

Course Title:**Real Time Operating System (RTOS) Fundamentals and application****Introduction:**

This course covers fundamental of RTOS and use FreeRTOS as example.

FreeRTOS is well-known open source real-time kernel. It is specifically designed for embedded applications and run on variety of micro-controllers.

This course combined ARM and FreeRTOS to enable participants benefit from the strength of their respective characteristics.

Objective:

At the end of the program, participants will be able to

- Develop RTOS application easily.
- Debug multi tasking application
- Porting to ARM Cortex based SoC
- Developing low level drivers and integrating with FreeRTOS

Suitable for:

Those who are involved in works related to embedded system

Pre Requisites:

Adequate C programming skills, knowledge of embedded microprocessor

Duration:

3 Days

Outline:**Day 1**

Introduction to OS

Scheduling algorithm

Task, process, thread and differences among them

Preemptive kernels

Synchronization mechanism semaphore, binary, counting and mutex

Signals and interrupt handlers in operating system

Inter task communication mechanism Message Queue, Pipes, mailbox, event registers

API interfaces
Protocol stacks, Socket interfaces
I/O management, device driver

Day 2

Introduction to FreeRTOS
Scheduler in FreeRTOS
task management in FreeRTOS
semaphores
inter task Communication in FreeRTOS
Case Study: Thermostat – interfacing sensors, implementing low-level drivers, partitioning to tasks, Managing tasks, synchronization.
Critical section FreeRTOS
Interrupt handling
Timers and watch dog timers
I/O and file systems
Porting issues

Day 3

Case Study: Thermostat (continues) – enhancing with further code hardening
Installing GNU/Real View tool chain,
Assembler, compilers
Implementing Start up code
Using gnu size, elf reader
Understanding SoC
Configuring Tool chain for FreeRTOS porting.
Understanding BSP for ARM Architecture.
Understanding context switching in FreeRTOS
Case Study: Thermostat (continues)
Porting FreeRTOS to Cortex hardware
Q&A sessions