ĐẠI HỌC QUỐC GIA TP.HCM
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH

THỰC HÀNH HỆ THỐNG NHÚNG (CO3054)
Lab 3: FreeRTOS Tasks Scheduling
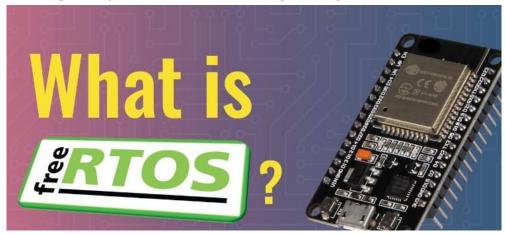
Giáo viên hướng dẫn: Vũ Trọng Thiên

Sinh viên thực hiện: Lê Thị Hồng Thắm - 2012069

Nghiêm Lương Sơn - 2014373

Tháng 11/2023

1. FreeRTOS task

- FreeRTOS cho phép chạy số lượng tác vụ không giới hạn miễn là phần cứng và bộ nhớ có thể xử lý được.
- Là một hệ điều hành thời gian thực, FreeRTOS có thể xử lý cả các tác vụ theo chu kỳ và không theo chu kỳ. Trong RTOS, một tác vụ được xác định bằng hàm C đơn giản, lấy tham số void* và không trả về gì (void).

2. Hiện thực

2.1. Prioritized Pre-emptive Scheduling with Time Slicing

```
------------------------------------
Task1 is Counting: 27368/9622100
Task2 is Counting: 0/7433210
Idle count: 10
------------------------------------
====================================
I (443) main_task: Returned from app_main()
====================================
Task 2 has terminated during: 2139 ms
====================================
Task 2 has ended!
Result of 2 tasks with time-slicing:
------------------------------------
Task1 is Counting: 7599030/9622100
Task2 is Counting: 7433210/7433210
Idle count: 10
------------------------------------
====================================
====================================
Task 1 has terminated during: 2480 ms
====================================
Task 1 has ended!
Result of 2 tasks with time-slicing:
------------------------------------
Task1 is Counting: 9622100/9622100
Task2 is Counting: 7433210/7433210
Idle count: 10
------------------------------------
====================================
```

2.2. Prioritized Pre-emptive Scheduling (without Time Slicing)

```
I (333) main_task: Calling app_main()
====================================
Task 1 will be started!

The state of Task1's counter before it started
------------------------------------
Task1 is Counting: 0/9622100
Task2 is Counting: 0/7433210
Idle count: 10
------------------------------------
====================================
====================================
Task 2 will be started!

The state of Task2's counter before it started
------------------------------------
Task1 is Counting: 27369/9622100
Task2 is Counting: 0/7433210
Idle count: 10
------------------------------------
====================================
I (443) main_task: Returned from app_main()
====================================
Task 2 has terminated during: 2139 ms
====================================
Task 2 has ended!
Result of 2 tasks without time-slicing:
------------------------------------
Task1 is Counting: 7599031/9622100
Task2 is Counting: 7433210/7433210
```

```
----------------------------------
Task1 is Counting: 27369/9622100
Task2 is Counting: 0/7433210
Idle count: 10
----------------------------------
==================================
I (443) main_task: Returned from app_main()
==================================
Task 2 has terminated during: 2139 ms
==================================
Task 2 has ended!
Result of 2 tasks without time-slicing:
----------------------------------
Task1 is Counting: 7599031/9622100
Task2 is Counting: 7433210/7433210
Idle count: 10
----------------------------------
==================================
==================================
Task 1 has terminated during: 2480 ms
==================================
Task 1 has ended!
Result of 2 tasks without time-slicing:
----------------------------------
Task1 is Counting: 9622100/9622100
Task2 is Counting: 7433210/7433210
Idle count: 10
----------------------------------
==================================
```

2.3. Co-operative Scheduling

```
I (323) app_start: Starting scheduler on CPU1
I (323) main_task: Started on CPU0
I (333) main_task: Calling app_main()
==================================
Task 1 will be start!
Task's counter before start
----------------------------------
Task1 is Counting: 0/1000000
Task2 is Counting: 0/2000000
Task3 is Counting: 0/3000000
Idle count: 10
----------------------------------
==================================
==================================
Task 2 will be start!
Task's counter before start
----------------------------------
Task1 is Counting: 92692/1000000
Task2 is Counting: 0/2000000
Task3 is Counting: 0/3000000
Idle count: 10
----------------------------------
==================================
==================================
Task 3 will be start!
Task's counter before start
----------------------------------
Task1 is Counting: 370579/1000000
Task2 is Counting: 207594/2000000
Task3 is Counting: 0/3000000
```

```
Task2 is Counting: 479680/2000000
Task3 is Counting: 3000001/3000000
Idle count: 10
------------------------------------
====================================
====================================
Task 1 terminate during: 783 ms
====================================
Task 1 ended!
Result:
------------------------------------
Task1 is Counting: 1000000/1000000
Task2 is Counting: 896091/2000000
Task3 is Counting: 3000001/3000000
Idle count: 10
------------------------------------
====================================
====================================
Task 2 terminate during: 921 ms
====================================
Task 2 ended!
Result:
------------------------------------
Task1 is Counting: 1000000/1000000
Task2 is Counting: 2000000/2000000
Task3 is Counting: 3000001/3000000
Idle count: 10
------------------------------------
====================================
None
```

3. Tư liệu

+ **Link github:** nghiemluongson/embedded_System_All_Labs at Lab3 (github.com)

+**Demo ex1:**
https://drive.google.com/file/d/12Q1jkS45ZyhwuqWz7Imy1VO4GI3Zguq9/view?usp=sharing

+ **Demo ex2:**
https://drive.google.com/file/d/14GV5OpwWpEFg_r86MGpW8mdDDk1t789J/view?usp=drive_link

+ **Demo ex3:** https://drive.google.com/file/d/1chVRvqhgxvu2PHZVp45dnc6M41Vo0hT-/view?usp=sharing