

RaspberryPi

Raspberry pi interfacing

Lecture 1: Introduction



Course content

➤ Introduction

- Introduction to Embedded Systems.
- Introduction to Linux operating system.
- Introduction to Embedded Linux.

➤ Linux operating system

- Linux file system hierarchy.
- Linux commands.
- Files permissions.
- Input-Output redirection.
- Environment variables.
- Process management.
- Linux shell scripting.

Course content

➤ **Embedded Systems Communication Protocols**

- UART protocol.
- I2C protocol.
- SPI protocol.

➤ **Hardware interfacing special cases**

- Analog to Digital Converter (ADC).
- Pulse Width Modulation (PWM).
- Polling V.S Interrupt based interfacing.

Course content

➤ **Raspberry Pi – Hardware interfacing**

- Hardware interfacing using Python language.
- Installing Linux on the Pi.
- Setting a static IP address and local host name for the Pi.
- Configuring the Wi-Fi dongle.
- Working with Pi GPIO.
- Working with Pi UART.
- Working with Pi PWM.

Introduction Agenda

➤ **Introduction to Embedded Systems.**

- What is an embedded system ?
- Embedded systems in our life.
- Embedded systems characteristics.
- Hardware, Software Co-Design.
- Micro-Controller concepts.
- Embedded systems market in Egypt.

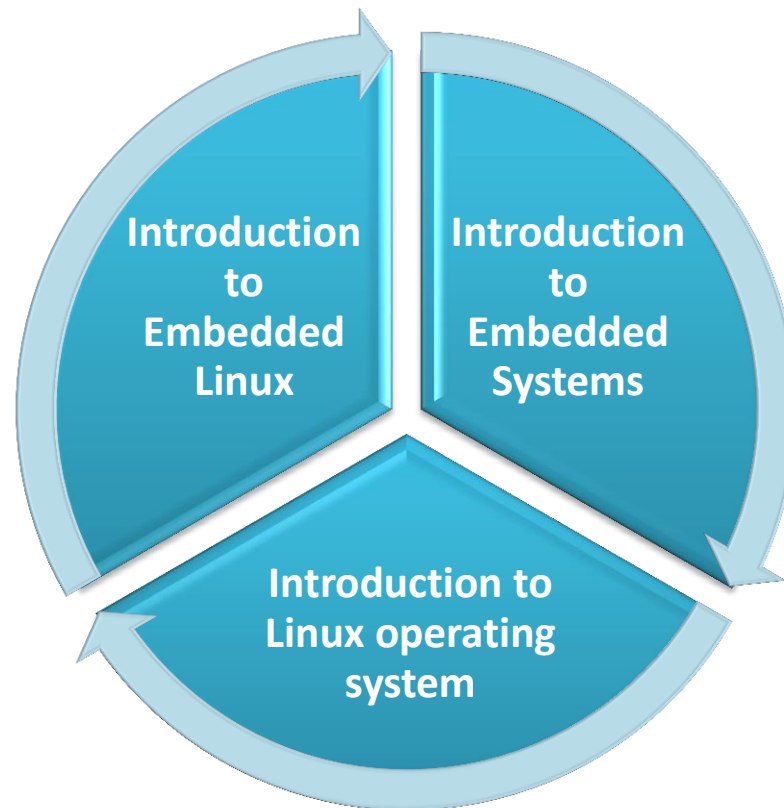
➤ **Introduction to Linux operating system.**

- What is an operating system ?
- Why to use an operating system ?
- What is Linux operating system ?
- Linux OS main components.
- Linux features.

