# Generic course information

# Introduction to Embedded System

# Embedded System

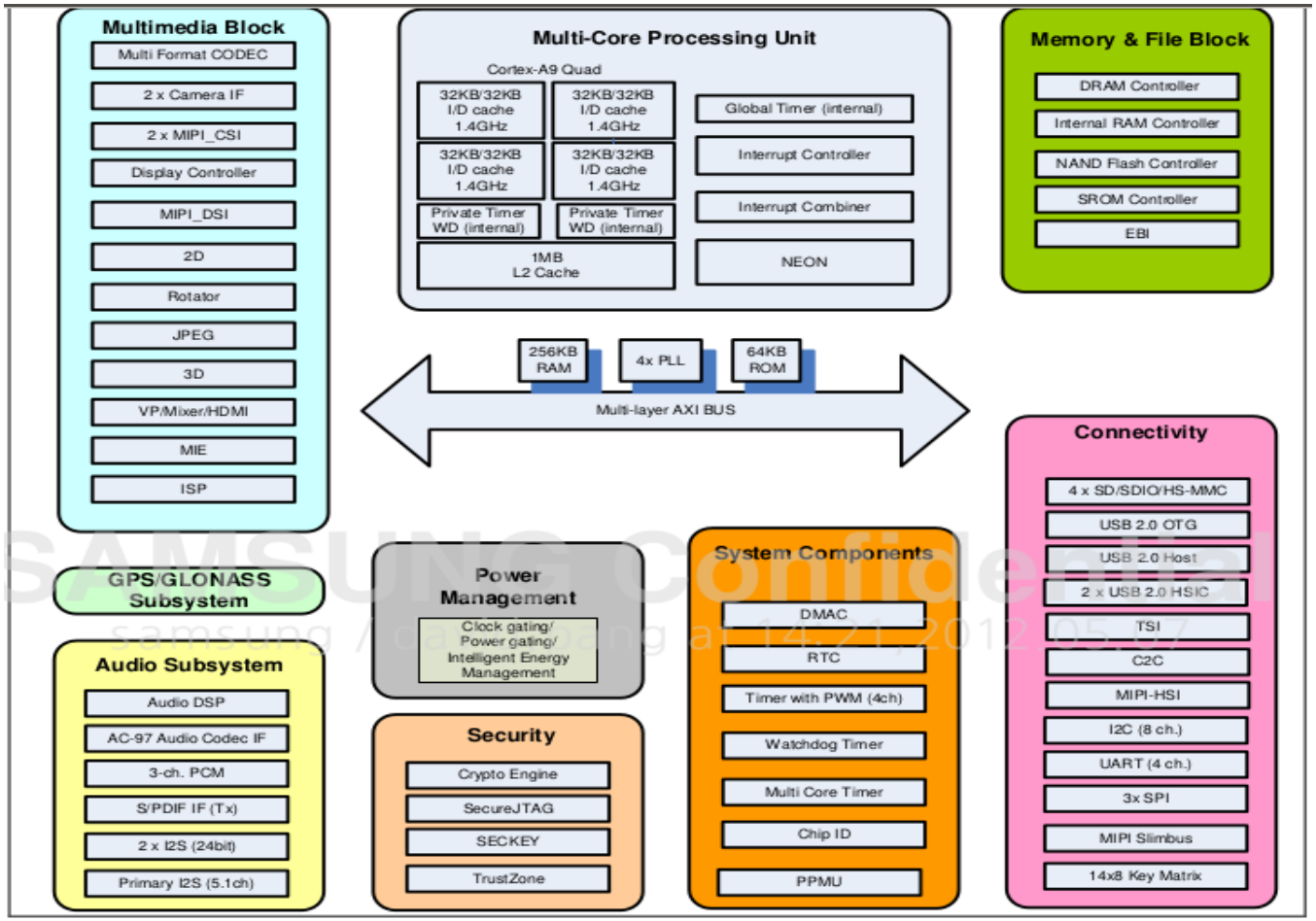
- An embedded system
  - combination of computer hardware and software
  - specifically designed for a particular function
- Applications
  - Mobile phone
  - Digital camera
  - Smart TV
  - ATM
  - Navigation system

# Feature

- Designed to do some specific task
  - Low power
  - Small size
  - Special operating ranges
  - Low cost
- Install OS ?

# SOC

- System On Chip



# Component of embedded system

- Processor
  - ARM, X86, MIPS ….
- RAM
  - 8MB ~ 32 MB
- Storage
  - Nand, Nor flash
  - SD/MMC/eMMc
- System Bus
  - AMBA, AHB, APB, AXI ...

# Component of embedded system

- Communication
  - I2C, I2S, USB, PCI/PCIe ...
- Media system
  - JPEG, H.264 ..
- System component
  - DMA, RTC ..

# Experiment EVB

## Tiny4412 | Exynos4412 ARM Cortex-A9 Board

FriendlyARM Tiny 4412 Stamp Module with 1.5 GHz Samsung Exynos4412 ARM Cortex-A9 processor.

### Specification: Stamp Module

- **Dimension:** 74 x 55 mm
- **CPU:** 1.5 GHz Samsung Exynos4412 ARM Cortex-A9 (Quad-Core)
- **RAM:** 1 GB, 32 bit Bus
- **Flash:** up to 32GB eMMC Flash
- **User Outputs:** 4x LEDs
- **Expansion headers** (2.0 mm)
- **Power:** 2-6V
- **OS Support**
  - Linux
  - Ubuntu
  - Android

# Experiment EVB

## Specification: SDK-Board

- **Dimension:** 180 x 130 mm
- **EEPROM:** 256 Byte (I2C)
- **Ext. Memory:** SD-Card socket
- **Serial Ports:** DB9 connector (RS232), RS485, total: 4x serial port connectors)
- **USB:** USB-A Host 1.1, miniUSB Slave/OTG 2.0
- **mini PCIe**
- **Audio:** WM8960 codec
- **Audio Output:** 3.5 mm stereo jack, connector for a speaker (Class D Amp)
- **Audio Input:** 3.5mm jack + Condenser microphone
- **Ethernet:** RJ-45 10/100M (DM9000)
- **RTC:** Real Time Clock with battery
- **Beeper:** PWM buzzer
- **G-Sensor**
- **Camera:** 20 pin (2.0 mm) Camera interface
- **Monitor:** HDMI
- **LCD:** 40 pin FFC and 45 pin FFC connector
- **User Inputs:** 8x buttons and 1x A/D pot
- **Expansion:** (2.0 mm)
- **Power:** regulated 5V

# Introduction to Embedded Linux

# Birth

- 1991, Linus Torvalds, Linux kernel project, a Unix-like operating system kernel.
- 2000, Linux is more and more popular on embedded systems.
- 2008, Linux is more and more popular on mobile devices
- 2010, Linux is more and more popular on phones (Android ?)

