[Programa] Interruptor

Objetivo:

Utilizar o ESP32-WROOM-32E DevKitC v4 [1] em conjugação com um interruptor e monitorizar estados.

Lista de material:

- *ESP32-WROOM-32E DevKitC v4*
- Breadboards
- Cabo USB micro
- Interruptor (tipo botão)

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 [3].

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 22 e o GND do ESP32 como referência deve considerar-se o seguinte circuito:

interruptor_esq.png

Instruções:

(para uma primeira utilização do ESP32, sugere-se o tutorial *Primeiro Programa* [2])

- Montar o circuito esquematizado anteriormente (ver Secção **Montagem exemplificativa** , para mais detalhe)
- Ligar a placa ESP32 ao computador por via de cabo USB micro
- Abrir o IDE Arduino
- Selecionar o dispositivo ESP32 adequado
- Copiar o seguinte código

```
/*
interruptor□tipo botão
*/□

#define BUTTON_PIN 22  // definição de terminal GIOP22 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 série 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");
  else if (lastState == LOW && currentState == HIGH)
```

```
    Serial.println("o interruptor foi solto");

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

- Compilar e carregar o código para a placa ESP32

arduino_upload.png

- Verificar o resultado no monitor de comunicação série do IDE Arduino

eg_int_s.png	

Montagem exemplificativa:

board_kit_a.png	

Referências

[1] Ltd. Espressif Systems Co. ESP32-DevKitC V4 Getting Started Guide. url: https://docs.espressif.com/projects/esp-idf/en/latest/esp32/hw-reference/esp32/get-started-devkitc.html (acedido em 18/08/2023).

[2] Paulo Menezes. Artigos de Suporte - DEEC - Kit Eletrónica - Primeiro Programa. url: https://kb.deec.uc.pt/books/deec/page/primeiro-programa (acedido em 17/08/2023).

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

Revision #5
Created 2023-08-24 17:57:10 UTC by João Pedro Monteiro
Updated 2024-08-27 16:06:42 UTC by João Pedro Monteiro