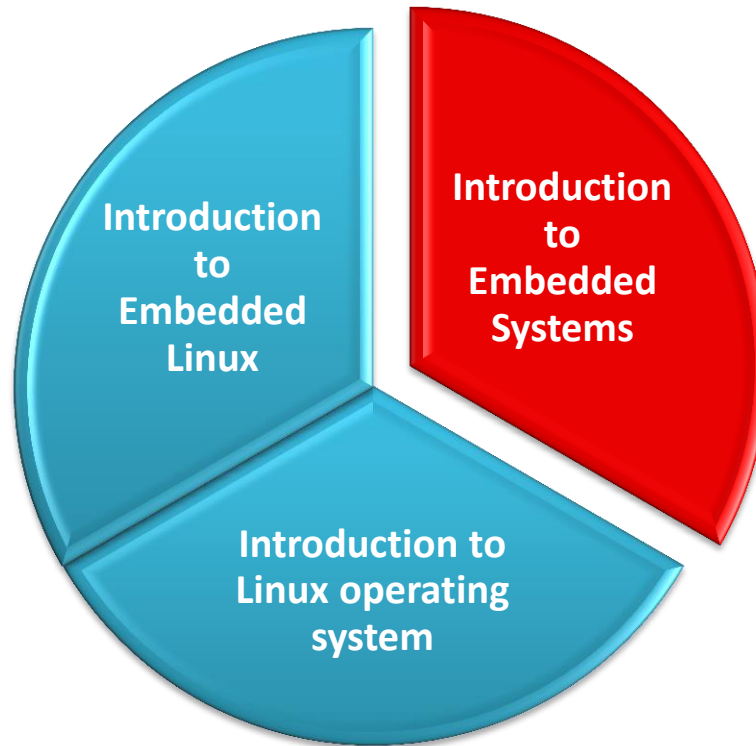
➤ **Introduction to Embedded Linux.**

- What is an embedded Linux ?
- Why to use embedded Linux ?
- Embedded Linux main components.

Introduction Agenda



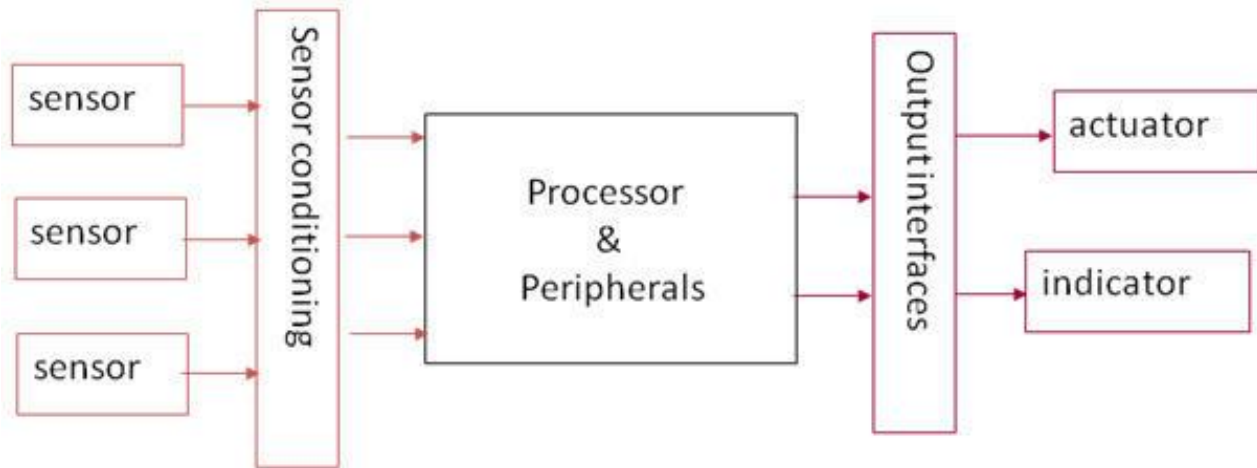
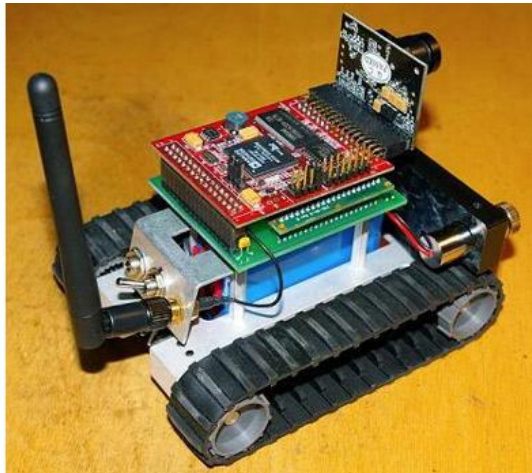
Introduction to Embedded Systems



Introduction to Embedded Systems

➤ What is an embedded system ?

An embedded system is a computer system with a dedicated function within a larger mechanical or electrical system, sometimes with real-time computing constraints. It is embedded as part of a complete device often including hardware and mechanical parts.



Introduction to Embedded Systems

➤ Embedded systems in our life



Introduction to Embedded Systems

➤ **Embedded systems characteristics**

- Cost of hardware and software.
- Memory.
- Power consumption.
- Operator interface.
- Reliability.
- Maintainability.
- Security.
- Safety.
- Real time critical.
- Interface to environment through sensors and actuators.

