

**DR. BABASAHEB AMBEDKAR TECHNOLOGICAL UNIVERSITY,
LONERE – RAIGAD -402 103**

Mid Semester Examination – October - 2017

Branch: M.Tech (Embedded System)

Sem.:- I

Subject with Subject Code:- Embedded Programming (MTESC102)

Marks: 20

Date:-

Time:- 1 Hr.

=====
Question 1 a) what are the architectural differences between standard Linux and Embedded linux? State the differences with reference to embedded linux architecture.

Little difference between architecture of a standard Linux system and that of an embedded Linux system based on the following aspects with respective to the embedded Linux:

1) Kernel's hardware requirements:

- 32bit CPU
- Memory Management Unit (MMU)
- Minimal amount of RAM
- ROM/flash/storage to load/mount root filesystem

2) Kernel's responsibilities:

- Drive devices
- Manage I/O access
- Manage memory
- Manage storage devices
- Control process scheduling
- Provide Unix API to applications ● etc.

3) Lowlevel interfaces:

- Very hardware specific
- Provide hardwareindependent API
- Typically handles: CPUspecific operations, architecturespecific memory ops, basic device interfaces.

4) Highlevel abstractions:

- Provide Unix abstractions (processes, files, sockets, and signals.)
 - Code implementing these abstractions is identical across all architectures (with some minor exceptions.)
- 27

5) Filesystems:

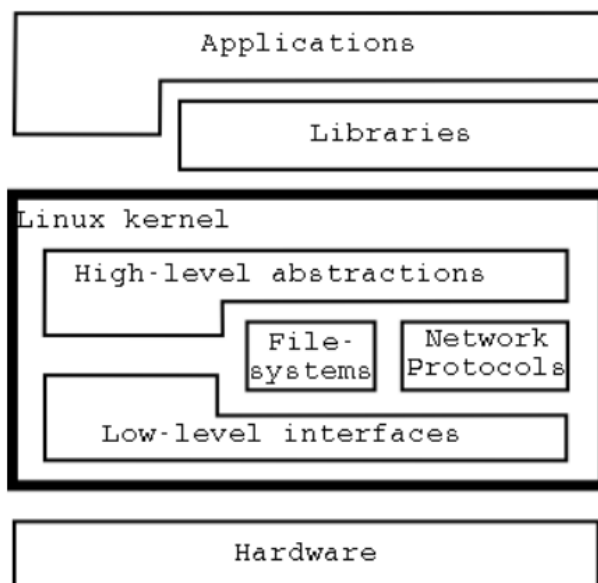
- Organize storage devices into recognizable formats
- Linux supports over 40 different filesystems
- Virtual Filesystem layer provides transparent and uniform API to all filesystems.
- Linux requires at least one properly structured root filesystem
- Networking protocols
- Organize the traffic on the wires
- Linux supports more than a dozen different protocols
- Socket API provides uniform API to all networking 28
- Librari

6) Libraries:

- Applications almost never access the Linux kernel's services directly.
- Libraries provide more abstract services than those provided by the kernel.
 - The main library used with the kernel: glibc
 - Substitutes for glibc include: uClibc and eglibc
- Dynamic linking allows only one copy of each library to be present at all times.
-

Static linking is preferable if a limited set of applications are using a limited set of a library's functionality.

Architecture of Embedded Linux is as follows:



Each aspect 1 marks (1x6) marks and architecture 2 marks

Question 1 b) Describe the specific process involved in booting with embedded Linux.

Linux boot process involves the following steps:

1. After a processor is reset, it executes ROM startup code.
2. The ROM startup code initializes the CPU, memory controller, and on-chip devices, and it configures the memory map. The ROM startup code then executes a bootloader.
3. The bootloader decompresses the Linux kernel into RAM from Flash memory or a TFTP server transfer. It then executes a jump to the kernel's first instruction. The kernel first configures microprocessor registers and then invokes `start_kernel`, which is the architecture-independent starting point.
4. The kernel initializes its caches and various hardware devices.
5. The kernel mounts the root filesystem.
6. The kernel executes the `init` process.
7. The executing `init` process loads shared runtime libraries.
8. `init` reads its configuration file, `/etc/inittab`, and executes scripts. Typically, `init` executes a startup script, `/etc/rc.d/rcS`, which configures and starts networking and other system services.
9. `init` enters a runlevel where system duties can be performed or the `login` process can start, allowing for user sessions.

Any 8 points each one marks.

Question 2 a) why Linux is mostly preferred for embedded system design?

Due to following specialities Linux is mostly preferred for embedded system design as:

Multiple choices vs. sole source
Source code freely available
Robust and reliable
Modular, configurable, scalable
Superb support for networking and Internet
No runtime licenses
Large pool of skilled developers

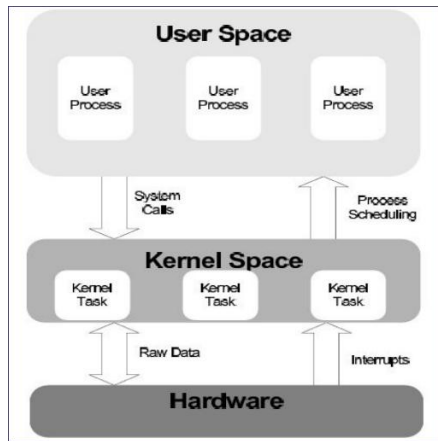
Any 4 points 1 mark.

Question 2 b) State different Open Source available Embedded Linux .

- 1) Embedded Debian Project : Convert Debian to an embedded OS
- 2) ETLinux : It is small, modular, flexible and complete
- 3) uCLinux : For microprocessors that don't have MM
- 4) uLinux (muLinux) : Distro fits on a single floppy

Each point 1 marks.

Question 2 c) How to utilize the space in Linux kernel architecture? Discuss.



Linux kernel architecture diagram 2 description 2 marks.

Question 2 d) How to Select the Most Appropriate Filesystem?

When installing Linux, the optimal selection of filesystem(s) depends on several factors, particularly the intended application for the computer(s) and the types of partitions on which they are to be installed. In the case of most computers for individual users, ext2 or ext3 (the default on many such systems) is usually quite adequate.

For the boot and root partitions, it can be advantageous to use an ext2 or ext3 filesystem because this will allow booting in an emergency even with an older kernel. For other Linux partitions, ext3 or ReiserFS are usually the best choices, the former where ext2 compatibility is emphasized and the latter where performance is paramount. When it is desired for partitions to be accessible to both Linux and Microsoft Windows, FAT should be selected.