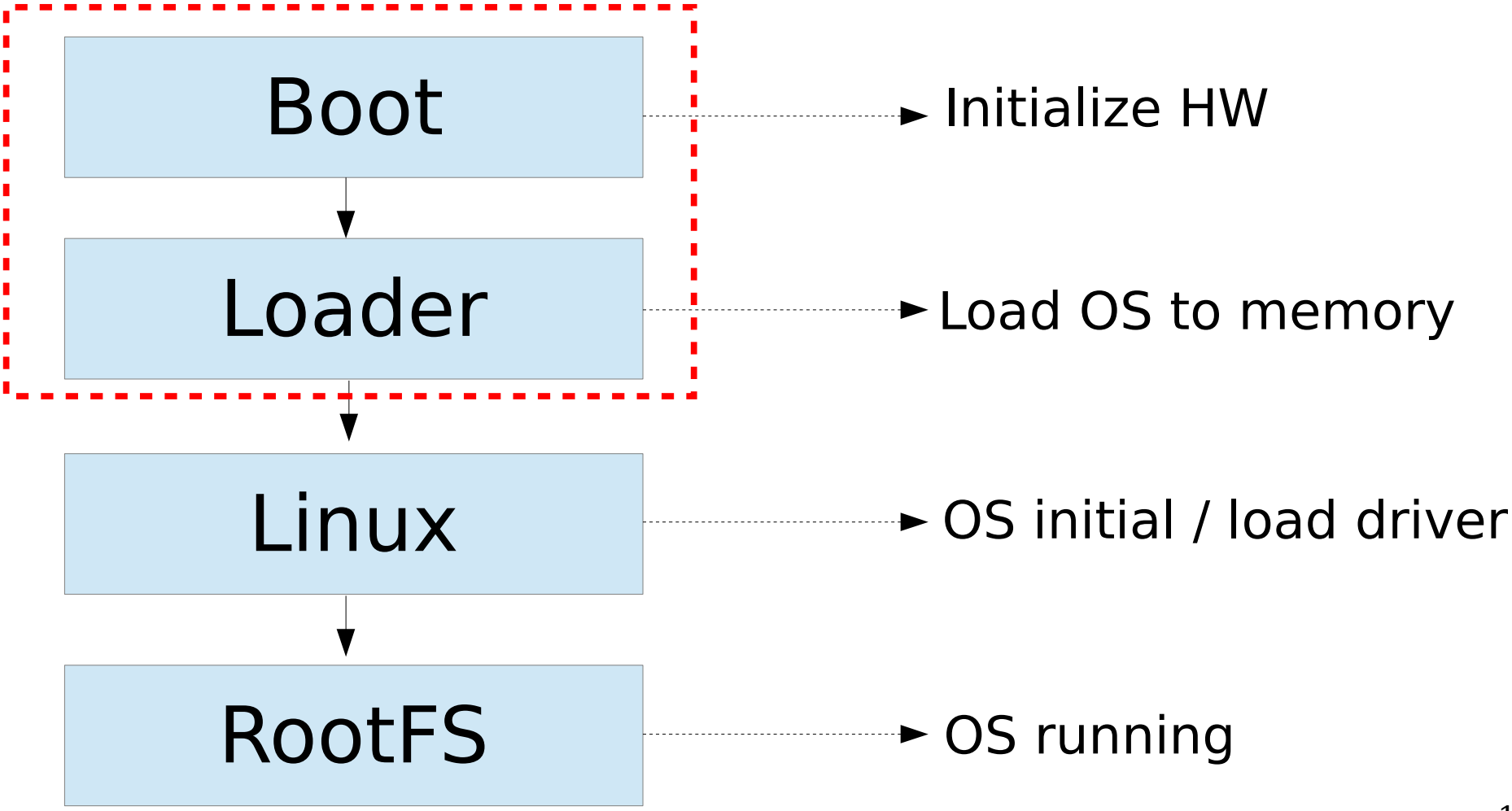
# Embedded Linux ?

Embedded Linux is the usage of the Linux kernel and various open-source components in embedded systems (from Free Electrons)

# Advantages

- Re-use components
- Quickly design and develop complicated products
- No need to re-develop components
  - TCP/IP stack, USB stack, PCI stack ...
- Allow you modify components
- Low cost (?)

# Embedded Linux System Booting



# Embedded Linux System Software components

- Cross-compilation toolchain
- Bootloader
- Linux Kernel
- Rootfs
- C library
- Libraries and applications
- BSP (Board Support Package)