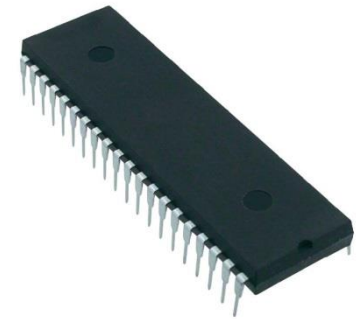
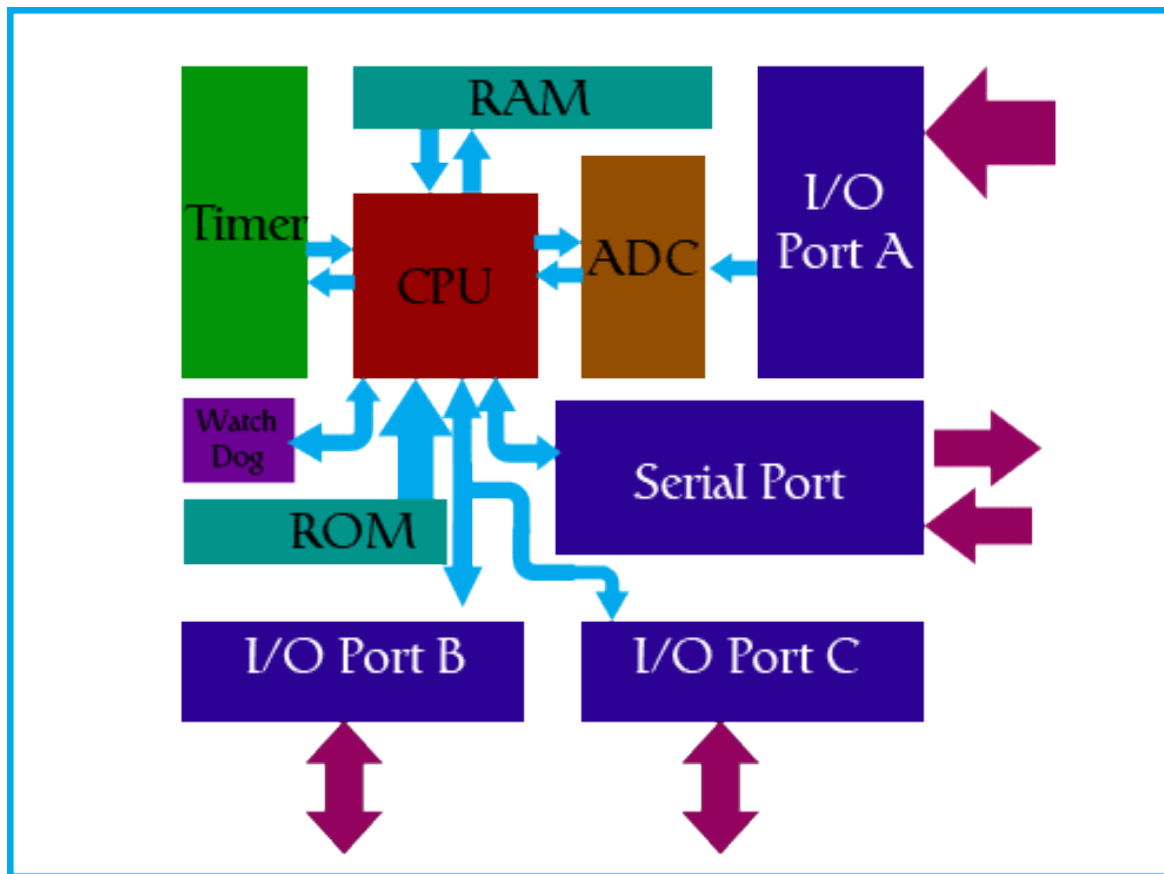
Introduction to Embedded Systems

➤ **Hardware, Software Co-Design**

- Break system into subsystems.
- Identify concurrency and dependencies in over all system.
- Allocate system components between hardware and software.
- Trade off between hardware and software.
- Break software into layers.
- Design user interface system.

