

# PRACTICA 2 : INTERRUPTCIONES PARTE B

## Código Practica B:

```
#include <Arduino.h>

volatile int interruptCounter;
int totalInterruptCounter;

hw_timer_t * timer = NULL;
portMUX_TYPE timerMux = portMUX_INITIALIZER_UNLOCKED;

void IRAM_ATTR onTimer() {
    portENTER_CRITICAL_ISR(&timerMux);
    interruptCounter++;
    portEXIT_CRITICAL_ISR(&timerMux);
}

void setup() {

    Serial.begin(115200);

    timer = timerBegin(0, 80, true);
    timerAttachInterrupt(timer, &onTimer, true);
    timerAlarmWrite(timer, 1000000, true);
    timerAlarmEnable(timer);
}

void loop() {

    if (interruptCounter > 0) {

        portENTER_CRITICAL(&timerMux);
        interruptCounter--;
        portEXIT_CRITICAL(&timerMux);

        totalInterruptCounter++;

        Serial.print("An interrupt as occurred. Total number: ");
        Serial.println(totalInterruptCounter);

    }
}
```

## **Explicación:**

### **Practica B:**

Esta práctica al igual que la A también se divide por partes, en este caso se ven dos claramente marcados, el setup y el loop.

En el setup comenzamos con la instrucción que nos permitirá mostrar las dadas `Serial.begin(115200)`. Las siguientes son de declaración del temporizador (`timerAttachInterrupt(timer, &onTimer, true);` ) y de lo que ha de hacer (`timerAlarmWrite(timer, 1000000, true);` ) y cuando lo ha de hacer (`timerAlarmEnable(timer);` ). La instrucción (`timerAlarmWrite(timer, 1000000, true);` ) el 1000000 nos indica el tiempo entre interrupciones.

En el loop podemos apreciar que se ha realizará siempre que `interruptCounter` sea mayor que 0 se imprimirá por pantalla que hay una interrupción, y el número que es, gracias a las siguientes instrucciones `Serial.print("An interrupt as occurred. Total number: ");`  
`Serial.println(totalInterruptCounter);`. El número se va actualizando por esta instrucción `totalInterruptCounter++;`.

Para resumir el funcionamiento de la práctica primero se crea el temporizador y luego con el loop se realizan todas las interrupciones y se muestran por pantalla.

### **Funcionamiento de la práctica:**

#### **Practica B**

Comprobación interrupción:

PLATFORMIO

PROJECT TASKS

esp32dev

General

Build

Upload

Monitor

Upload and Monitor

Clean

Platform

Build Filesystem Image

Program Size

Upload Filesystem Image

Upload Filesystem Image OTA

Erase Flash

Advanced

Remote Development

QUICK ACCESS

PIO Home

Open

PIO Account

Inspect

Projects & Configuration

Libraries

Boards

Platforms

Devices

Debug

Pol-Naharro-informe-P2.MD

PIO Home

P2-1-Pol-Naharro

Pol-Naharro-informe-P2.MD

## PRACTICA 2 : INTERRUPTIONES

45

46

47

48

49

50

51

52

53

54

55

56

45

46

47

48

49

50

51

52

53

54

55

56

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

8: Task - PlatformIO: Monitor

> Executing task in folder P2-2-Pol-Naharro: C:\Users\Pol\.platformio\penv\Scripts\platformio.exe device monitor --environment esp32dev <

---

Available filters and text transformations: colorize, debug, default, direct, esp32\_exception\_decoder, hexlify, log2file, nocontrol, printable, send\_on\_enter, time

More details at <http://bit.ly/pio-monitor-filters>

Miniterm on COM3 115200,8,N,1 ---

Quit: Ctrl+C | Menu: Ctrl+T | Help: Ctrl+H ---

An interrupt as occurred. Total number: 1

An interrupt as occurred. Total number: 2

An interrupt as occurred. Total number: 3

An interrupt as occurred. Total number: 4

An interrupt as occurred. Total number: 5

An interrupt as occurred. Total number: 6

A

Ln 3, Col 1

Spaces: 2

UTF-8

CRLF

Markdown

Escribe aquí para buscar

16:32

01/04/2021