



SNS COLLEGE OF TECHNOLOGY

(An Autonomous Institution)

COIMBATORE-35

Accredited by NBA-AICTE and Accredited by NAAC – UGC with A++ Grade

Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai



19EEE305 / EMBEDDED SYSTEMS III YEAR / VI SEMESTER


UNIT-IV: RTOS BASED EMBEDDED SYSTEM DESIGN

RTS, RTOS, GPOS

RTS

- **REAL TIME SYSTEMS** : Real-time systems are those systems in which the correctness of the system depends not only on the Output , but also on the time at which the results are produced(Time constraints must be strictly followed).
- Real time systems are two types
 - (i) Soft real time systems
 - (ii) Hard real time systems

RTS types

- 
- A **Soft Real time system** is one in which the performance of the system is only degraded but, not destroyed if the timing deadlines are not met .

For Ex: Air conditioner, TV remote or music player, Bus reservation ,automated teller machine in a bank , A Lift etc.

- A **Hard Real time system** is one in which the failure to meet the time dead lines may lead to a complete catastrophe or damage to the system

For Ex: Air navigation system, Nuclear power plant , Failure of car brakes , Gas leakage system ,RADAR operation ,Air traffic control system

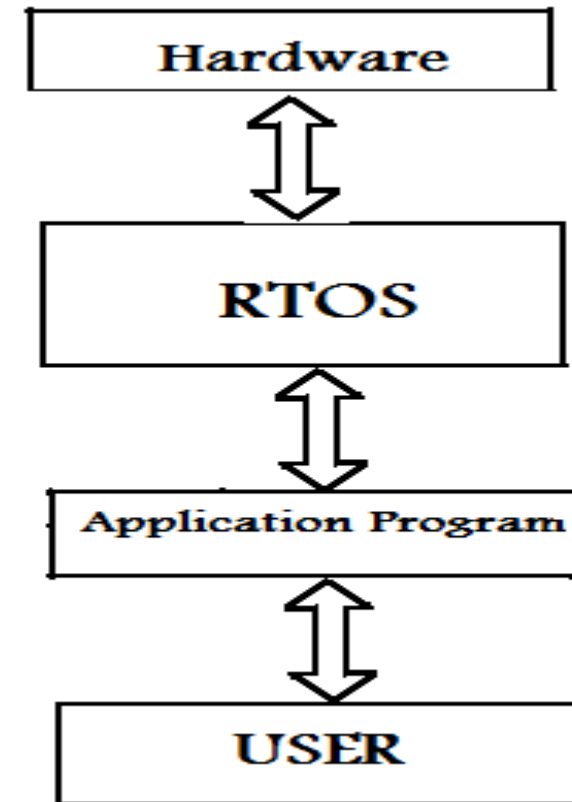
RTOS

- It is an operating system that supports real-time applications by providing logically correct result within the deadline set by the user.
- A real time operating system makes the embedded system into a real time embedded system.
- The basic structure of **RTOS is similar to regular OS** but, in addition, it provides mechanisms to allow real time scheduling of tasks.