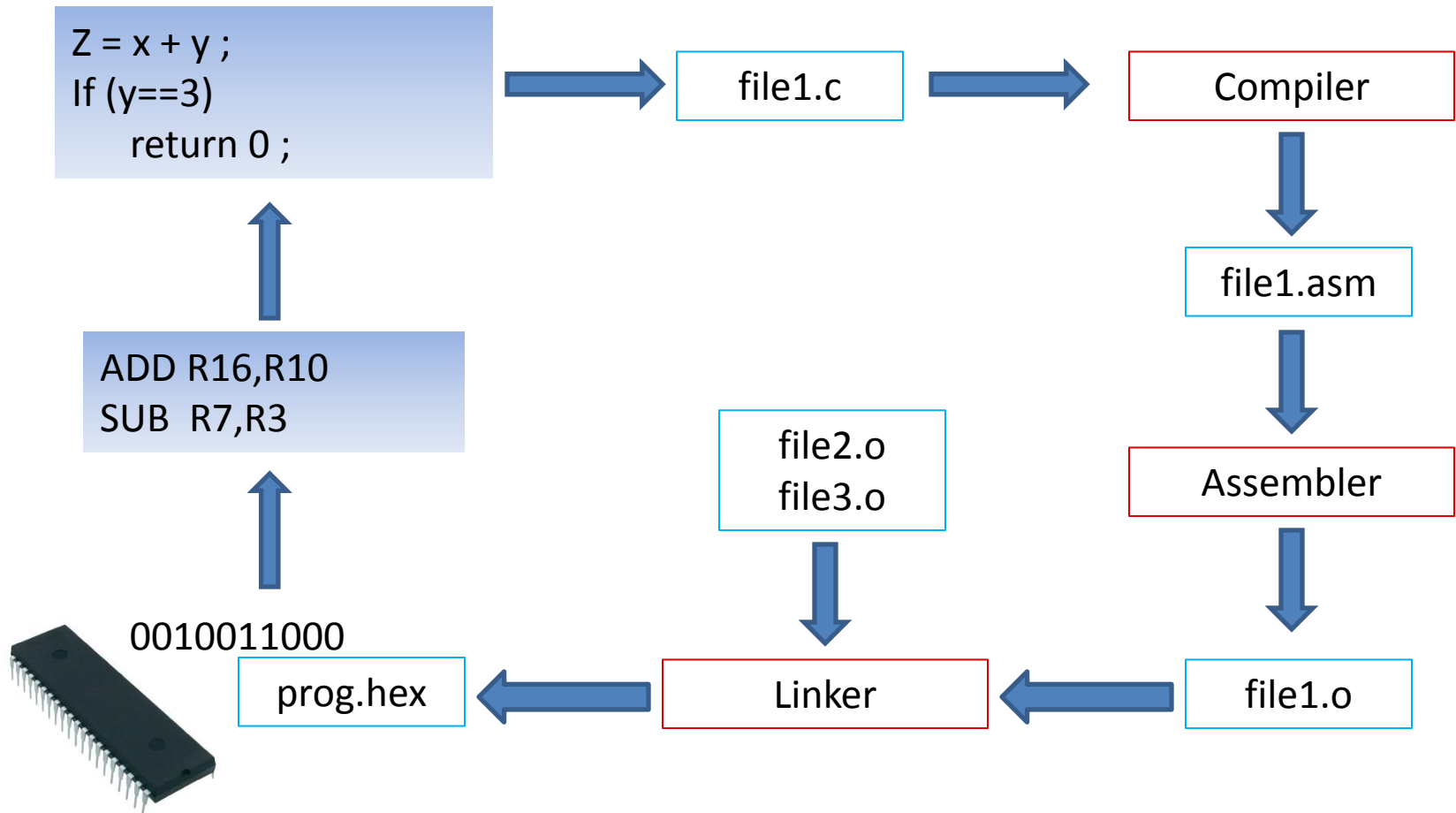
Introduction to Embedded Systems

➤ Micro-Controller concepts



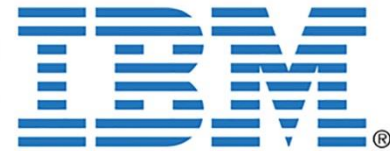
Introduction to Embedded Systems

➤ Micro-Controller concepts

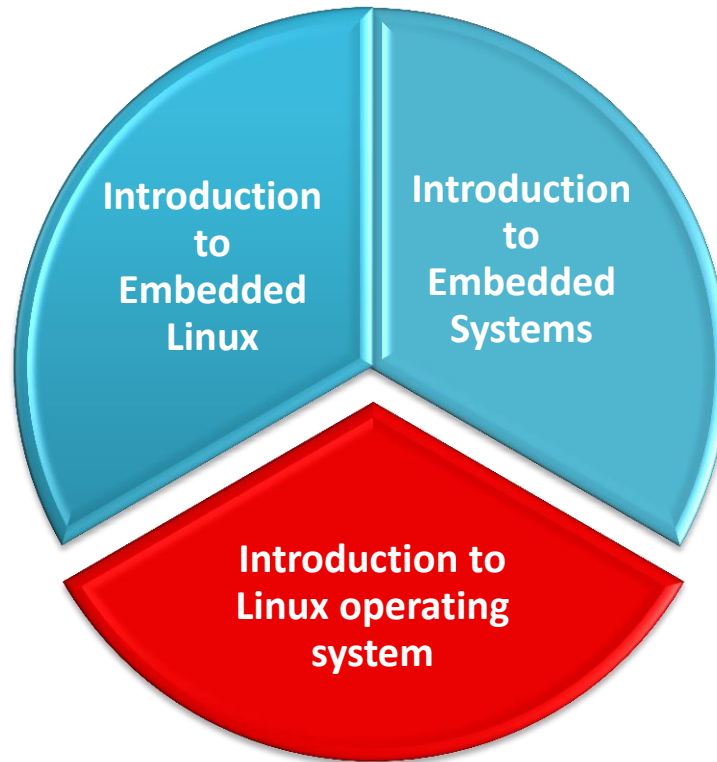


Introduction to Embedded Systems

➤ Embedded systems market in Egypt



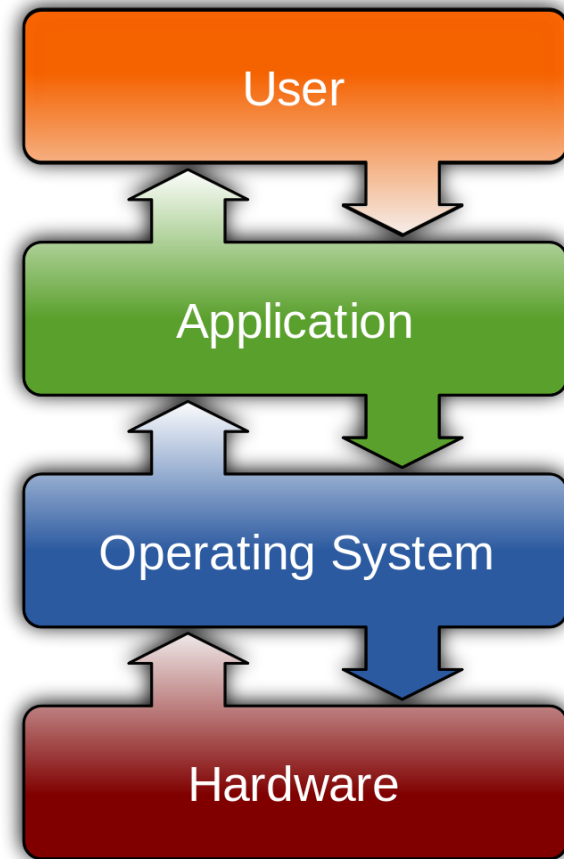
Introduction to Linux operating system



Introduction to Linux operating system

➤ What is an operating system ?

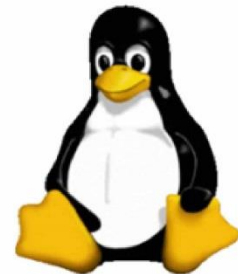
- An operating system acts as an intermediary between the user of a computer and the computer hardware. The purpose of