# Develop Environment

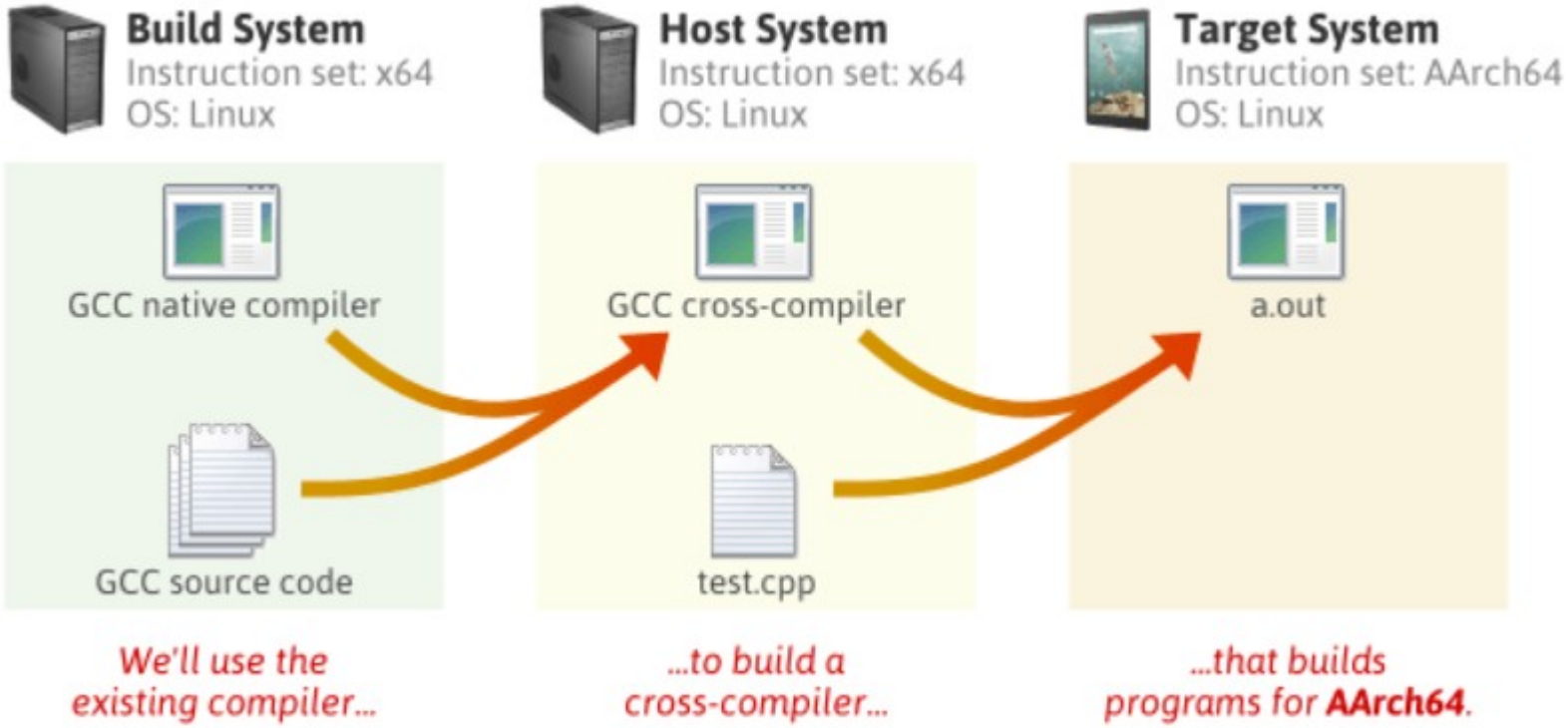
# Develop Environment

- Host PC
- Toolchain
- EVB
- BSP

# BSP

- Board Support Package
- From chip vendor
  - Bootloader
  - OS (Linux kernel)
  - Device driver
  - Shell (Android)
  - Rootfs

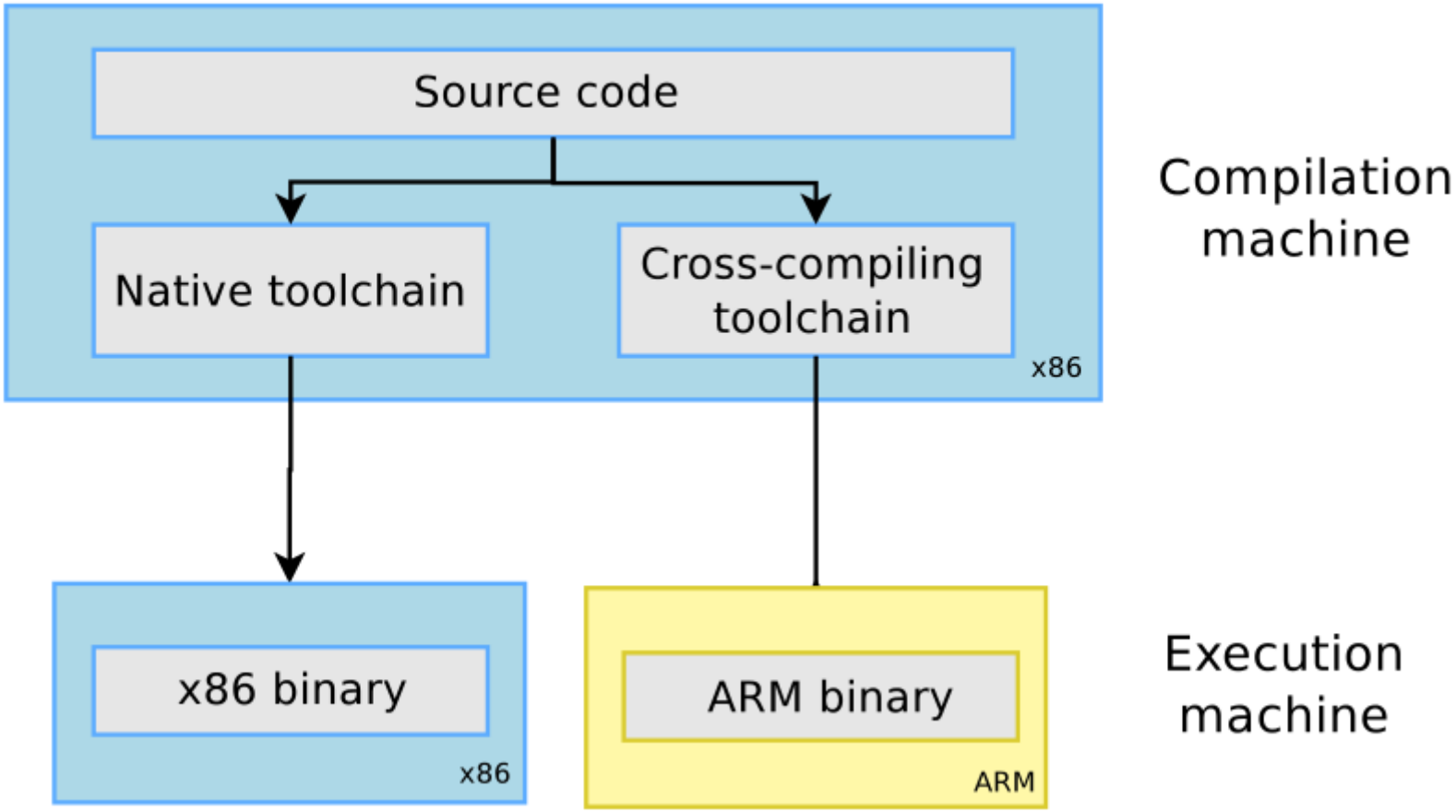
# Cross Compilation toolchain



# Cross Compilation toolchain

- The Build machine, where the toolchain is built
- The Host machine, where the toolchain will be executed
- The Target machine, where the binaries created by the toolchain are executed.
-

# Cross Compilation toolchain



# Cross Compilation toolchain


## Native build

used to build the normal gcc
of a workstation


## Cross-native build

used to build a toolchain that runs on your
target and generates binaries for the target


## Cross build

used to build a toolchain that runs
on your workstation but generates
binaries for the target

The most common case in embedded development


## Canadian build

used to build on architecture A a
toolchain that runs on architecture B
and generates binaries for architecture C

