Impact Factor (SJIF): 3.632

International Journal of Advance Research in Engineering, Science & Technology

e-ISSN: 2393-9877, p-ISSN: 2394-2444

Volume 3, Issue 6, June-2016

A Critical Analysis on High Performance Real Time Operating System

Jayna Donga[1], Vatsal Shah[2], Kanu Patel[2]

[1] Assistant Professor, MBICT College, New Vidhyanagar
[2]Assistant Professor, Birla VishvakarmaMahavidhyalaya College, VallabhVidhyanagar

Abstract – A real time operating system means a type of operating system which complete the task by providing correct result within given dead line. Real time operating system (RTOS) is specially used to meet the real time constraint and to support the sophisticated facilities provided by embedded system. This paper describe the importance of Real Time operating system in real life, also its working. This paper also give an overview of one of the best high performance real time operating system SCIOPTA.

Keywords – Real Time Operating System (RTOS), SCIOPTA.

I. INTRODUCTION

The Real Time Operating System (RTOS) is an operating system which support the real time embedded system like cell phone, a smartcard, a music player, a router, or the electronics in an automobile. The goal of Real Time Operating System (RTOS) is to provide the correct result within given dead line [1] [2] [4].

Real Time Operating System (RTOS) differs from the general purpose OS, RTOS is a real time OS working with real time constraints as power, time and efficient usage of memory. Most of the embedded systems are bound to real time constraints and it is achieved using real time system. General purpose operating systems are suitable to do multiple tasks at the same time, but synchronization is a problem with GPOS. To do multiple tasks with worst case execution time on a particular architecture real time OS is used in an embedded computing system. The main concern of RTOS is it produces an accurate output within the deadline or time [3].

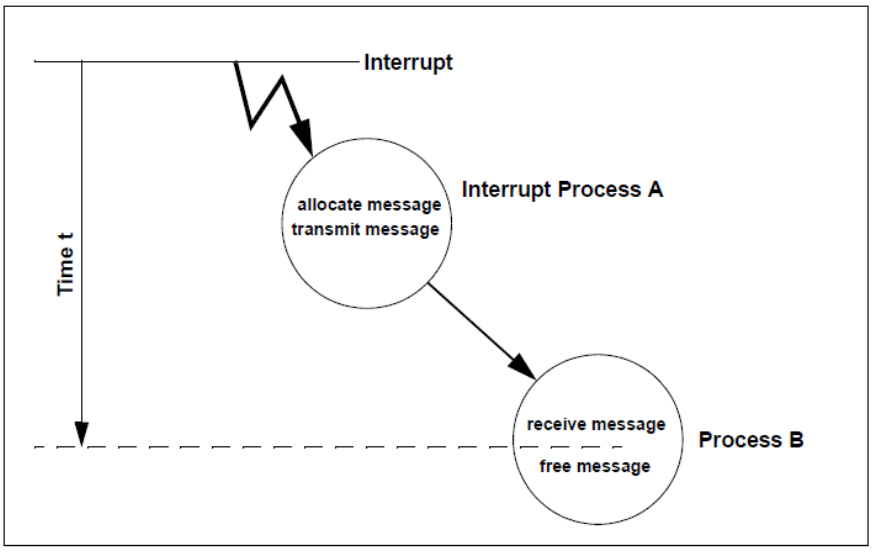

Figure 1. Real-Time System Definition [5]

A real-time operating system (RTOS) is the core control software in a real-time system. In a real-time system itmust be guaranteed that specific tasks respond to external events within a limited and specified time.[4] [5]

Table I: Difference between General Purpose OS and RTOS [2]

	RTOS	General Purpose OS
Determinism	Deterministic	Non-deterministic
Preemptive kernel	All kernel operations are preemptable.	Not Necessary
Priority Inversion	Have mechanisms to prevent priority inversion	No such mechanism is present
Task Scheduling	Scheduling is time based	Scheduling is process based
Latency	Have their worst case latency defined	Latency is not of a concern Purpose OS
Application	Typically used for embedded applications	General purpose OS is used for desktop PCs or other generally purpose PCs