an operating system is to provide an environment in which a user can execute programs in a convenient and efficient manner.
- An operating system is software that manages the computer hardware. The hardware must provide appropriate mechanisms to ensure the correct operation of the computer system and to prevent user programs from interfering with the proper operation of the system.



Introduction to Linux operating system

➤ Why to use an operating system ?

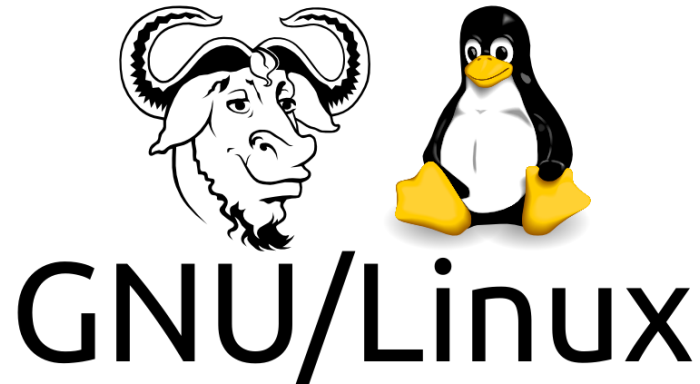
- Memory management.
- Storage management.
- Process management.
- Protection and security.
- Abstraction layer for every resource.



