

## Global Journal of Advanced Engineering Technologies and Sciences

### A NEW ALGORITHM FOR BEAGLEBONE BLACK

K.Tamilselvan\*, Dr.P.Thangaraj

\*Assistant Professor, Department of ECE Nandha Engineering College, Erode, Tamilandu.  
Professor & Head Department of CSE Bannariamman Institute of technology  
Sathy, Tamilnadu.

---

#### Abstract

The real time application of an embedded Linux is essential in the area of device driver platform. Device driver plays a vital role of both hardware and software. For making these terms in bridge connectivity for the purpose of reliable data transfer Multi-tasking is most important to run the task in perfect scheduling process with a minimized time. Kernel module requires initial boot loader for time requirement here start up time is more important. Execution parameter for hardware is reliable model, Here above model has been done by Beagle bone black board.

**Keywords:** Algorithm for Empty kernel & Algorithm For Beaglebone Black.

---

#### Introduction

BeagleBone Black is a low-cost, community-supported development platform for developers and hobbyists. Boot Linux in under 10 seconds and get started on development in less than 5 minutes with just a single USB cable. There are no licences involved in the usage of the BeagleBone Black design materials. We do not encourage the use of the board that we manufacture under the BeagleBoard.org logo in commercial products. We are not able to schedule parts and arrange for production for orders that we cannot see. Meeting demand is difficult as a result. In addition, we will make revisions to the board as we find necessary and we will not continue to make older revisions. This can result in supply and compatibility issues for those using them in a product.

#### Problem Identification

In the Modern world development real world application should have an excellent features is necessary. Regarding that multitasking, more efficiency, Time requirement, memory requirement and effective kernel development is necessary to overcome a drawback.

- GUIs take up a much larger amount of hard disk space than other interfaces.
- Multitasking system is absent
- They need significant more memory RAM to run than other interface types. Both code and data
- They can slow for experienced programmers to use. These people often find CLI interfaces faster than to use.
- More time is required for allocate individual application.
- Not able to execute multitasking section in dual performance.
- Flexibility is more.

#### Existing System

Existing system microcontroller will be configured RTOS code. There will not have a sufficient memory for a large code. Microcontroller not able to support for multitasking and scheduling process. Not able to run big data.

#### Drawbacks

- Boot loader size is previous size of 35 kb above this size It having unwanted tools like i2c tools, Memtest, etc..
- Kernel size was 2 MB it is a default allocation in beagle bone black.
- Boot time is more than 30 seconds. Here unwanted drivers run in the time of execution.
- Real time kernel patch have not the option of multi-threading.
- More drivers are required to execute.
- Limited Number of users can access the Wi-Fi

**PROPOSED SYSTEM**

The main objective of the system,

- To implement a pure kernel system in an Empty manner for creates an efficient platform for device driver.
- To make and configure they image data and beagle bone setup in terminal window.

**Advantages in Device driver Platform**

Fundamental layer executes data section between abstraction layer and application layer. The full C library can be linked in. The driver can perform many exotic tasks without resorting to external programs (the utility programs implementing usage policies that are usually distributed along with the driver itself).The programmer can run a conventional debugger on the driver code without having to go through contortions to debug a running kernel.

- If a user-space driver hangs, you can simply kill it. The driver can perform exotic tasks. Problems with the driver are unlikely to hang the entire system, unless the hardware being controlled is *really* misbehaving.
- Programmer can run conventional debugger on driver code.
- User memory is swappable, unlike kernel memory. An infrequently used device with a huge driver won't occupy RAM that other programs could be using, except when it is actually in use.
- A well-designed driver program can still allow concurrent access to a device.

**Algorithm for Empty kernel**

In Linux operating system will able to execute the instructions in the terminal window. Here various parameter and command sets will run in the terminal window. Creating a directory setup updating the essential packages. Then install Yocto project simulator tool is prospective manner from the company website. Formation of an image in the minimal execution method. Device driver bridge module will be created by runqemux module. More executed parameter run on the executed terminal window.

- go to terminal and connect to internet
- sudo apt-get update
- sudo apt-get install build-essential
- git clone -b dylan  
git://git.yoctoproject.org/poky.git
- cd poky ( getting into the folder of yocto)
- source oe-init-build-env build-tamil-arm-simulation (creating a build directory in the name of yours)
- cd build-tamil-arm-simulation (getting into that directory)
- cd conf (getting into conf directory)
- geditlocal.conf ( editing the file )
- change the line following
- MACHINE ??= "qemux86" to MACHINE  
?= "qemuarm"

**Comparision**

*Table 1.1 Comparison of Parameters*

PARAMETERs	EXISTING SYSTEM	PROPOSED SYSTEM
Boot loader size	40 KB	32 KB
kernel size	2MB	1.5MB
boot time	30 Sec	25 Sec
Threading	Single Thread	Multi thread
No of Devices Connectivity	Limited to 15 Devices	N number of Device Connectivity

**CONCLUSION**

Embedded Linux is an essential platform for advanced real world interfaces. Here kernel development will Executed in the idea of image formations. Various command sets are used to develop a kernel in the research idea of bit bake

executions. Here poky setup will identify directory setup respective progress. Development of kernel by various instruction of command sets. Here bit bake will helps to identify the beagle configurations in a conversion module. Here setup of a core images are configured in poky configuration of a tool. YOCTO project are used to make a simulate and analyse the hardware bridge module as a device driver section. Finally creation of an empty kernel in a reduced boot time execution.

### Future Enhancement

The kernel development can be further enhanced to real time application and implementation of software will extend to hardware. Then number of Wi-Fi device will automatic make in beagle bone configuration in extensive communication also can be introduced in the future implementation. Image formation is run by bit bake execution.

### References

1. AndiKleen "On submitting kernel patches" article 2010.
2. TODOROI "Creativity's Kernel Development for Conscience Society" InformaticaEconomicăvol. 16, no. 1/2012
3. Divya Sharma "Porting the Linux Kernel to Arm System-On-Chip And Implementation of RFID Based Security System Using ARM" Volume 3, Issue 5, May 2013
4. K.Tamilselvan "SD Card Based Data Logging And Retrieval For Microcontrollers To Using  $\mu$ c/os-II" , International journal for Engineering Research and Technology, Vol. 2, Issue 11, November 2013
5. K.Tamilselvan, G.Sivasankari, and S.Menakambal "Multitasking Operating System for ARM Processors with LED Display", International Journal of Advanced Science Engineering and Technology, Vol. 3 issue.1 No.13, March 2014.
6. P.Jammuna G.Sivasankari and K.Tamilselvan "Smart card system with high authentication technology", International Journal of Advanced and Innovative research, Volume 3, Issue 10, November 2013
7. K.Tamilselvan, Dr.A.Satheesh "Research Methodology of Kernel Development in Device Driver Platform" International Journal of Advanced and Innovative research Volume 3, Issue 10, November 2013
8. K.Tamilselvan, M.Nirubha, and Dr.A.Satheesh, "Intelligent control of Electronic Circular Using Wi-Fi Technology" International journal for Applied Engineering Research, Volume 10, No.1 2015
9. K.Tamilselvan, Dr.A.Satheesh, Dr.s.Natarajan "Real time kernel based Hot spot communication using Raspberry Pi" IJSRDI Vol 3, No.1 (Accepted).