



2023

### 3. Detetar o intruso

Número do projeto: **2021-1-FR01-KA220-SCH-000031617**



 **Co-funded by  
the European Union**

O apoio da Comissão Europeia à produção desta publicação não constitui um endosso do conteúdo, que reflete apenas as opiniões dos autores, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito das informações nele contidas.

Parceria SCRAPY  
31/05/2023



## Índice

Experiência 3: Detetar o intruso.....	2
<b>Objetivos</b> .....	2
<b>Materiais a utilizar</b> .....	2
<b>Passos a seguir</b> .....	3
<b>Diagrama de ligação</b> .....	4
<b>Código</b> .....	4

## Experiência 3: Detetar o intruso

### Breve Descrição

Com esta experiência, os alunos serão capazes de criar um alarme de movimento para informá-los quando alguém estiver por perto.

### Descrição Estendida

Nesta atividade vamos usar um sensor chamado HC-SR501 Sensor de Detecção de Movimento. Este sensor deteta o movimento medindo as mudanças nos níveis infravermelhos (calor) emitidos pelos objetos circundantes. Quando o movimento é detetado, o sensor PIR emite um sinal alto em seu pino de saída. HC-SR501 PIR Sensor tem um atraso ajustável antes de disparar e sensibilidade ajustável.

Vamos usar este sensor juntamente com um buzzer, para nos informar que alguém está perto e usar a sensibilidade ajustável do sensor para melhor atender às necessidades em termos de distância.

### Objetivos

Com esta atividade, os alunos irão experimentar sensores, buzzers e pensar em forma de ajustar o sensor às suas necessidades e objetivo da atividade.

Em termos de conhecimentos, os alunos irão:

1. Entenda como um sensor de movimento PIR deteta o movimento usando ondas de calor.
2. Explique por que um sensor de movimento "viu" ou não um objeto em movimento relacionado com a distância que esse objeto está do sensor.
3. Trabalhe e envolva-se com a computação física e uma linguagem de codificação.
4. Encontre soluções e resolva problemas durante a fase de programação e teste.

### Materiais a utilizar

- 1 x Raspberry Pi Pico
- 1 x kit de breadboard Pico
- 1 x Breadboard de tamanho normal
- 1 x Sensor Detetor de Movimento PIR HC-SR501
- 1 x LED Vermelho
- 1 x campainha
- 1 x resistor de 220 ohm
- Fios de jumper

## Passos a seguir

Vamos criar um alarme de movimento que nos avisará se alguém chegar perto de onde estamos!

Para isso, vamos começar por

1. Conecte o sensor do detetor de movimento PIR HC-SR501 à placa Raspberry Pi Pico usando fios de conexão.
2. Conecte o buzzer e os LEDs à placa Raspberry Pi Pico usando fios de conexão e as resistências de 220 ohm para limitar o fluxo de corrente.
3. Escreva um programa Python para controlar a placa Raspberry Pi Pico e use o PIR Motion Detetor Sensor HC-SR501 para detetar movimento.
4. Programe a placa Raspberry Pi Pico para ligar o LED vermelho e tocar a campainha quando um obstáculo passar pelo sensor.
5. Teste o sistema de alarme movendo diferentes objetos na frente do sensor e certifique-se de que os LEDs e o buzzer fornecem o feedback apropriado.

### Placa Raspberry Pi Pico:

- GP28: Pino de gatilho do sensor do detetor de movimento PIR HC-SR501
- GP15: Pino positivo do LED Vermelho
- GP14: Pino positivo da campainha
- GND: Pino de terra da placa

### Sensor Detetor de Movimento PIR HC-SR501

- VCC: Conecte-se à fonte de alimentação de 5V
- GND: Conecte-se ao GND da placa Raspberry Pi Pico
- Trig: Conecte-se ao GP28 da placa Raspberry Pi Pico

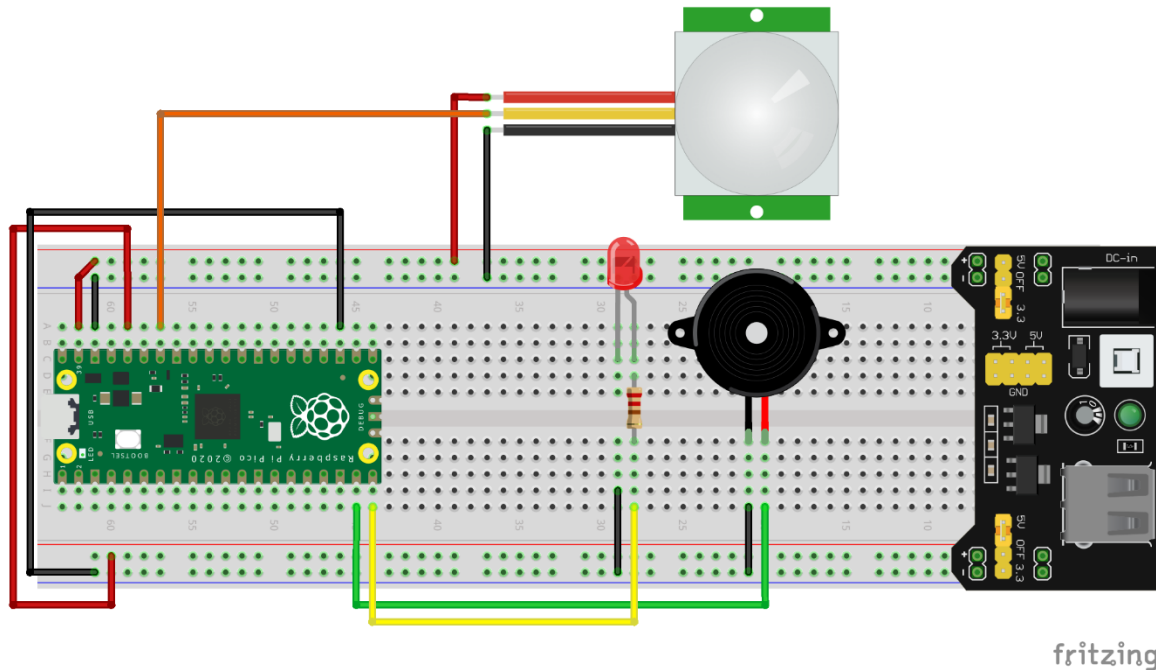
### LED vermelho:

- Perna positiva: Conecte-se ao GP15 da placa Raspberry Pi Pico através de uma resistência de 220 ohm
- Perna negativa: Conecte-se ao GND da placa Raspberry Pi Pico

### Buzzer:

- Perna positiva: Conecte-se ao GP14 da placa Raspberry Pi Pico
- Perna negativa: Conecte-se ao GND da placa Raspberry Pi Pico

## Diagrama de ligação



## Código

```
from machine import Pin
from time import sleep

sensor_pir = Pin(28, Pin.IN, Pin.PULL_UP)
led = Pin(15, Pin.OUT)
buzzer = Pin(14, Pin.OUT)

while True:
    if sensor_pir.value() == 1:
        print("ALARM! Motion detected!")
        led.toggle()
        buzzer.toggle()
        sleep(1)
```