International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-2 Issue-2, January 2013

Performance Analysis of Embedded Linux in Embedded System

Charulata Ingle

Abstract- Nowadays, embedded systems with Linux OS are commonly applied in automotive industry. This paper represents two features of embedded system namely multithreading and semaphore in embedded linux. The flow charts are used to represent the program written in 'C', run on linux operating system .In multithreading the same string is printed continuously till you print 'stop' while by using semaphore the string will print only once and asking for another one. By using this we can build our own embedded system using linux as the kernel and freely available open source.

I. INTRODUCTION

1.1 Linux

The history begins in the late 1960s, when a concerted effort to develop a new OS techniques occurred. In 1968, a consortium of researchers from General Electric, AT&T Bell Laboratories carried out a OS research project name as MULTICS (Multiplexed Information & Computing Service). MULTICS incorporated many new concepts in multitasking, file management, and user interaction.

UNIX: In 1969, Ken Thomson, Dennis Ritchie, & researches at AT&T Bell Laboratories developed the Unix OS, incorporating many features of *MULTICS* research project. Unix was an affordable, efficient multi-user & multitasking OS. The first versions of Unix were distributed free to the Computer Science departments of the University of California, Berkeley.

Originally designed specifically for Intel-based personal computers, Linux started out as a personal project of a computer science student named Linus Torvalds at the University of Helsinki. At that time students were making use of a program called Minix, which highlighted different Unix features. Minix was created by Prof. Andrew Tanenbaum and widely distributed over the Internet to student around the world.

Linus's intention was to create an effective PC version of Unix for Minix user. It was named Linux & in 1991.

1.2 Why Linux as Embedded OS?

Royalty free licensing Reliable IP stack and TCP/IP Applications Source code for the OS Kernel is Open Source code for the Tool chains is Open.

It is very modular in nature, since all features of the system that are not needed for a specific embedded system can be removed from the kernel. In addition, Linux has been ported successfully to a large number of processor architectures, which allows it to run on many different types of CPUs.

Manuscript published on 30 December 2012.
*Correspondence Author(s)
Charulata Pravin Ingle K.J Somaiya Polytechnic, Computer Department,Vidyavihar East , Mumbai India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license http://creativecommons.org/licenses/by-nc-nd/4.0/

II. LINUX SYSTEM

User Commands	
Shell	
Kernel	File System Device Drivers
Hardware	

User Commands

User commands includes executable programs and scripts. The Linux command line interface consists of single line into which we enter commands with any of their options and arguments. Bash shell is a default shell.

Common commands

Pwd, cd<*dir*>,ls,ls –l,cp,mv, rm, mkdir, rmdir, who, whoami, ps, ps, echo

Shell

The shell interprets user commands. It is responsible for finding the commands and starting their execution. Several different shells are available.

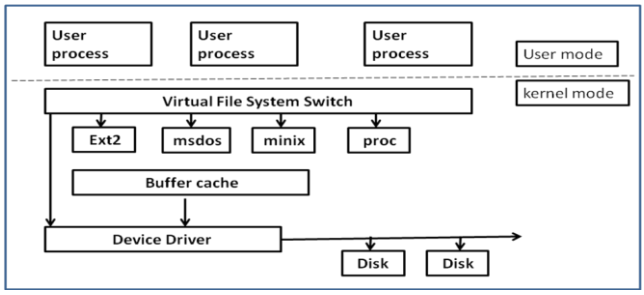
Type of shells:-

- The Borne again BASH shell.
- Public Domain Korn shell (PDKSH).
- TCSH shell.
- Z shell.

Kernal

The kernel manages the hardware resources for the rest of the system.

Linux File System:



III. PROGRAM FLOW CHART

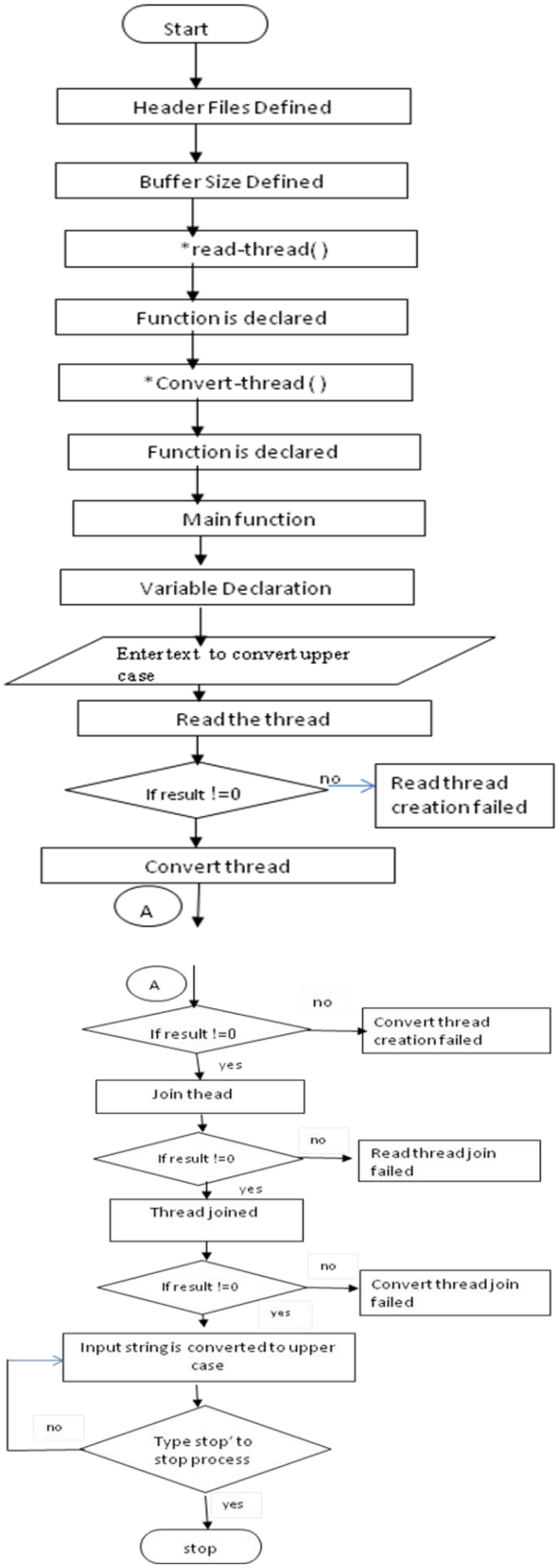
Multithreading

The word **multithreading** can be translated as **multiple threads of control** or **multiple flows of control**. While a traditional UNIX process always has contained and still does contain a single thread of control, multithreading (MT) separates a process into many execution threads,

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

each of which runs independently.Multithreading your code can improve application responsiveness, use multiprocessors more efficiently, improve program structure, and use fewer system resources.