# Introduction Operating System

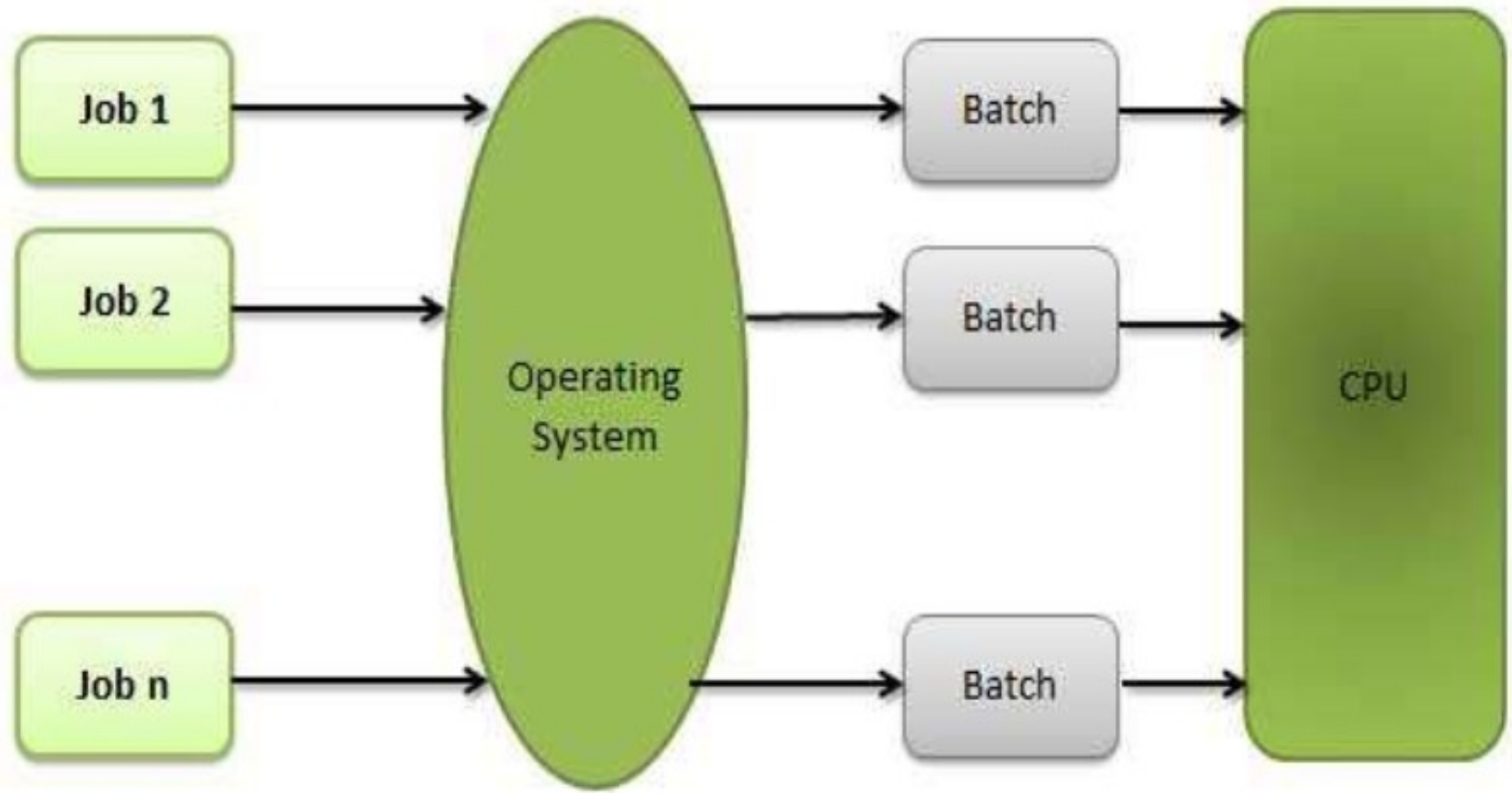
# Operating System

- Device management
- Processing management
- Memory management
- File system
- Networking
- Security
- User interface

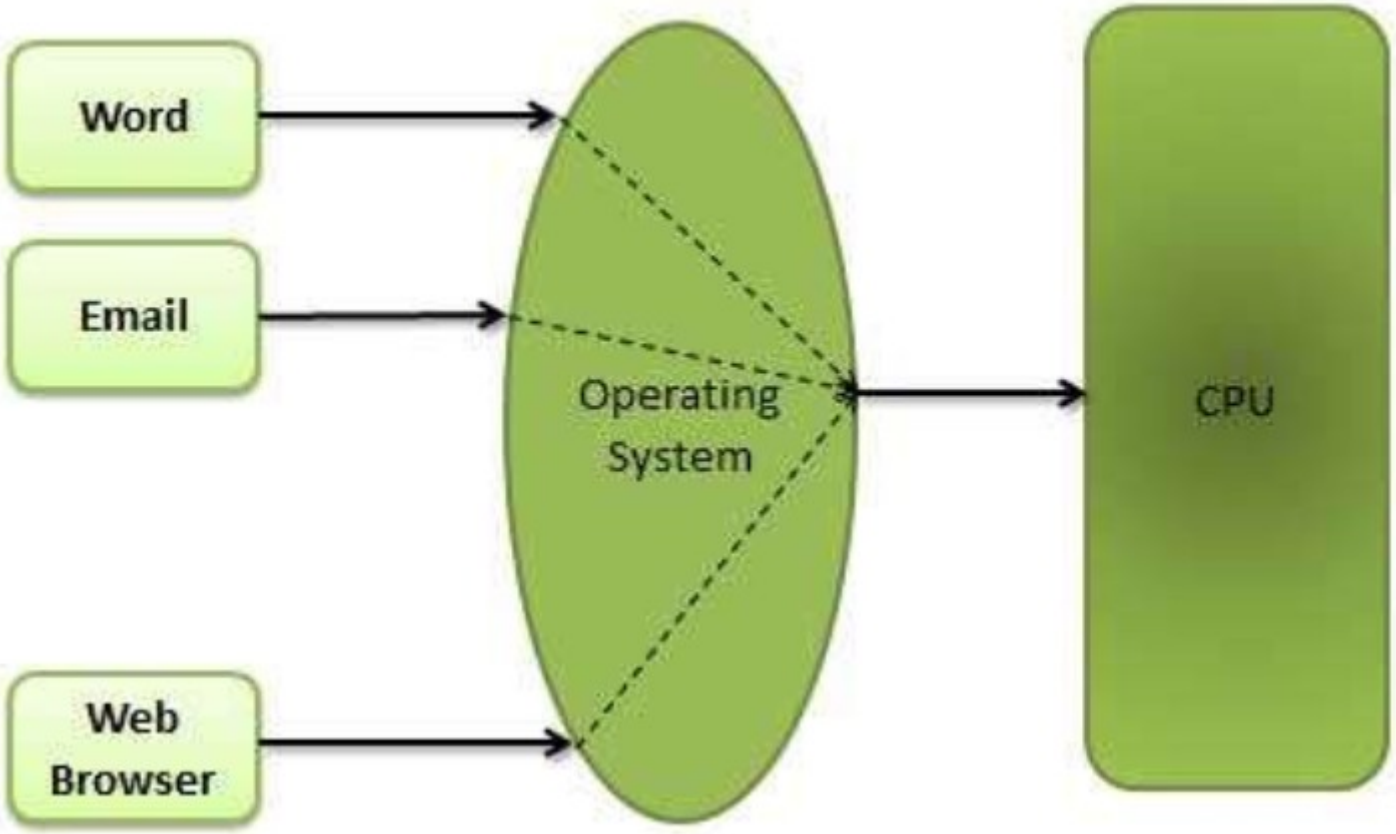
# Operating System Properties

- Batch processing
- Multitasking
- Multiprogramming
- Real Time System
- Distributed Environment
- Spooling

