

ĐẠI HỌC BÁCH KHOA TP. HỒ CHÍ MINH
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



BÁO CÁO HỆ THỐNG NHÚNG

Bài thực hành số 5

Lương Hữu Phú Lợi - 1911545
Cao Thanh Lương - 1914076
Nguyễn Văn Việt - 1912436
Huỳnh Ngọc Bảo Trân - 1912269

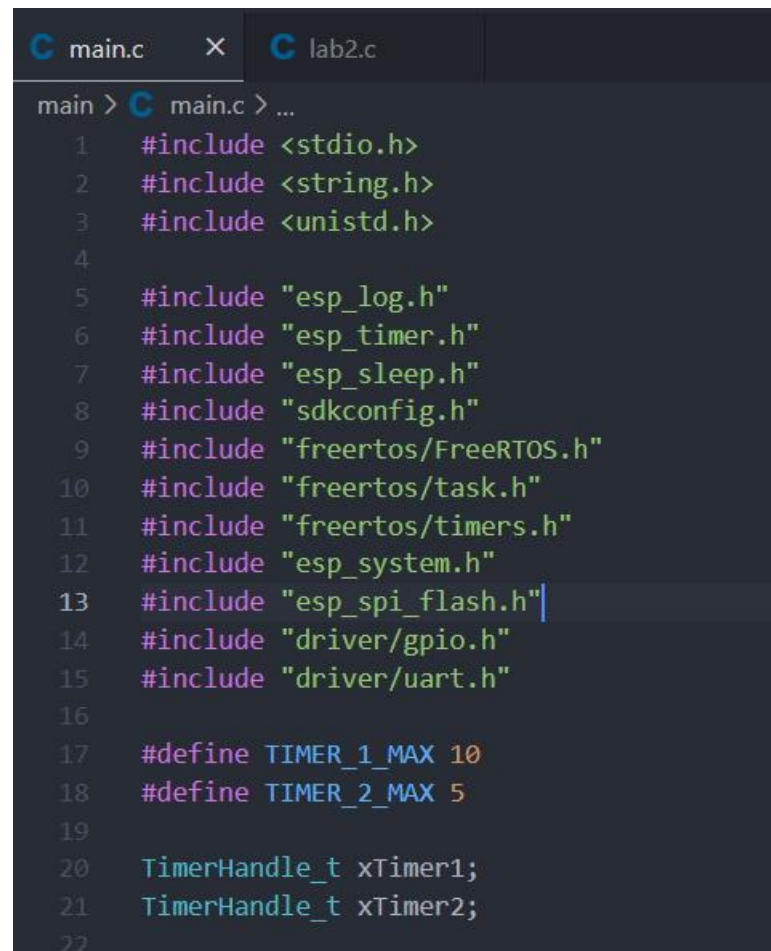
Họ và tên	MSSV	Đóng góp
Lương Hữu Phú Lợi	1911545	100%
Cao Thanh Lương	1914076	100%
Nguyễn Văn Việt	1912436	100%
Huỳnh Ngọc Bảo Trân	1912269	100%

GitHub:

[PhuLoi-1911545/school-LAB-embedded-ESP-IDF \(github.com\)](https://github.com/PhuLoi-1911545/school-LAB-embedded-ESP-IDF)

1. Thực hiện

Tạo một Project ESP_IDF, sau đó code file main với nội dung như sau



```
main > C main.c > ...
1  #include <stdio.h>
2  #include <string.h>
3  #include <unistd.h>
4
5  #include "esp_log.h"
6  #include "esp_timer.h"
7  #include "esp_sleep.h"
8  #include "sdkconfig.h"
9  #include "freertos/FreeRTOS.h"
10 #include "freertos/task.h"
11 #include "freertos/timers.h"
12 #include "esp_system.h"
13 #include "esp_spi_flash.h"
14 #include "driver/gpio.h"
15 #include "driver/uart.h"
16
17 #define TIMER_1_MAX 10
18 #define TIMER_2_MAX 5
19
20 TimerHandle_t xTimer1;
21 TimerHandle_t xTimer2;
22
```

```

19
20 TimerHandle_t xTimer1;
21 TimerHandle_t xTimer2;
22
23
24 void periodic_timer_callback(TimerHandle_t arg) {
25     const char *name = pcTimerGetTimerName(arg);
26     uint32_t count = (uint32_t)pvTimerGetTimerID(arg);
27     count ++;
28
29     if(strcmp(name,"Timer 1")) {
30         if(count > TIMER_1_MAX) {
31             printf("STOP Timer 1\n");
32             xTimerStop(arg,0);
33         }
34         else {
35             printf("ahihi - Nhom 3 L01 - HCMUT K19\n");
36             vTimerSetTimerID(arg,(void*)count);
37         }
38     }
39     else if (strcmp(name,"Timer 2")) {
40         if(count > TIMER_2_MAX) {
41             printf("STOP Timer 2\n");
42             xTimerStop(arg,0);
43         }
44         else {
45             printf("ihaha - Nhom 3 L01 - HCMUT K19\n");
46             vTimerSetTimerID(arg,(void*)count);
47         }
48     }
49 }
50
51
52 void app_main(void)
53 {
54
55     xTimer1 = xTimerCreate(
56         // NULL,          /* Task function. */
57         "Timer 1",        /* String with name of task. */
58         pdMS_TO_TICKS(2000), /* Stack size in bytes. */
59         pdTRUE,           /* Parameter passed as input of the task */