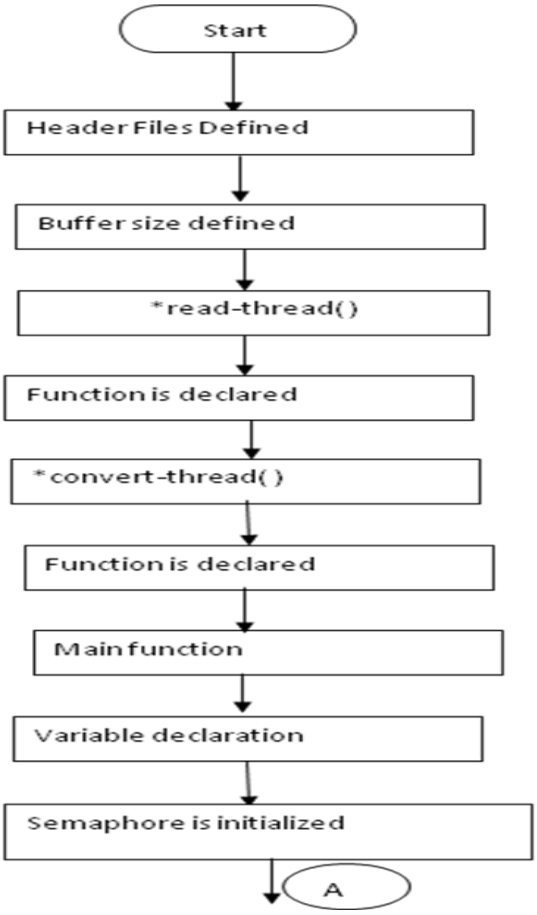
Flowchart



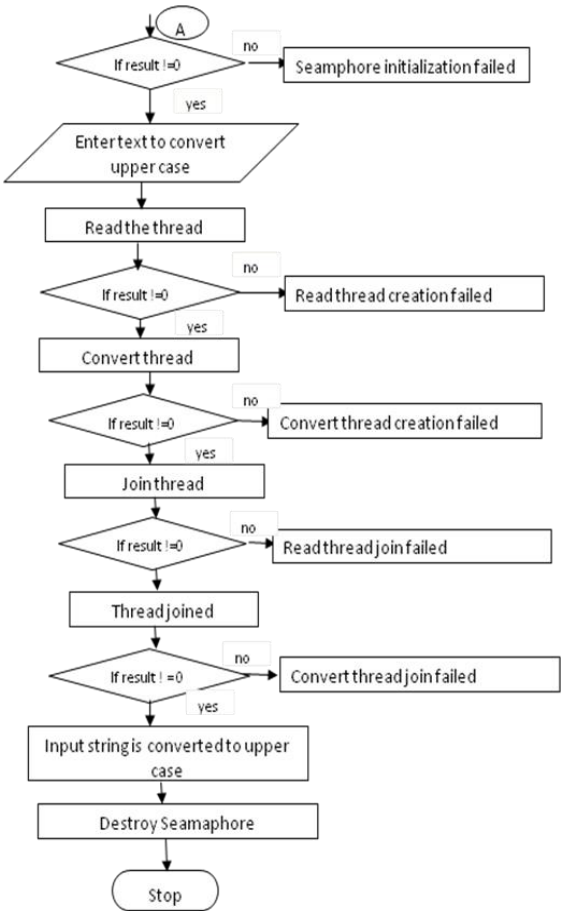
Semaphore

Semaphore is a special helper object with very simple logic which is typically used to synchronize the execution of concurrent threads. A semaphore object can be considered as an integer value which has one additional feature: if its value is 0, an attempt to decrement this value will cause the calling thread to "hang" until some other thread increments it. "Hanging" on the semaphore means entering effective wait state and consuming no or little CPU time, depending on the operating system.

Flowchart



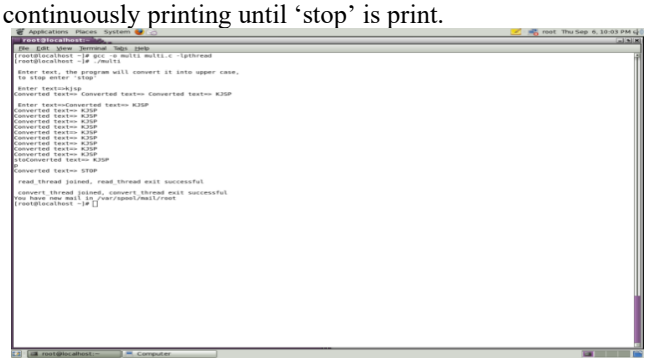
Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.



IV. RESULTS

Multithreading:

We can observe the results of multithreading. The string is continuously printing until 'stop' is print.



Semaphore:

We can see the results are changed. By using semaphore the string prints only once and asking for new one.

V. ACKNOWLEDGMENTS

The author would like to thank Mrs.Rajashree Daryapurkar for guiding to write this paper along with Mrs. Rupali Patil, the colleges K.J.Somaiya Polytechnic and K.J.Somaiya college of Engineering for availing the required software for testing.

REFERENCES

1. Tatsuo Nakajima, Masatoshi Iwasaki-IEEE Paper- Issues on linux making predictable Proceedings of the 2002 Symposium on Applications and the Internet (SAINT.02w) (0-7695-1450-2/ $17.00 © 2002 IEEE)
2. Chun-yue Bi, Yun-peng LiuIEEE Paper-Research of Key Technologies for Embedded Linux Based on ARM -201O International Conference on Computer Application and System Modeling (ICCASM 2010) (978-1-4244-7237-6/10/$26.00 ©201O IEEE)
3. Uday Shankar Macha- Embedded Linux Operating System
4. Dr. K.V.K.Prasad-Embedded system/Real Time operating systems
5. Christopher Hallinan Embedded Linux Primer
6. Karim Yaghour -Building Embedded Linux
7. http://www.embeddedtux.org/
8. http://www.embedded-computing.com/ssi/print/print.php

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.