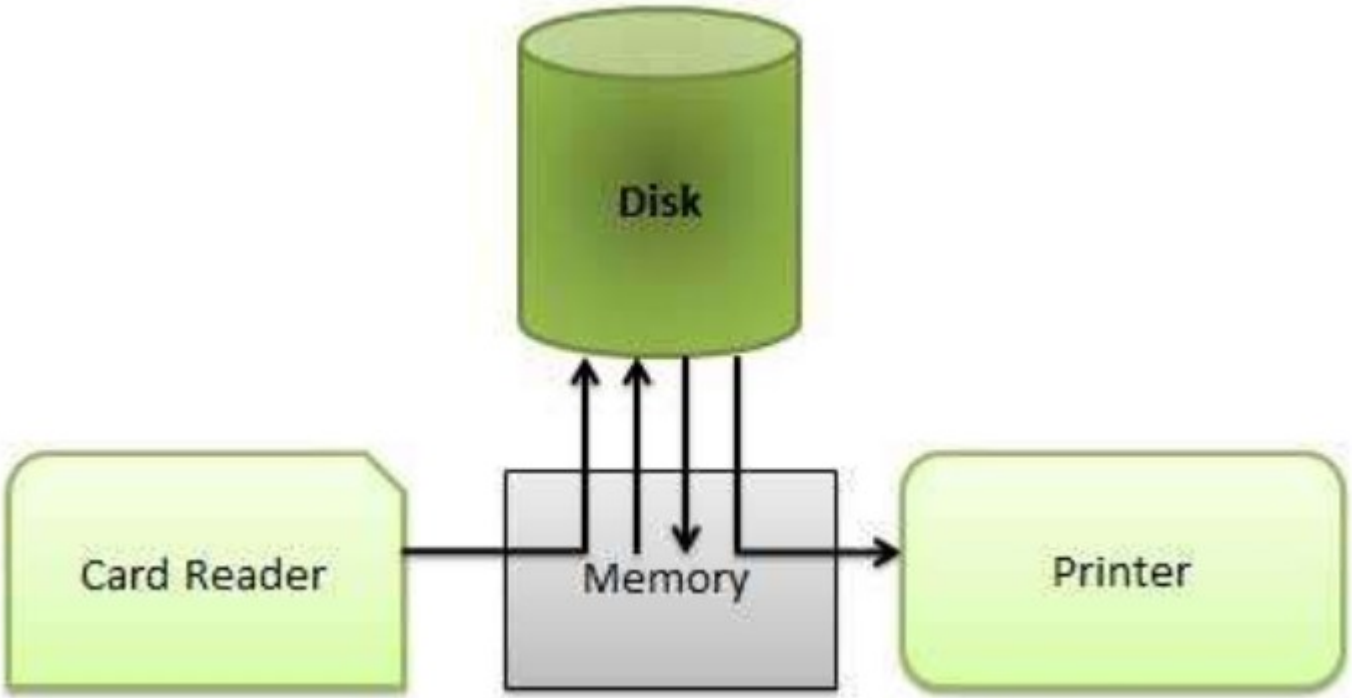
# Batch processing



# Multitasking



# Spooling

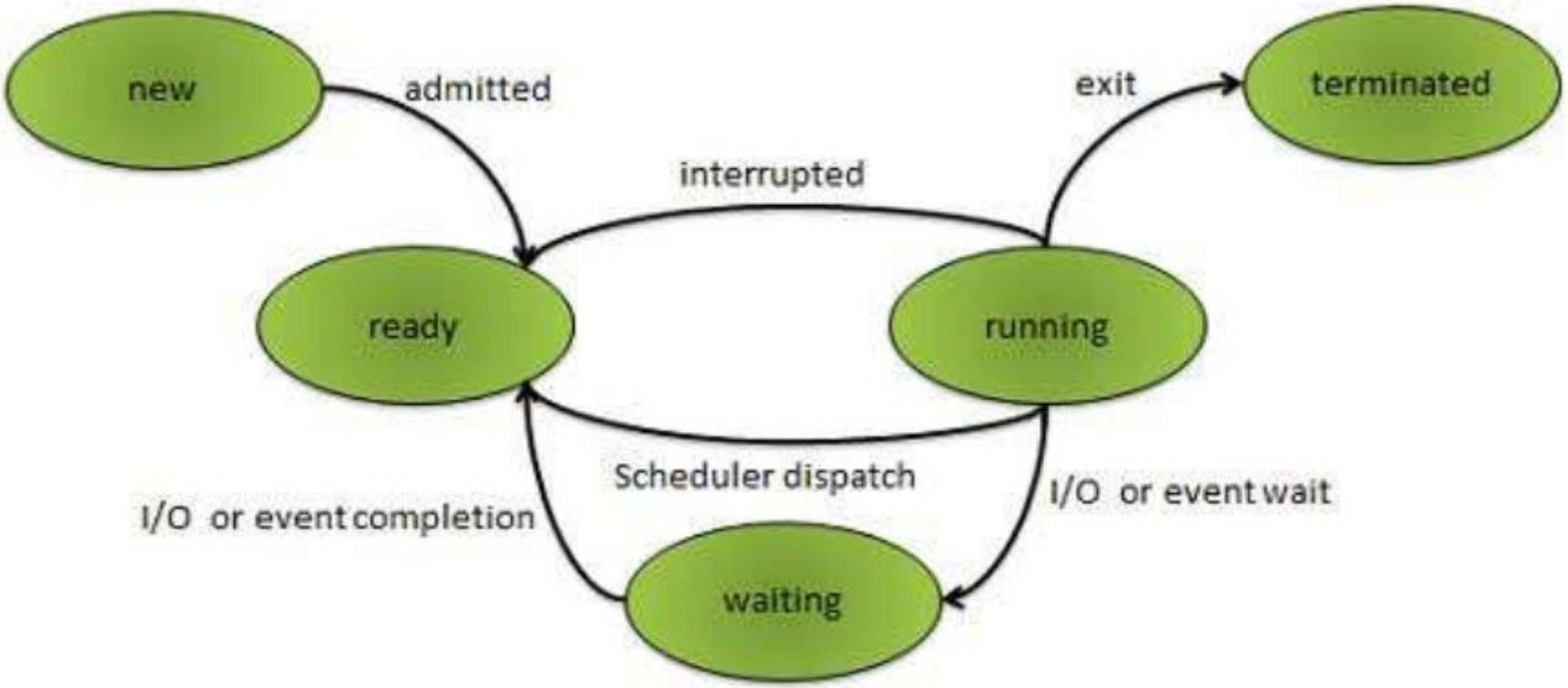


# Operating System Processes

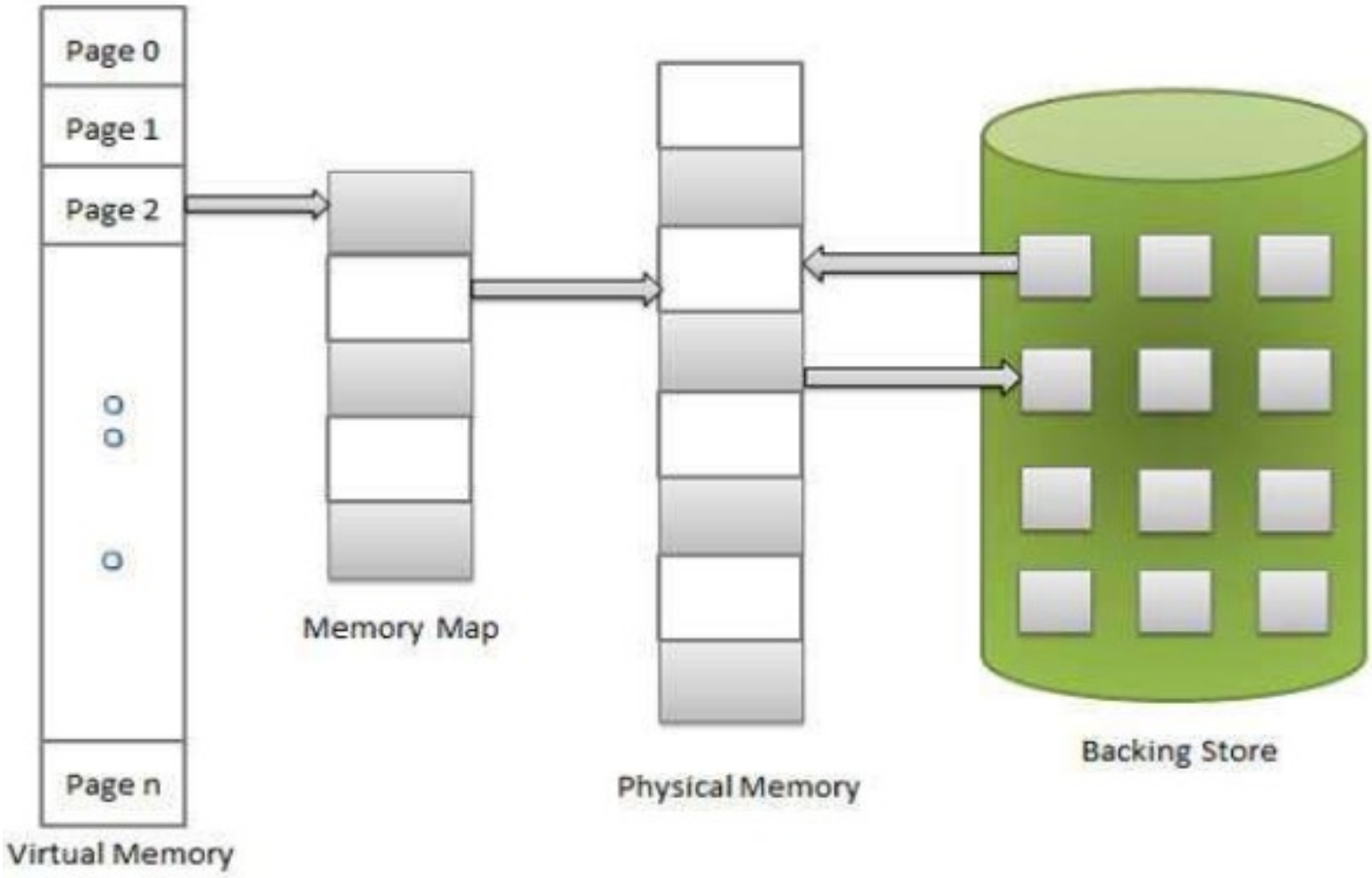
## Process States

| S.N. | State & Description |
|---|---|
| 1 | **New**<br>The process is being created. |
| 2 | **Ready**<br>The process is waiting to be assigned to a processor. Ready processes are waiting to have the processor allocated to them by the operating system so that they can run. |
| 3 | **Running**<br>Process instructions are being executed (i.e. The process that is currently being executed). |
| 4 | **Waiting**<br>The process is waiting for some event to occur (such as the completion of an I/O operation). |
| 5 | **Terminated**<br>The process has finished execution. |

