



(Print)

JCIT Vol. 9(1), 14-16 (2018). Periodicity-2-Monthly

(Online)



JOURNAL OF COMPUTER & INFORMATION TECHNOLOGY
An International Open Free Access Peer Reviewed Research Journal of Computer
Science Engineering & Information Technology
website:- www.compitjournal.org

Modification of Linux based Real Time Operating System

ARSHI KHAN¹ and AKHILESH JAIN²

¹M.tech scholar NRI Institute of Information Science & Technology, Bhopal (India)

²Assistant Professor NRI Institute of Information Science & Technology, Bhopal (India)

Corresponding Author Email: Arshikhan800@gmail.com

<http://dx.doi.org/10.22147/jucit/090103>

Acceptance Date 01st February 2018,

Online Publication Date 2nd February, 2018

Abstract

With the widely use of Linux systems in the simulation field, improving the performance of Linux to make it compatible with the real-time simulation operating system is becoming an attractive direction of research. Linux-based simulation platform as Linux operating system is free and open source, and it can be tailored and customized, also, it has the characters of comprehensive network function, stability and safety. Although Linux has many incomparable advantages over windows, the fact that the installation of Linux is less convenient than windows, interface is less friendly, configuration is easier, software support is less plentiful, and so on. These characters Increase the difficulty of the simulation based on the Linux platform. Therefore, It's meaningful to build a semi-structurally simulated platform that use windows as the host machine to develop, compile, debug and monitor, Linux as target machine to implement simulation program.

Introduction to operating systems

An operating system is system software that controls the internal activities of computer hardware and provides user interface. It is the first programmer loaded into the computer memory after the computer is switched on. Example - MS DOS, UNIX, LINUX, WINDOWS 98, WINDOWS XP, WINDOWS VISTA, WINDOWS 7, WINDOWS 8 ETC.

- What's responsible for managing hardware resources and hosting applications that run on the computer?
"Operating systems".
- What is Operating Systems?"
It is just a collection of function or system calls that provides interface between hardware and application programs."

- What does and Operating System provide?

"Input, output, multitasking, interrupt and event handling, Inter task communication, memory management etc."

RTOS (Real Time Operating Systems) :

But the core of Operating Systems is kernel which is typically a small, highly optimized set of libraries. Thus we can conclude that (RTOS) is an operating system that basically supports real time applications by providing logically correct result within the dead line required. Basic structure is similar to operating system in addition it gives mechanism to allow real time scheduling of tasks. It may not increase the execution, but provide much more precise and predictable timing characteristics than general purpose operating system based upon degree of tolerance in meeting deadlines, (RTOS) are classified into following categories:

1. Hard real time: where degree of tolerance for missed deadline is negligible, A missed deadline can result in catastrophic failure of the system.

2. Soft real time: where deadline may be missed occasionally, but system doesn't lead to fail.

For example, imagine that some engineers developed an automatic sensor based system for landing gears of aero planes (*i.e.* the plane automatically detects the runway and the wheels come out before landing without any instruction given by the pilot). Imagine how catastrophic it could be if the landing gears don't come out in real time causing death or injury to so many passengers on board. What if the landing gears open after the plane hits the ground. To stop this hard real-time system is needed. We need assurance as the system designer that no single operation will exceed little timing constraints. On the other hand, a soft real-time operating system may suffice for lesser important things like video calling software. Even if the video doesn't stream in real time it won't do any harm. Well what's important is that if programmed correctly, a RTOS can guarantee that a program will run with very consistently and efficiently. Real-time operating systems do this by giving programmers with a high degree of command over how tasks are prioritized, and typically also allow checking to make sure that important deadlines are met first. RTOS are designed to run critical applications reliably and with precise timing whereas Operating systems like Windows are designed to maintain user responsiveness with many programs and services running, while it is possible (and even sometimes preferable) to implement real-time systems without any kind of operating system, doing so is usually not very convenient, nor is it easy. In this work we will only discuss real-time with respect to operating systems, focusing on Linux like OS.

