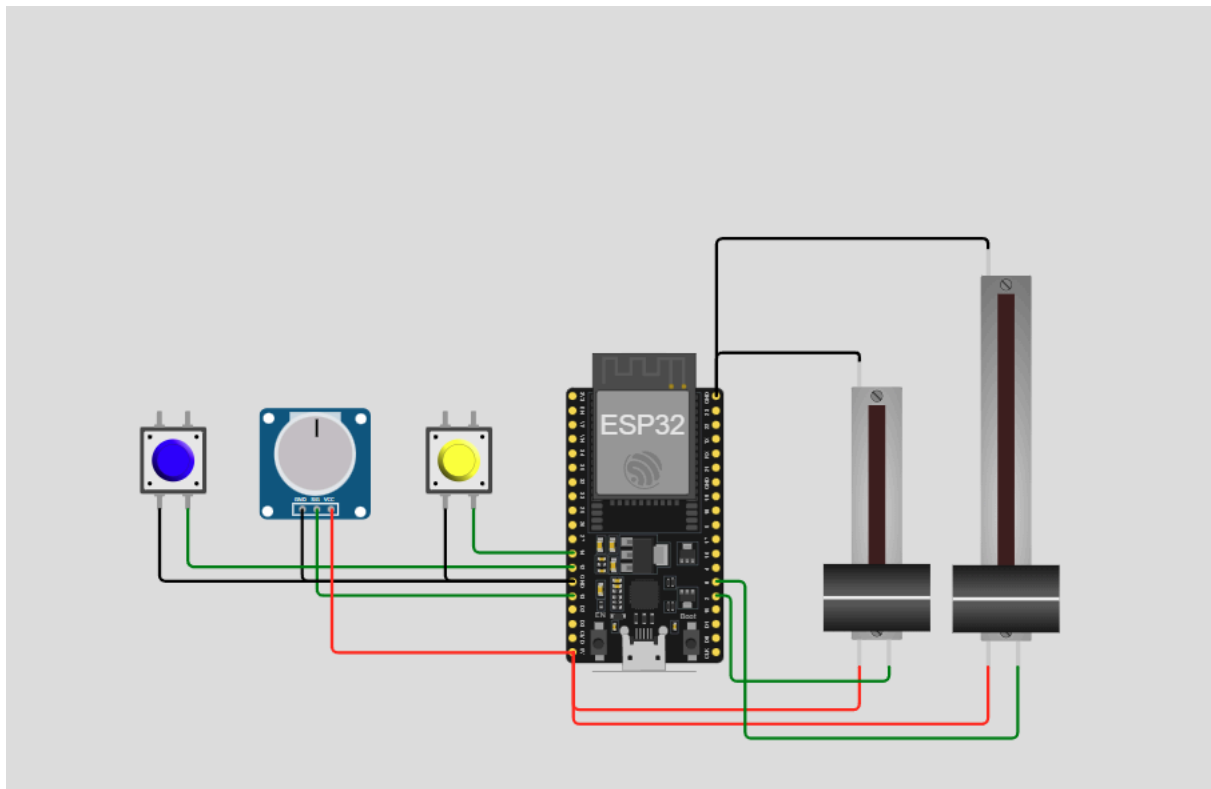
    if (valorRecebido > 0) {
        while (time <= 5000){
            digitalWrite(4, HIGH);
            tone(2, 1000);
            delay(500);
            digitalWrite(4, LOW);
            noTone(2);
            delay(500);
            time += 1000;
        }

        time = 0;

    }
    Serial.println(valorRecebido);
}

```

Com a finalidade de Segurança Perimetral: Monitorar o perímetro de uma propriedade para detectar movimentos suspeitos antes de uma possível invasão.



CÓDIGO CARRO FÓRMULA UM

Feito no wokwi

```
int steeringWheel;
```

```
int x;
```

```
int gasPedal;
```

```
int breakPedal;
```

```
int upShift;
```

```
int downShift;
```

```
int gear = 0;
```

```
void setup() {
```

```
    // put your setup code here, to run once:
```

```
Serial.begin(115200);  
pinMode(13, INPUT);  
pinMode(0, INPUT);  
pinMode(2, INPUT);  
pinMode(12, INPUT_PULLUP);  
pinMode(14, INPUT_PULLUP);}
```

```
void loop() {  
  //Steering Wheel  
  steeringWheel = analogRead(13);  
  steeringWheel = map(steeringWheel, 0, 4095, -135, 135);  
  x = steeringWheel * -1;  
  if(steeringWheel>0){  
    Serial.print(steeringWheel);  
    Serial.print("° Right");  
  }  
  else if(steeringWheel<0){  
    Serial.print(x);  
    Serial.print("° Left");  
  }  
  else{  
    Serial.print("Straight");  
  }  
  Serial.print("\n");  
  
  //Gas Pedal  
  gasPedal = analogRead(0);  
  gasPedal = map(gasPedal, 0, 4095, 0, 100);
```

```
Serial.print(gasPedal);
```

```
Serial.print("% Throttle \n");
```

```
//Break Pedal
```

```
breakPedal = analogRead(2);
```

```
breakPedal = map(breakPedal, 0, 4095, 0, 100);
```

```
Serial.print(breakPedal);
```

```
Serial.print("% Break \n");
```

```
//Gear
```

```
upShift = digitalRead(14);
```

```
downShift = digitalRead(12);
```

```
if (gear>0 && gear<8){
```

```
    if(upShift == 0){
```

```
        gear++;
```

```
    }
```

```
    if(downShift == 0){
```

```
        gear--;
```

```
    }
```

```
}
```

```
else if(gear == 0){
```

```
    if(upShift == 0){
```

```
        gear++;
```

```
    }
```

```
}
```

```
else if(gear == 8){
```

```
    if(downShift == 0){
```

```
        gear--;
```

```

    }
}

if(gear == 0){
    Serial.print("N");
}
else {
    Serial.print(gear);
    if(gear == 1){
        Serial.print("st");
    }
    else if(gear == 2){
        Serial.print("nd");
    }
    else if(gear == 3){
        Serial.print("rd");
    }
    else{
        Serial.print("th");
    }
    Serial.print(" Gear");
}
Serial.print("\n");

```

Sua finalidade é de Educação e Aprendizado: Pode ser usado para ensinar sobre eletrônica e programação com microcontroladores e sensores, demonstrando como componentes interagem entre si. Servindo também para Design e Prototipagem para jogos sobre corrida de carros e ou projeto de Hobby e Entretenimento.

Oferecendo uma atividade divertida e desafiadora para entusiastas de automobilísticos e eletrônica, combinando interesses com aprendizado técnico.

```
delay(100); // this speeds up the simulation  
}
```