# Operating System Processes



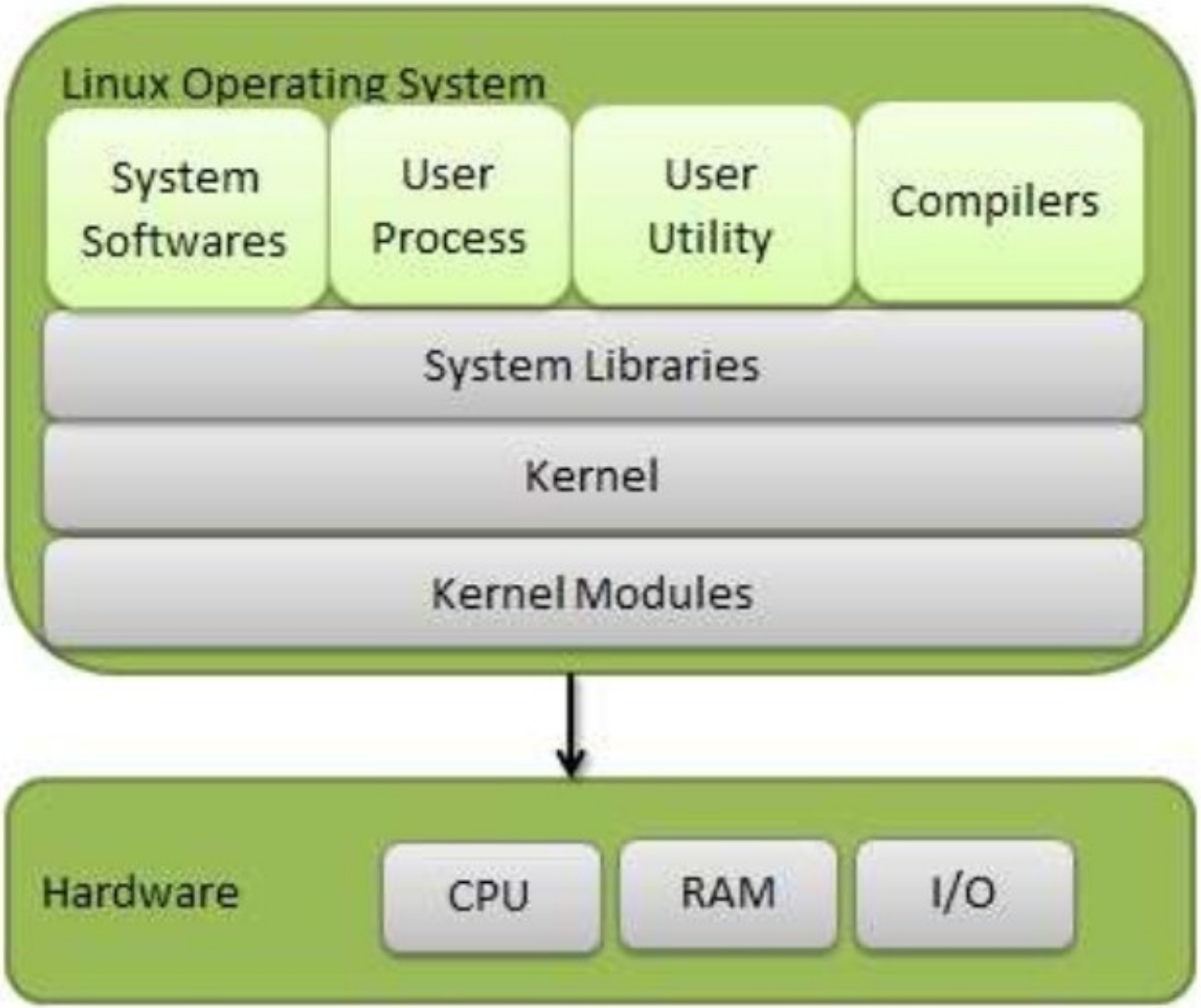
# Virtual Memory



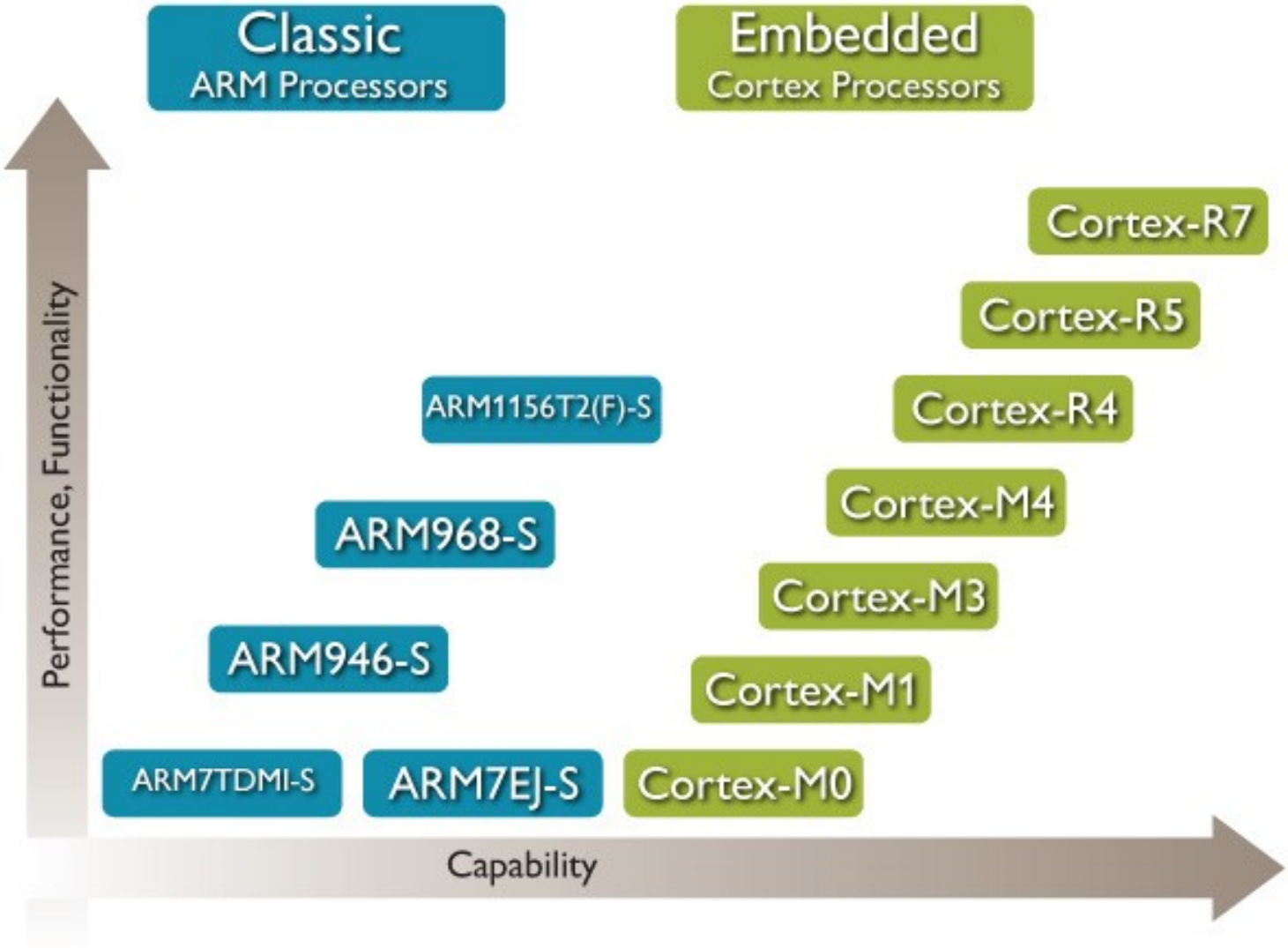
# File System

- A file is a named collection of related information that is recorded on secondary storage such as magnetic disks, magnetic tapes and optical disks. In general, a file is a sequence of bits, bytes, lines or records whose meaning is defined by the files creat or and user

