

Basic RTOS concepts:

1.What is an RTOS, what does Real Time means?

2.Legacy systems and how the requirement for RTOS came?

3.Embedded and RTOS relation

4.Scheduling algorithms:

- 4.1 Round-robin
- 4.2 Rate Monotonic (RMA)
- 4.3 Least Slack Slack-Time
- 4.4 Priority scheduling
- 4.5 Pre-emptive scheduling
- 4.6 Earliest Deadline First (EDF)

5.Thread:

- 5.1 Create
- 5.2 Delete
- 5.3 Suspend
- 5.4 Resume
- 5.5 Task statistics or Idle thread.
- 5.6 Task control block(TCB)
- 5.7 Relinquish
- 5.8 Thread stack

6.Inter task communication and synchronization:

- 6.1 Mutex
- 6.2 Critical section
- 6.3 Semaphore
- 6.4 Message queue
- 6.5 Events
- 6.7 Pipes
- 6.8 Signals

7.Memory management

- 7.1 Virtual memory
- 7.2 Dynamic memory
- 7.3 Stack
- 7.4 Shared memory

8.RTOS terminology:

- 8.1 Priority
- 8.2 Preemption
- 8.3 Priority inheritance
- 8.4 Priority inversion
- 8.5 Reentrancy
- 8.6 Context switching(save and restore).
- 8.7 Dead lock(Details, startagies to avoid and come out)
- 8.8 Live lock
- 8.9 Latency
- 8.10 Binary semaphore
- 8.11 OS Timers
- 8.12 Starvation
- 8.13 spin lock (busy waiting).
- 8.14 Atomic section

9. How to select an RTOS for the system?

10. Applications of RTOS