Introduction to Linux operating system

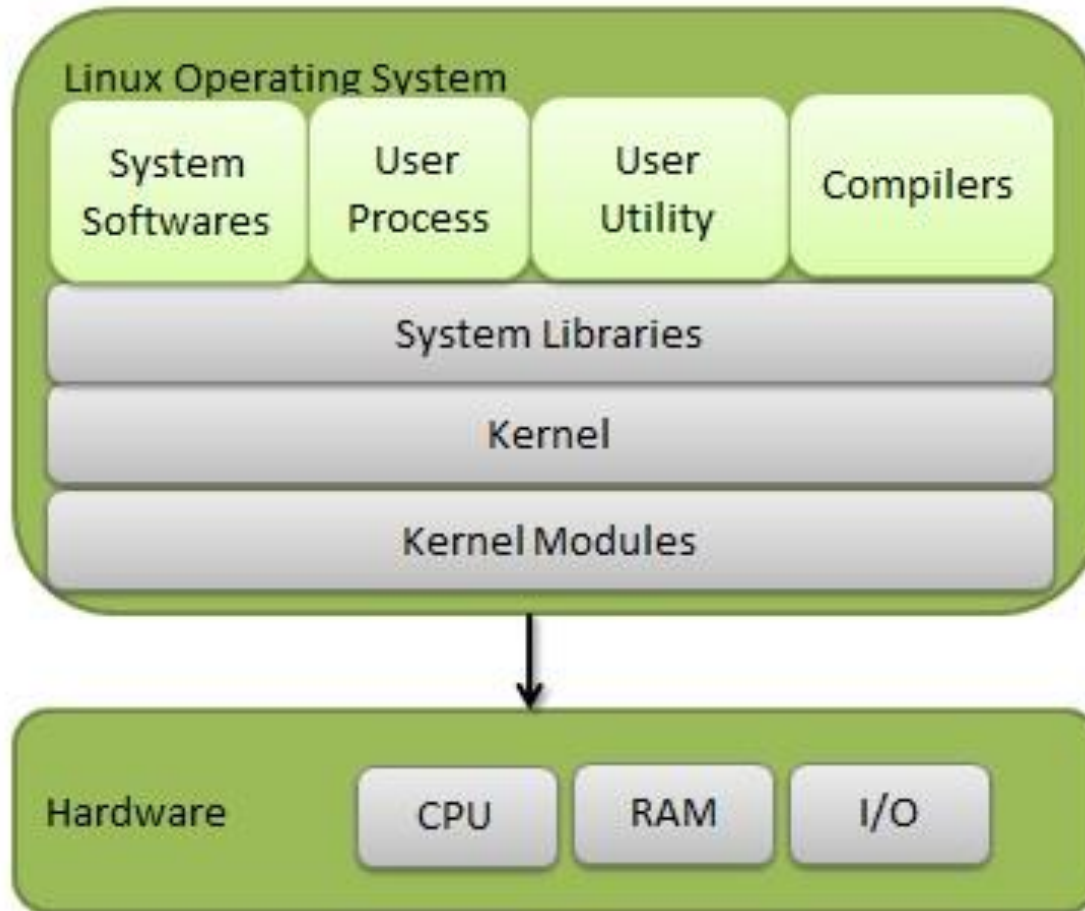
➤ What is Linux operating system ?

- In 1983, Richard Stallman, founder of the Free Software Foundation, set forth plans of a complete Unix-like operating system, called GNU, composed entirely of free software.
- By 1991 the lower level (kernel, device drivers, system-level utilities and daemons) was still mostly lacking.
- In 1991, Linus Torvalds released the first version of the Linux kernel. Early Linux developers ported GNU code, including the GNU C Compiler, to the kernel. The free software community adopted the use of the Linux kernel as the missing kernel for the GNU operating system.