There are three types of real time operating system depending upon the nature of application. [3]

A. Soft Real Time Operating System
B. Hard Real Time Operating System
C. Firm Real Time Operating System

A. Soft Real Time Operating System

Soft real time OS is a type of OS where certain deadlines may be missed, they will respond at a time t=0+. Soft real time systems are not constrained to extreme rules. The critical time of the soft real time may be delayed to some extent. The expected latency between the tasks and time constraints may be deviated. The preemption period for a soft real time task is about few milliseconds [3].

Example:
Digital camera, mobile phones, online data base etc.

B. Hard Real Time Operating System

Hard real time OS is a type of OS we can predict the deadline, they will respond at a time t=0. Hard real time systems are constrained to predicted time constraints, deadlines and latency.The preemption period for hard real time system is almost less than few microseconds [3].

Examples:
Air bag control in cars, anti-lock brake, engine control system etc.

C. Firm Real Time Operating System

This type of RTOS has certain time constraints which are not strict and it may cause undesired effect [3].

Example for firm RTOS is automated visual inspection in industrial automation. This system examines and detects the defected parts of assembly line. This type is also called event response system.

II. FEATURES OF REAL TIME OPERATING SYSTEM

Real Time operating system provides following features. [1] [2]

- Synchronization:
  Synchronization is necessary for real time tasks to share mutually exclusive resources. For multiple threads to communicate among themselves in a timely fashion, predictable inter-task communication and synchronization mechanisms are required.

- Interrupt Handling:
  Interrupt Service Routine (ISR) is used for interrupt handling. Interrupt latency is defined as the time delay between the occurrence of an interrupt and the running of the corresponding Interrupt Service Routine (ISR).

- Timer and clock:
  Clock and timer services with adequate resolution are vital part of every real-time operating system.

- Real-Time Priority Levels:
  A real-time operating system must support real-time priority levels so that when once the programmer assigns a priority value to a task, the operating system does not change it by itself.

- Fast Task Preemption:
  For successful operation of a real-time application, whenever a high priority critical task arrives, an executing low priority task should be made to instantly yield the CPU to it.

- Memory Management:
  Real-time operating system for large and medium sized application are expected to provide virtual memory support, not only to meet the memory demands of the heavyweight real-time tasks of an application, but to let the memory demanding non-real-time applications such as text editors, e-mail etc. An RTOS uses small memory size by including only the necessary functionality for an application while discarding the rest.

