

Hardware, Microcontroladores y Sistemas Embebidos

Práctica de Laboratorio 4

Contenido involucrado con Sistemas Operativos de Tiempo Real y Threads.

Ejercicio 1 - Threading Raspberry Pi (Linux)

- Correr dos programas de prueba, POSIX (C, C++) threads y Python threads

[Linux Tutorial: POSIX Threads](#)

[Multi-Threaded Programming With POSIX Threads](#)

[An Intro to Threading in Python – Real Python](#)

Ejercicio 2 - FreeRTOS Arduino

- Correr dos programas de prueba
- Semáforos
 - [Using FreeRTOS Semaphores in Arduino IDE](#)
- Multitask
 - [Using FreeRTOS multi-tasking in Arduino](#)

Ejercicio 3 - Interrupciones y Timers Arduino

- Correr con ejemplos de Adafruit

[Overview | Multi-tasking the Arduino - Part 2](#)

- TMR1
- TMR2

- **Nota:** este ejercicio no vale puntos para el laboratorio, puede servir para repasar todo el concepto de máquinas de estado y para un mejor diseño de sus algoritmos y programas.

Ejercicio 4 - Máquinas de Estados Arduino/Raspberry Pi

- Correr 1 programa de prueba
 - Candado con botones
 - Semáforo

[Overview | Multi-tasking the Arduino - Part 1](#)
[Building a simple State Machine in Python.](#)

Links Utilidad

1. [FreeRTOS](#)
2. [Phillip Stevens - Hackster Profile](#)
3. [Arduino FreeRTOS](#)
4. [Introduction to RTOS](#)
5. [Quick start guide](#)
6. [FreeRTOS API categories](#)
7. [StateMachine](#)
8. [jrullan/StateMachine: State machine library for Arduino](#)
9. [Finite State Machines in Embedded Programming](#)
10. [Using Finite State Machines](#)
11. [Raspberry Pico: Programming with PIO State Machines | by Sebastian | Geek Culture](#)
12. [Implementing Finite State Machines](#)
13. [Digital Watch on Arduino using Finite State Machine](#)
14. [Finite State Machine \(Finite Automata\)](#)
15. [Async IO in Python: A Complete Walkthrough – Real Python](#)
16. [C++ Multithreading](#)
17. [Interlude: Thread API](#)