Introduction to Linux operating system

➤ Linux OS main components



Introduction to Linux operating system

➤ Linux features

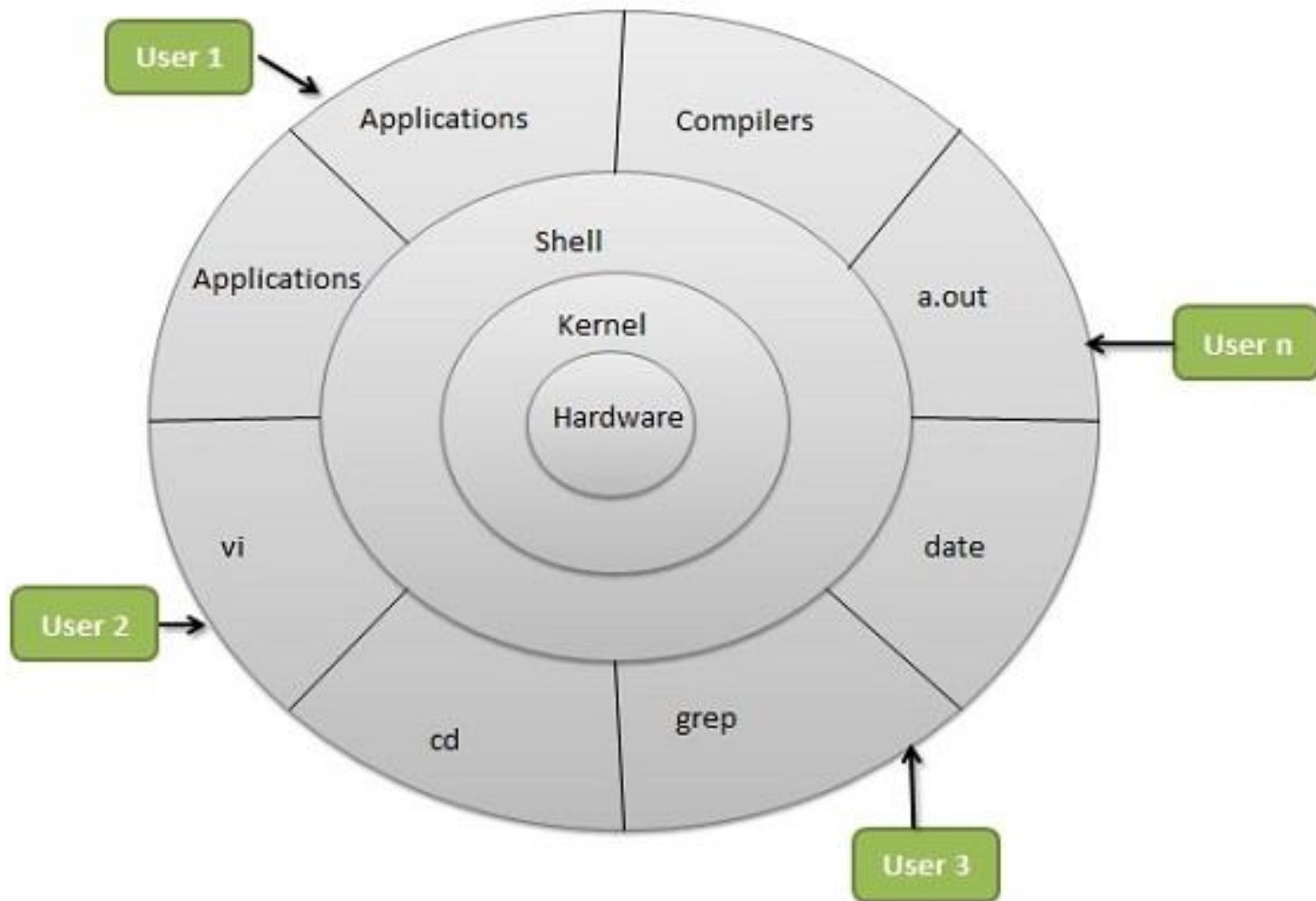
- **Open Source:** Linux source code is freely available and it is community based development project. Multiple teams works in collaboration to enhance the capability of Linux operating system and it is continuously evolving.
- **Multi-User:** Linux is a multiuser system means multiple users can access system resources like memory, ram, application programs at same time.
- **Multiprogramming:** Linux is a multiprogramming system means multiple applications can run at same time.
- **Hierarchical File System:** Linux provides a standard file structure in which system files, user files are arranged.