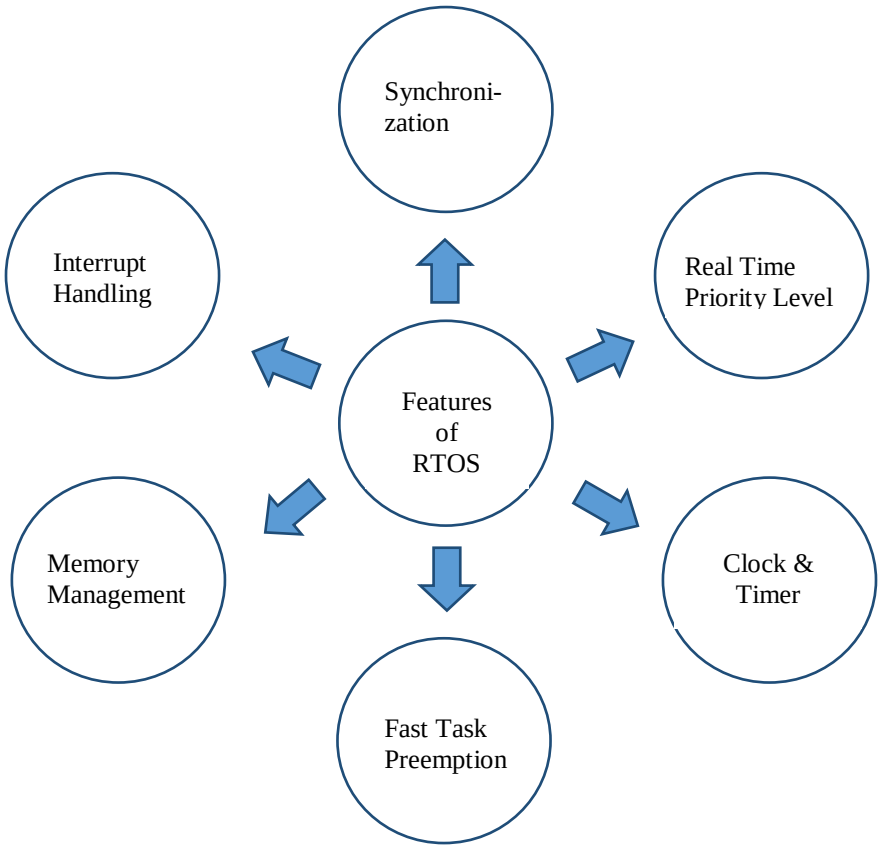

Figure 2: Features Provided By RTOS [2]

III. SCIOPTA

The SCIOPTA is one of the best high performance real time operating system. SCIOPTA is registered trademark of Litronic AG.Founded as Litronic AG in 1988, SCIOPTA Systems AG is based in Bottmingen/Basel (Switzerland). Since the inception SCIOPTA Systems AG is specialized in real-time systems [5].

Futures of SCIOPTA [5].

A. Message-Based Architecture

SCIOPTA is designed on a message-based architecture allowing direct message passing between processes. Messages are mainly used for interprocess communication and synchronization. SCIOPTA messages are stored and maintained in memory pools. The kernel memory pool manager is designed for high performance and memory fragmentation is avoided.

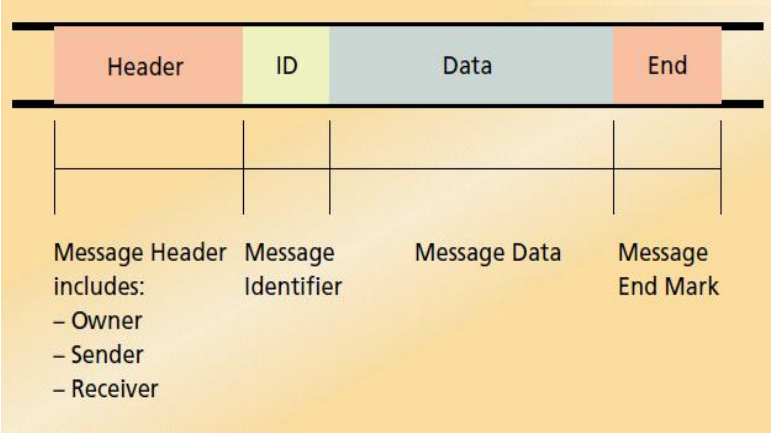

Figure 3. SCIOPTA message architecture [5]

B. Reduced Time-to-Market

A powerful set of system calls managing the message passing and the resources of SCIOPTA allows you to shorten the development time and thus to reduce the time-to-market for your products. SCIOPTA is a message-based real-time operating system. Standardized processes and interprocess communication result in clear system designs and are easy to write, to read and to maintain. As processes are communicating with well defined messages and processes can be grouped into modules, SCIOPTA systems are very well suited for teamwork in big projects.

C. Easy to Debug

A SCIOPTA system can be easily debugged by tracing messages until a breaking or blocking situation is reached. The message trace allows analysis the message sequence preceding a possible faulty system state. A SCIOPTA message contains not only the message data but also administrative information such as the owner, sender and addressee process. The debugger can, therefore, quickly find lost messages or messages which have been sent to wrong processes, by analysing the message pools.