```

```
43     }
44     else {
45         printf("Timer 2 - Nhóm 3 L01 - HCMUT K19\n");
46         vTimerSetTimerID(arg, (void*)count);
47     }
48 }
49 }
50
51
52 void app_main(void)
53 {
54
55     xTimer1 = xTimerCreate(
56         // NULL,          /* Task function. */
57         "Timer 1",        /* String with name of task. */
58         pdMS_TO_TICKS(2000), /* Stack size in bytes. */
59         pdTRUE,           /* Parameter passed as input of the task */
60         0,                /* Priority of the task. */
61         periodic_timer_callback /* Task handle. */
62     );
63
64     xTimer2 = xTimerCreate(
65         // NULL,          /* Task function. */
66         "Timer 2",        /* String with name of task. */
67         pdMS_TO_TICKS(3000), /* Stack size in bytes. */
68         pdTRUE,           /* Parameter passed as input of the task */
69         0,                /* Priority of the task. */
70         periodic_timer_callback /* Task handle. */
71     );
72
73     xTimerStart(xTimer1, 0);
74     xTimerStart(xTimer2, 0);
75 }
76
```

PROBLEMS	OUTPUT	DEBUG CONSOLE	TERMINAL
Total sizes:			
Used static DRAM:		11568 bytes (169168 remain, 6.4% used)	
.data size:		8976 bytes	
.bss size:		2592 bytes	
Used static IRAM:		48594 bytes (82478 remain, 37.1% used)	
.text size:		47567 bytes	
.vectors size:		1027 bytes	
Used Flash size :		115999 bytes	
.text :		84547 bytes	
.rodata :		31196 bytes	
Total image size:		173569 bytes (.bin may be padded larger)	

Giải thích:

- Hàm callback được sử dụng chung của 2 timer hàm void **periodic_timer_callback**
- Hàm sẽ xét tên của timer gọi nó và thực hiện tác vụ tương ứng
- 2 timer ở đây là **Timer 1** và **Timer 2**