Introduction to Linux operating system

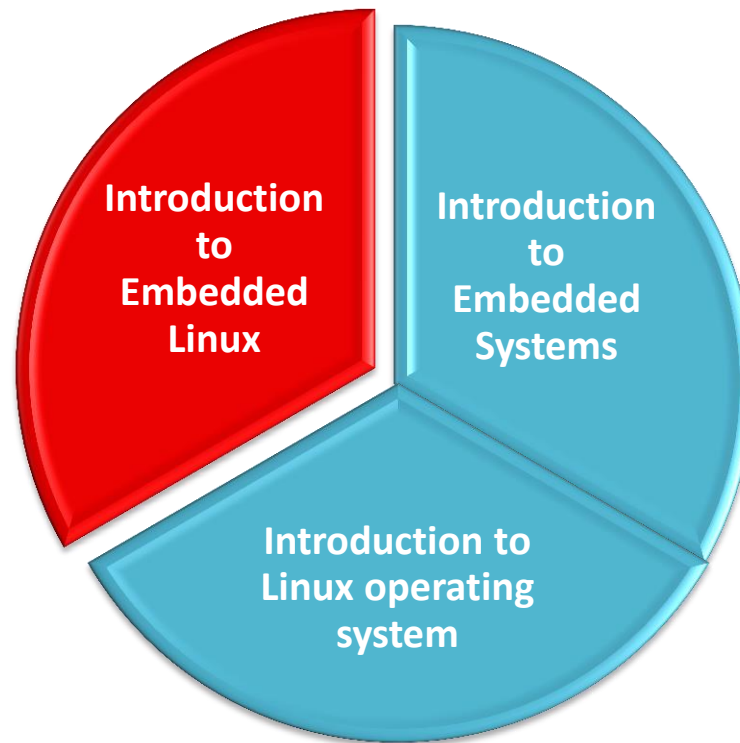
➤ Linux features

- **Shell:** Linux provides a special interpreter program which can be used to execute commands of the operating system. It can be used to do various types of operations, call application programs etc.
- **Security:** Linux provides user security using authentication features like password protection, controlled access to specific files, encryption of data.

Introduction to Linux operating system



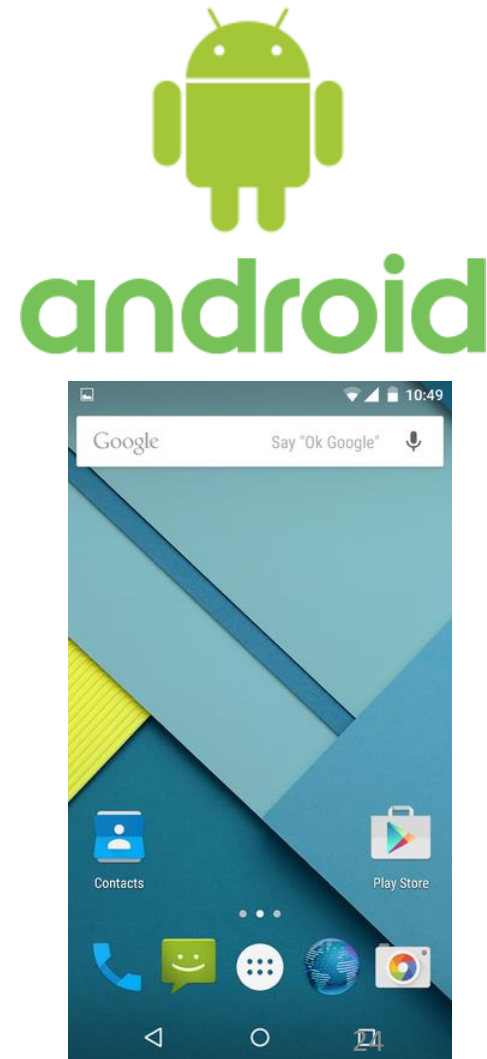
Introduction to Embedded Linux



Introduction to Embedded Linux

➤ What is an embedded Linux ?

- Operating systems based on the Linux kernel are used in embedded systems such as consumer electronics (i.e. set-top boxes, smart TVs, in-vehicle infotainment (IVI), networking equipment (such as wireless routers), machine control, industrial automation, navigation equipment, spacecraft flight software, and medical instruments in general).
- Thanks to their versatility, operating systems based on the Linux kernel can be also found in mobile devices that are actually touchscreen-based embedded devices, such as smartphones and tablets, together with personal digital assistants (PDAs) and portable media players that also include a touchscreen.



Introduction to Embedded Linux

➤ Why to use embedded Linux ?

- Linux kernel supports wide range of architectures like:
 - X86 and x64 in PC and multimedia.
 - ARM.
 - PowerPC mainly in real time applications.
 - And much more architectures.
- A very basic Linux system can work on 8 MB of RAM and 4 MB of storage.
- Linux kernel has support for many communication busses like:
 - USB.
 - SPI.
 - CAN.

Introduction to Embedded Linux

➤ Why to use embedded Linux ?

- Linux kernel has support for many networking stacks like:
 - Ethernet.
 - Wi-Fi.
 - Bluetooth.
 - IPv4, IPv6, TCP, UDP.
 - Firewall and advanced routing.

Introduction to Embedded Linux

➤ **Embedded Linux main components**

- **Cross-compilation Tool Chain:** used to compile source code for the required target hardware.
- **Boot loader:** used to initialize hardware and load the Linux kernel.
- **Linux Kernel:** the main software component of the operating system.
- **System libraries:** System libraries are special functions or programs used by application programs or system utilities to access Kernel's features.
- **File system with other libraries & apps.**