Linux based OS have shown low predictability when it comes to real-time applications. They have also proven to have the advantages such as stable performance speed and flexibility towards different systems. This thesis aims is to customize and modify bare Linux kernel to be acceptable as a real time operating system and gives deterministic timing behaviour in real-time environments.

Real time kernel : The heart of a real-time OS (and the heart of every OS, for that matter) is the kernel. A kernel is the central core of an operating system, and it takes care of all the OS jobs:

1. Booting
2. Task Scheduling

1.1 Problem Statement and Motivation :

Linux can run on the most widely available computers:-IBM PCs and compatible machines.

There are various advantages and disadvantages of LINUX being used as a hard real-time OS.

Advantages:

1. Excellent stability.
2. Source code availability.
3. Non restricted License.
4. Substantial user base.
5. It is compatible with various computers in different engineering labs without any further investment.
6. It has all features of a modern UNIX system like graphical user interface tool kits, networking, Databases, programming languages, debuggers.

Disadvantages:

1. Interrupts are often disabled during the course of execution of the kernel.
2. Time sharing scheduling is always a problem.
3. Virtual memory system times unpredictably.
4. Lacks in high granularity timers.

But with modification of Linux can help in overcoming these problems and making it more reliable and efficient.

1.2 Brief Introduction About Real-time Implementation of Linux :

Linux system's designing goal is to obtain its optimum average performance to serve as much as possible and achieve large throughput. The real-time operating system is required to improve performances in the worst condition, shorten interruptive time lag and provide predictable scheduling strategy. Therefore, standard Linux can't meet the needs of real-time applications in many aspects.

There are two typical solutions for Linux to achieve its hard real-time performance:

1. **Dual-core structure:** in this way, two system kernels are working and coordinate together on the same hardware platform. One kernel provides accurate real-time and multi-task management, and the other kernel provides complex non-real-time common functions.
2. **Modifying the kernel of Linux directly:** the common characters of this solution are that according to the real-time performance defect of Linux mentioned above, it can modify the kernel directly to enhance its real-time performance. As for real-time performance, it provides real-time priority-based scheduler

1.3 Requirements for Good RTOS :

- 1) **Multitasking Capabilities:** A RT application is divide into multiple tasks. The separation into tasks help to

keep the CPU busy.

2) Short Interrupt Latency:

Interrupt Latency = Hardware delay to get interrupt signal to the processor + time to complete the current instruction + time executing system code in preparation for transferring execution to the device interrupt handler.

3) Fast Context switch: The OS recognizing that the awaited event has arrived and the beginning of the waiting task is called context switch time. This switching time should be minimum.

1.4 Scheduling in RTOS :

- More information about the tasks are known
 - √ Number of tasks
 - √ Resource Requirements
 - √ Execution time
 - √ Deadlines

1.6 Scheduling algorithms in RTOS :

- √ Clock driven scheduling.
- √ Weighted Round Robin scheduling.
- √ Priority scheduling.

Conclusions

The goal of thesis were obtained by study of

basic concept about real time operating system, study of Linux kernel and evaluating its performance when dealing with real-time tasks along with the study of all other real time operating system with their capabilities. Implementation of the test method in the system with the assistance of a real-time module and evaluation of the results of the testing method helped it further. Study of approaches suggested for enhancing the real-time performance of Linux OS.

References

1. Ingo molnar, linux low latency patch for multimedia applications. <http://people.redhat.com/mingo/lowlatency-patches/>. [Online; accessed October-2013].
2. A. C. Heursch, D. Grambow, A. Horstkotte, and H. Rzehak. Steps towards a fully preemptable linux kernel. (2003).
3. C. Scordino and G. Lipari. Linux and real-time: Current approaches and future opportunities. In ANIPLA International Congress, Rome (2006).
4. Real-Time Linux – What was/is expected? (Assignment: 6 Patel, Shital (900-01-9649))