2. Nạp và kết quả

- Nạp vào esp, ta thấy kết quả hiện lên màn hình serial

```
31         printf("STOP Timer 1\n");
32         xTimerStop(arg,0);
33     }
34     else {
35         printf("ahihi - Nhóm 3 L01 - HCMUT K19\n");
36         vTimerSetTimerID(arg,(void*)count);
37     }
38 }
39 else if (strcmp(name,"Timer 2")) {
40     if(count > TIMER_2_MAX) {
41         printf("STOP Timer 2\n");
42         xTimerStop(arg,0);
43     }
44 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
I (61) boot: ## Label          Usage          Type ST Offset   Length
I (68) boot: 0 nvs             WiFi data      01 02 00009000 00006000
I (76) boot: 1 phy_init        RF data        01 01 0000f000 00001000
I (83) boot: 2 factory         factory app     00 00 00010000 00100000
I (91) boot: End of partition table
I (95) boot_comm: chip revision: 1, min. application chip revision: 0
I (102) esp_image: segment 0: paddr=00010020 vaddr=3f400020 size=07ad4h ( 31444) map
I (122) esp_image: segment 1: paddr=00017afc vaddr=3ffb0000 size=02310h ( 8976) load
I (126) esp_image: segment 2: paddr=00019e14 vaddr=40080000 size=06204h ( 25092) load
I (140) esp_image: segment 3: paddr=00020020 vaddr=400d0020 size=14a44h ( 84548) map
I (171) esp_image: segment 4: paddr=00034a6c vaddr=40086204 size=05bd0h ( 23504) load
I (181) esp_image: segment 5: paddr=0003a644 vaddr=50000000 size=00010h (    16) load
I (187) boot: Loaded app from partition at offset 0x10000
I (187) boot: Disabling RNG early entropy source...
I (202) cpu_start: Pro cpu up.
I (202) cpu_start: Starting app cpu, entry point is 0x40081090
0x40081090: call_start_cpu1 at C:/Users/luong/esp/esp-idf/components/esp_system/port/cpu_start.c:148

I (0) cpu_start: App cpu up.
I (216) cpu_start: Pro cpu start user code
I (216) cpu_start: cpu freq: 160000000
I (216) cpu_start: Application information:
I (221) cpu_start: Project name:   template-app
I (226) cpu_start: App version:    1
I (230) cpu_start: Compile time:   Dec 12 2022 21:53:29
I (236) cpu_start: ELF file SHA256: 7da3ef58e92f3916...
I (242) cpu_start: ESP-IDF:        v4.4.3
I (248) heap_init: Initializing. RAM available for dynamic allocation:
I (255) heap_init: At 3FFAE6E0 len 00001920 (6 KiB): DRAM
I (261) heap_init: At 3FFB2D30 len 0002D2D0 (180 KiB): DRAM
I (267) heap_init: At 3FFE0440 len 00003AE0 (14 KiB): D/IRAM
I (273) heap_init: At 3FFE4350 len 00018CB0 (111 KiB): D/IRAM
I (280) heap_init: At 4008BDD4 len 0001422C (80 KiB): IRAM
I (287) spi_flash: detected chip: generic
I (291) spi_flash: flash io: dio
I (296) cpu_start: Starting scheduler on PRO CPU.
I (0) cpu_start: Starting scheduler on APP CPU.
ihaha - Nhóm 3 L01 - HCMUT K19
ahihi - Nhóm 3 L01 - HCMUT K19
ihaha - Nhóm 3 L01 - HCMUT K19
ahihi - Nhóm 3 L01 - HCMUT K19
ihaha - Nhóm 3 L01 - HCMUT K19
```



```

I (0) cpu_start: App cpu up.
I (216) cpu_start: Pro cpu start user code
I (216) cpu_start: cpu freq: 160000000
I (216) cpu_start: Application information:
I (221) cpu_start: Project name:      template-app
I (226) cpu_start: App version:      1
I (230) cpu_start: Compile time:      Dec 12 2022 21:53:29
I (236) cpu_start: ELF file SHA256:  7da3ef58e92f3916...
I (242) cpu_start: ESP-IDF:          v4.4.3
I (248) heap_init: Initializing. RAM available for dynamic allocation:
I (255) heap_init: At 3FFAE6E0 len 00001920 (6 KiB): DRAM
I (261) heap_init: At 3FFB2D30 len 0002D2D0 (180 KiB): DRAM
I (267) heap_init: At 3FFE0440 len 00003AE0 (14 KiB): D/IRAM
I (273) heap_init: At 3FFE4350 len 0001BCB0 (111 KiB): D/IRAM
I (280) heap_init: At 4008BDD4 len 0001422C (80 KiB): IRAM
I (287) spi_flash: detected chip: generic
I (291) spi_flash: flash io: dio
I (296) cpu_start: Starting scheduler on PRO CPU.
I (0) cpu_start: Starting scheduler on APP CPU.
ihaha - Nhom 3 L01 - HCMUT K19
ahihi - Nhom 3 L01 - HCMUT K19
ihaha - Nhom 3 L01 - HCMUT K19
ahihi - Nhom 3 L01 - HCMUT K19
ihaha - Nhom 3 L01 - HCMUT K19
ihaha - Nhom 3 L01 - HCMUT K19
ahihi - Nhom 3 L01 - HCMUT K19
ihaha - Nhom 3 L01 - HCMUT K19
ahihi - Nhom 3 L01 - HCMUT K19
STOP Timer 2
ahihi - Nhom 3 L01 - HCMUT K19
ahihi - Nhom 3 L01 - HCMUT K19
ahihi - Nhom 3 L01 - HCMUT K19
ahihi - Nhom 3 L01 - HCMUT K19
ahihi - Nhom 3 L01 - HCMUT K19
ahihi - Nhom 3 L01 - HCMUT K19
STOP Timer 1

```

- Chuỗi “Timer 2 - Nhom 3 L01 - HCMUT K19 “ được in ra 3 giây sau mỗi lần và in ra 5 lần
- Chuỗi “Timer 1 - Nhom 3 L01 - HCMUT K19 “được in ra 2 giây sau mỗi lần và in ra 10 lần