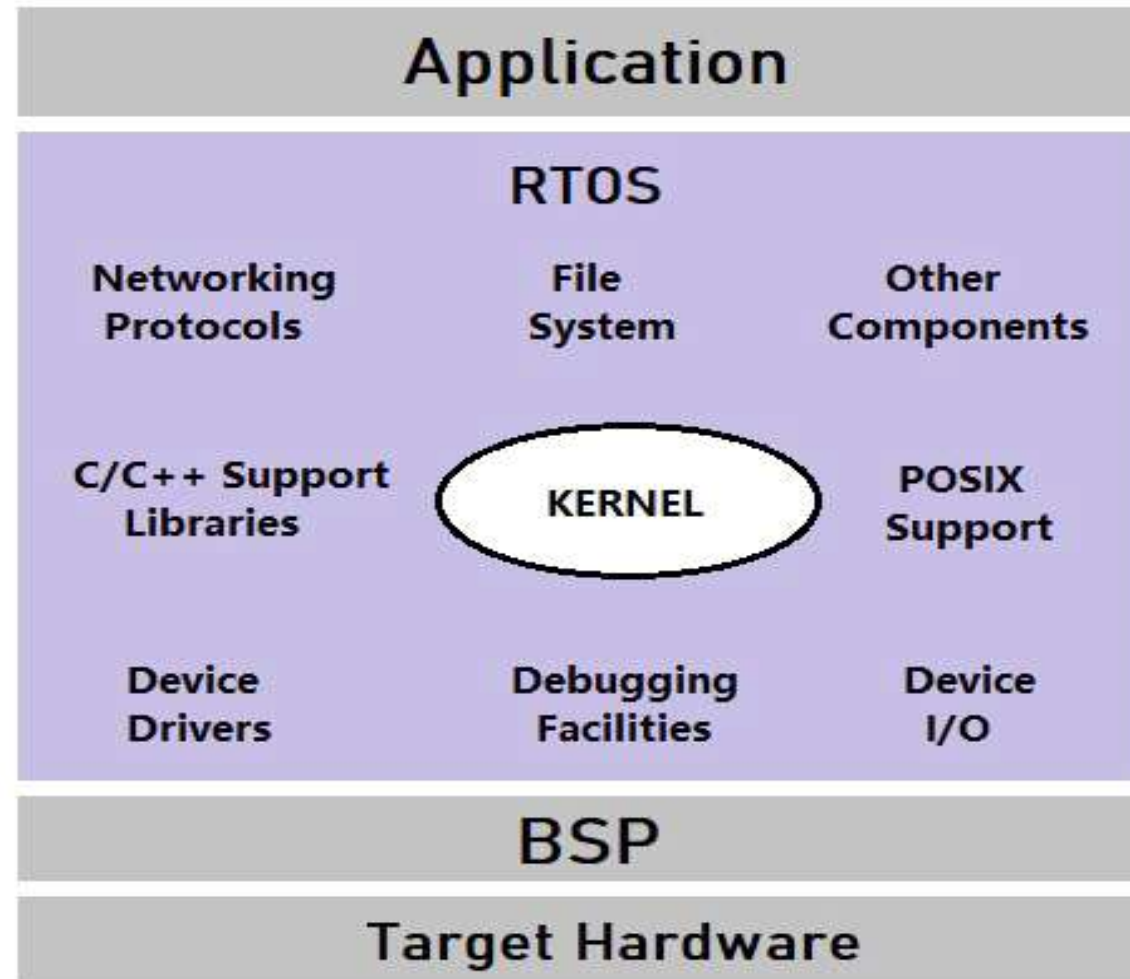
RTOS

- All the embedded systems are not designed with RTOS. **Low end application systems** do not require the RTOS but only **High end application oriented embedded systems** which require scheduling alone need the RTOS.
- An embedded system which measures Temperature or Humidity etc. do not require any operating system.
- Where as a Mobile phone , RADAR or Satellite system used for high end applications require an operating system.





RTOS - Architecture





RTOS - Examples



Best RTOS for Embedded System





RTOS - Examples

RTOS	Applications/Features
Windows CE (Microsoft Widows)	Used small foot print mobile and connected devices Supported by ARM, MIPS, SH4 & x86 architectures
LynxOS	Complex, hard real-time applications - POSIX- compatible, multiprocess, multithreaded OS. - Supported by x86, ARM, PowerPC architectures
VxWorks (Wind river)	- Most widely adopted RTOS in the embedded industry. - Used in famous NASA rover robots Spirit and Opportunity - Certified by several agencies and international standards for real time systems, reliability and security-critical applications.
Micrium μ C/OS-II	- Ported to more than a hundred architectures including x86, Mainly used in microcontrollers with low resources. - Certified by rigorous standards, such as RTCADO-178B
QNX	- Most traditional RTOS in the market. - Microkernel architecture; completely compatible with the POSIX - Certified by FAADO-278 and MIL-STD-1553 standards.
Symbian	Designed for Smartphones Supported by ARM, x86 architecture
VRTX	- Suitable for traditional board based embedded systems and SoC architectures - Supported by ARM, MIPS, PowerPC & other RISC architectures
RTLINUX	Open source
Neutrino	



RTOS VS GPOS



RTOS

Real-Time Operating System

- Deterministic: no random execution pattern
- Predictable Response Times
- Time Bound
- Preemptive Kernel

Examples:

Contiki source code, FreeRTOS™, Zephyr™ Project

Use Case:

Embedded Computing

GPOS

General-Purpose Operating System

- Dynamic memory mapping
- Random Execution Pattern
- Response Times not Guaranteed

Examples:

Microsoft® Windows® operating system, Apple® macOS® operating system, Red Hat® Enterprise Linux® operating system

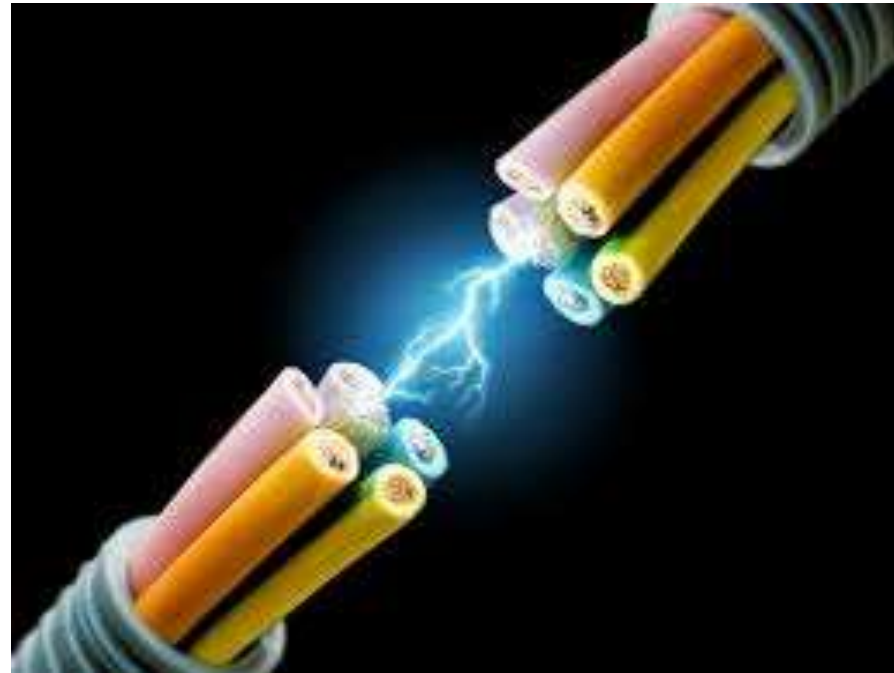
Use Case:

Desktop, Laptop, Tablet computers





RECAP....



...THANK YOU

