# [Programa] Interruptor

## Objetivo:

Utilizar o *ESP32-C6-DevKitM-1* em conjugação com um interruptor e monitorizar estados.

## Lista de material:

- *ESP32-C6-DevKitM-1*
- Breadboard
- Cabo USB C
- Interruptor (tipo botão)
- Fios de ligação

material_list.png

## Contextualização:

Um interruptor é um componente elétrico que pode desconectar ou conectar o caminho condutor de um circuito elétrico, interrompendo a corrente elétrica ou conduzindo-a de um terminal para outro [1].

### Interruptor

Um interruptor tipo botão tem tipicamente quatro terminais que estão conectados internamente aos pares. Por isso, numa utilização típica, apenas dois terminais, que não ligados internamente, são usados, como se ilustra na imagem seguinte:

| | | | |
|---|---|---|---|
| switch_a.png | switch_b.png | switch_c.png | switch_d.png |

Ao ser pressionado o interruptor alterna entre os estados conetado e não conetado:

| | |
|---|---|
| switch_e.png | switch_e.png |
| (pressionado) | (não pressionado) |
| switch_f.png | switch_g.png |
| (fechado) | (aberto) |

## Procedimento:

Escolhendo o terminal 23 e o GND do ESP32 como referência deve considerar-se o seguinte circuito:

interruptor_esq.png

## Instruções:

- Montar o circuito esquematizado anteriormente
- Ligar a placa ESP32 ao computador por via de cabo USB micro
- Abrir o IDE Arduino
- Selecionar o dispositivo ESP32 adequado ( *ESP32C6 Dev Module* )
- Copiar o seguinte código

```
/*
interruptor tipo botão
*/

#define BUTTON_PIN 23  // definição de terminal GPIO23 ligado ao interruptor

// Variáveis
int lastState = LOW;  // o estado anterior do terminal de entrada
int currentState;     // o estado atual do terminal de entrada

void setup() {
  // inicializa a comunicação serial a 115200 bits por segundo:
  Serial.begin(115200);
  // inicializa o terminal do interruptor tipo botão:
  // (o terminal de entrada toma o valor de HIGH quando o interruptor estiver aberto,
  // e LOW quando estiver fechado)
  pinMode(BUTTON_PIN, INPUT_PULLUP);
}

void loop() {
  // leitura do estado do interruptor tipo botão:
  currentState = digitalRead(BUTTON_PIN);

  if (lastState == HIGH && currentState == LOW) {
    Serial.println("o interruptor foi pressionado");
    digitalWrite(RGB_BUILTIN, HIGH);
  }
```

```
  else if (lastState == LOW && currentState == HIGH) {
    Serial.println("o interruptor foi solto");
    digitalWrite(RGB_BUILTIN, LOW);
  }

  // guarda o último estado
  lastState = currentState;
}
```

- Compilar e carregar o código para a placa ESP32
- Verificar o resultado no monitor de comunicação serial do IDE Arduino

switch_f.png

# Referências

[1] Wikipedia. Switch. url: https://en.wikipedia.org/wiki/Switch (acedido em 22/08/2023).

---

Revision #8
Created 2024-08-28 13:52:26 UTC by João Pedro Monteiro
Updated 2024-09-06 09:32:48 UTC by João Pedro Monteiro