[Programa] LED e interruptor

Objetivo:

Utilizar o ESP32-WROOM-32E DevKitC v4 [1] para alimentar um LED em conjugação com um interruptor para alternar entre os estados ligado e desligado do LED.

Lista de material:

- *ESP32-WROOM-32E DevKitC v4*
- Breadboards
- Resistência de 330 ohm (Ω)
- Interruptor (tipo botão)
- Díodo emissor de luz (LED) vermelho
- Fio de ligação
- Cabo USB micro

material_all.png

Contextualização:

Em eletrónica, um circuito LED é um circuito usado para alimentar um díodo emissor de luz (LED). O circuito deve fornecer corrente suficiente para acender o LED e evitar danos ao mesmo. O circuito mais simples para acionar um LED é através de uma resistência em série [3]. Por outro lado, um interruptor é um componente elétrico que pode desconectar ou conectar o caminho condutor de um circuito elétrico, interrompendo a corrente elétrica ou desviando-a de um condutor para outro [4].

Procedimento:

Atentando no objetivo enunciado, e escolhendo os terminais 21 (para o LED), 22 (para o interruptor) e o GND do ESP32 como referência deve considerar-se o seguinte circuito:

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

```
/*
tipo botão LED
*/

#define BUTTON_PIN 22  // Terminal GPIO22 do ESP32 que liga ao interruptor
#define LED_PIN    21  // Terminal GPIO21 do ESP32 que liga ao led

// variáveis:
int led_state = LOW;    // o estado atual do LED
int button_state;       // o estado atual do interruptor
int last_button_state;  // o estado anterior do interruptor

void setup() {
  Serial.begin(115200);              // inicializa comunicação série
  pinMode(BUTTON_PIN, INPUT_PULLUP); // estabelece o modo do terminal do ESP32
  pinMode(LED_PIN, OUTPUT);          // estabelece o modo do terminal do ESP32

  button_state = digitalRead(BUTTON_PIN);
}

void loop() {
  last_button_state = button_state;       // guarda o estado anterior
  button_state = digitalRead(BUTTON_PIN); // lê o novo estado

  if (last_button_state == HIGH && button_state == LOW) {
    Serial.println("o interruptor foi pressionado");

    // modifica o estado do LED
    led_state = !led_state;

    // controla o LED de acordo com o estado do interruptor
    digitalWrite(LED_PIN, led_state);
  }
}
```

- Compilar e carregar o código para a placa ESP32
- Verificar o resultado

Montagem exemplificativa:

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. LED circuit. url: https://en.wikipedia.org/wiki/LED_circuit (acedido em 18/08/2023).

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

Revision #4
Created 2023-08-24 18:34:01 UTC by João Pedro Monteiro
Updated 2024-08-27 16:07:06 UTC by João Pedro Monteiro