D. Error Handling

SCIOPTA features centralized error handling by using SCIOPTA error hooks. Each time SCIOPTA detects an error the error hook will be called. This guarantees consistent error handling covering the whole system. Problems common in traditional operating systems when using individual error handling by different team members spread over the whole application code can be avoided in SCIOPTA.

E. SCIOPTA Modules

Processes can be grouped in SCIOPTA modules, which allows you to design a very modular system. Modules can be static or created and killed during run time. SCIOPTA also supports a friend concept. Modules can be friend to other modules. Friend modules have privileged access to either modules system resources. SCIOPTA modules can be used to encapsulate whole system blocks (such as a communication stack) and protect them from other modules in the system.

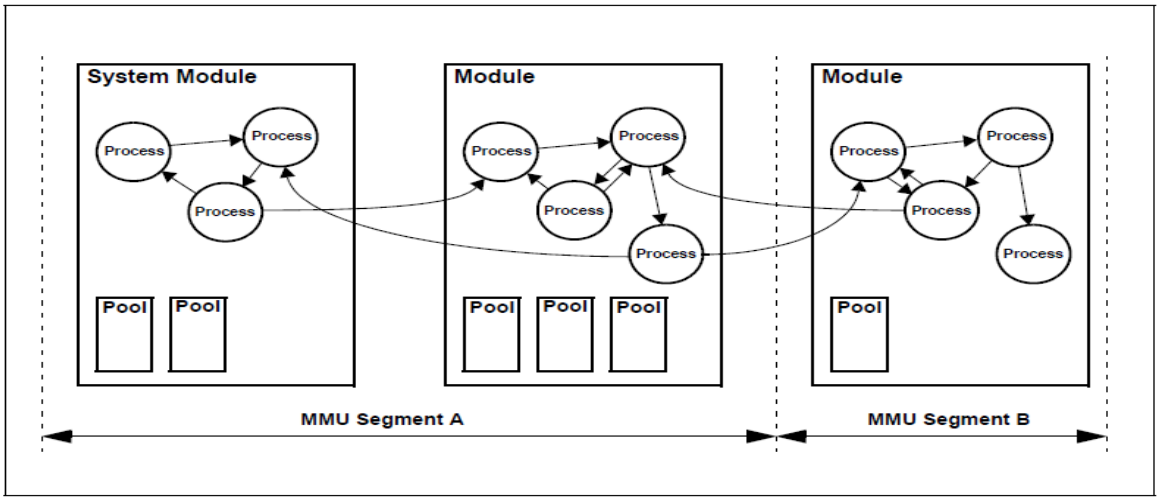

Figure 4. SCIOPTA Module Architecture [5]

IV. CONCLUSION

Real time system is typically used to achieve the correct result in deadline. Its feature like memory management and process scheduling are very important aspect. Currently So much research is going on this topics. And SCIOPTA is one of the best real time operating system which can also be used in distributed manner.

V. ACKNOWLEDGEMENT

We are very thankful to Litronic AG for finding such an amazing product SCIOPTA (A High Performance Real Time Operating System) and providing its brochure online.

References

[1] Jane W. S. Liu, “Real-time System”, (published by Person Education, Page no. 20-40).
[2] DiptiDiwase, Shraddha Shah, TusharDiwase and PriyaRathod, "Survey Report on Memory Allocation Strategies for Real Time Operating System in Context with Embedded Devices", published in International Journal of Engineering Research and Applications, Vol. 2, Issue 3, May-Jun 2012, pp.1151-1156.
[3] "REAL TIME OPERATING SYSTEM (RTOS)" published by robomart.com blog on September 7, 2015
[4] Vatsal Shah, Dr. Apurva Shah, “An Analysis and Review on Memory Management Algorithms for Real Time Operating System”, published in International Journal of Computer Science and Information Security, Volume 14 Issue 5, June 2016
[5] Brochure of "SCIOPTA High Performance Real-Time Operating Systems" published by byLitronic AG.