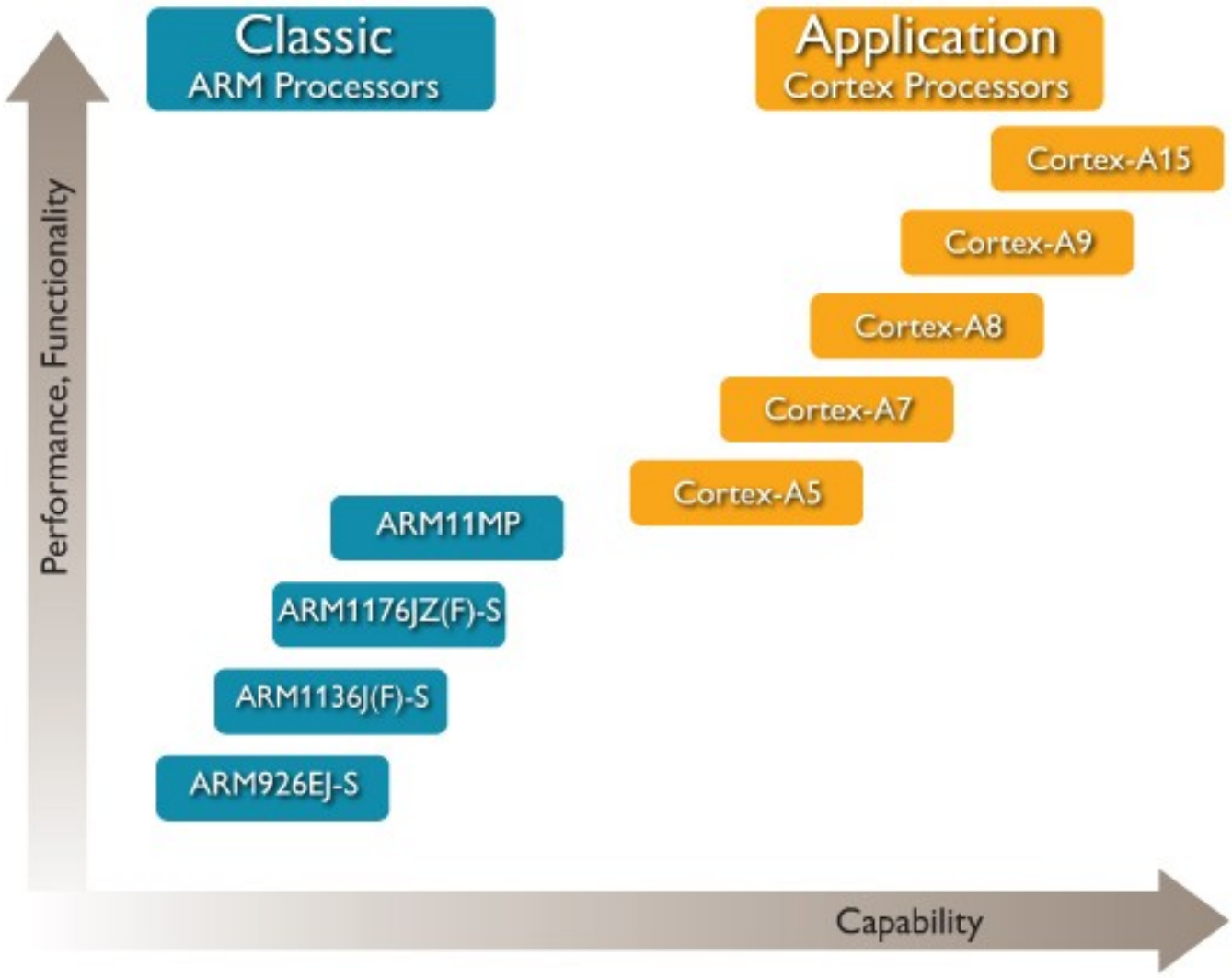
# Components of Linux System



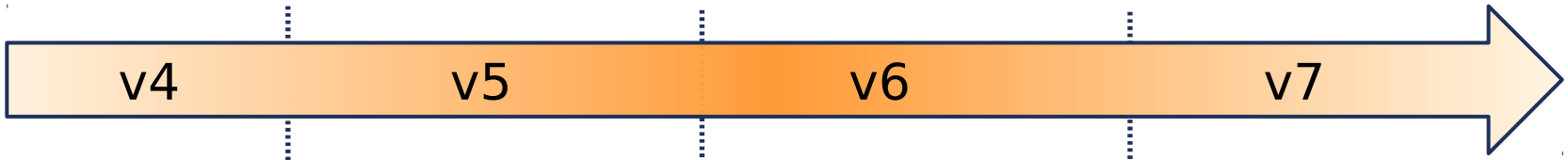
# Embedded Processors



# Application Processors



# Development of the ARM Architecture




**v4**

Halfword and signed halfword / byte support

System mode

Thumb instruction set (v4T)

**v5**

Improved interworking
CLZ
Saturated arithmetic
DSP MAC instructions

Extensions:
- Jazelle (5TEJ)

**v6**

SIMD Instructions
Multi-processing
v6 Memory architecture
Unaligned data support

Extensions:
- Thumb-2 (6T2)
- TrustZone® (6Z)
- Multicore (6K)
- Thumb only (6-M)

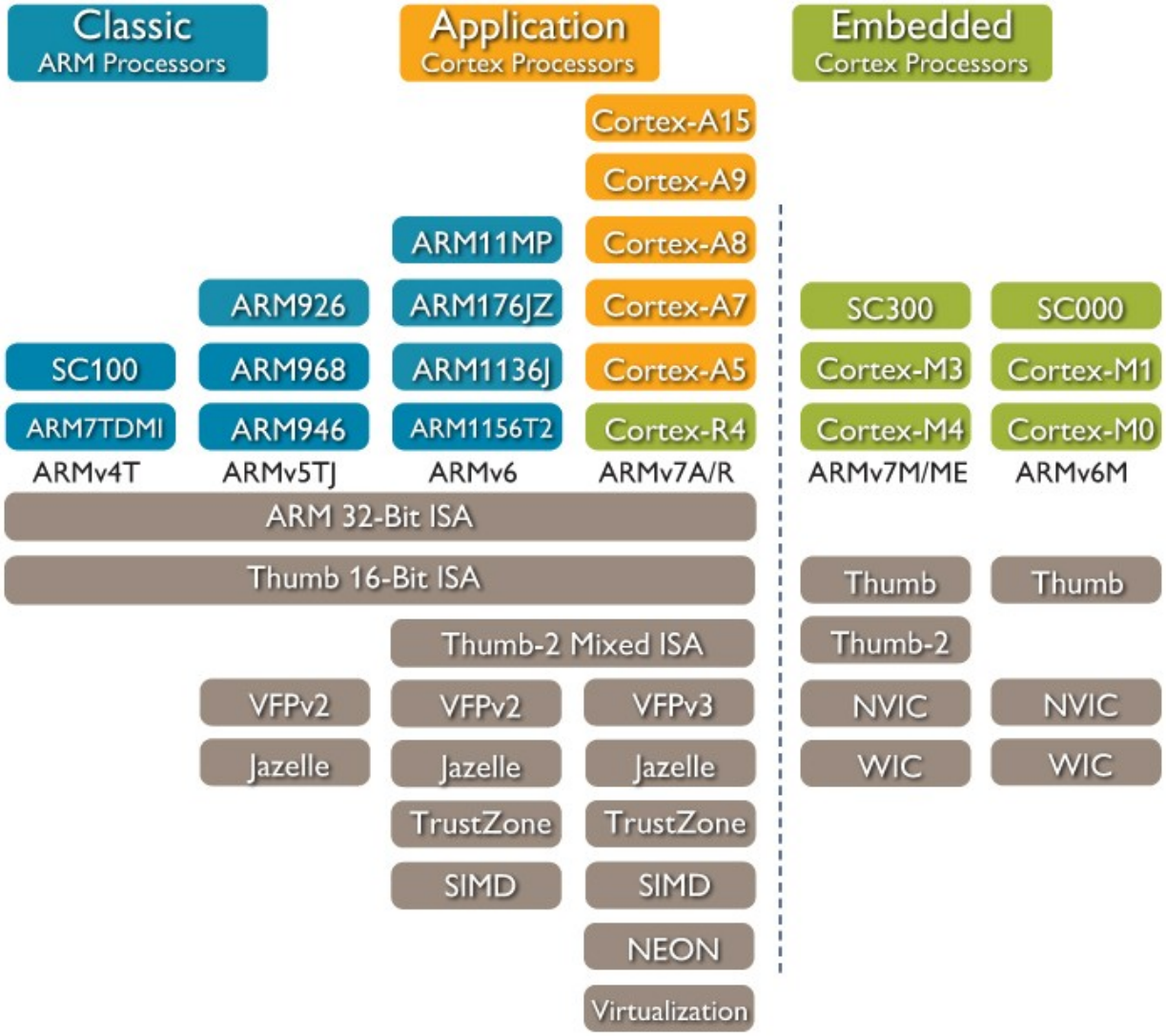
**v7**

Thumb-2

Architecture Profiles
- 7-A - Applications
- 7-R - Real-time
- 7-M - Microcontroller

- **Note that implementations of the same architecture can be different**
  - Cortex-A8 - architecture v7-A, with a 13-stage pipeline
  - Cortex-A9 - architecture v7-A, with an 8-stage pipeline

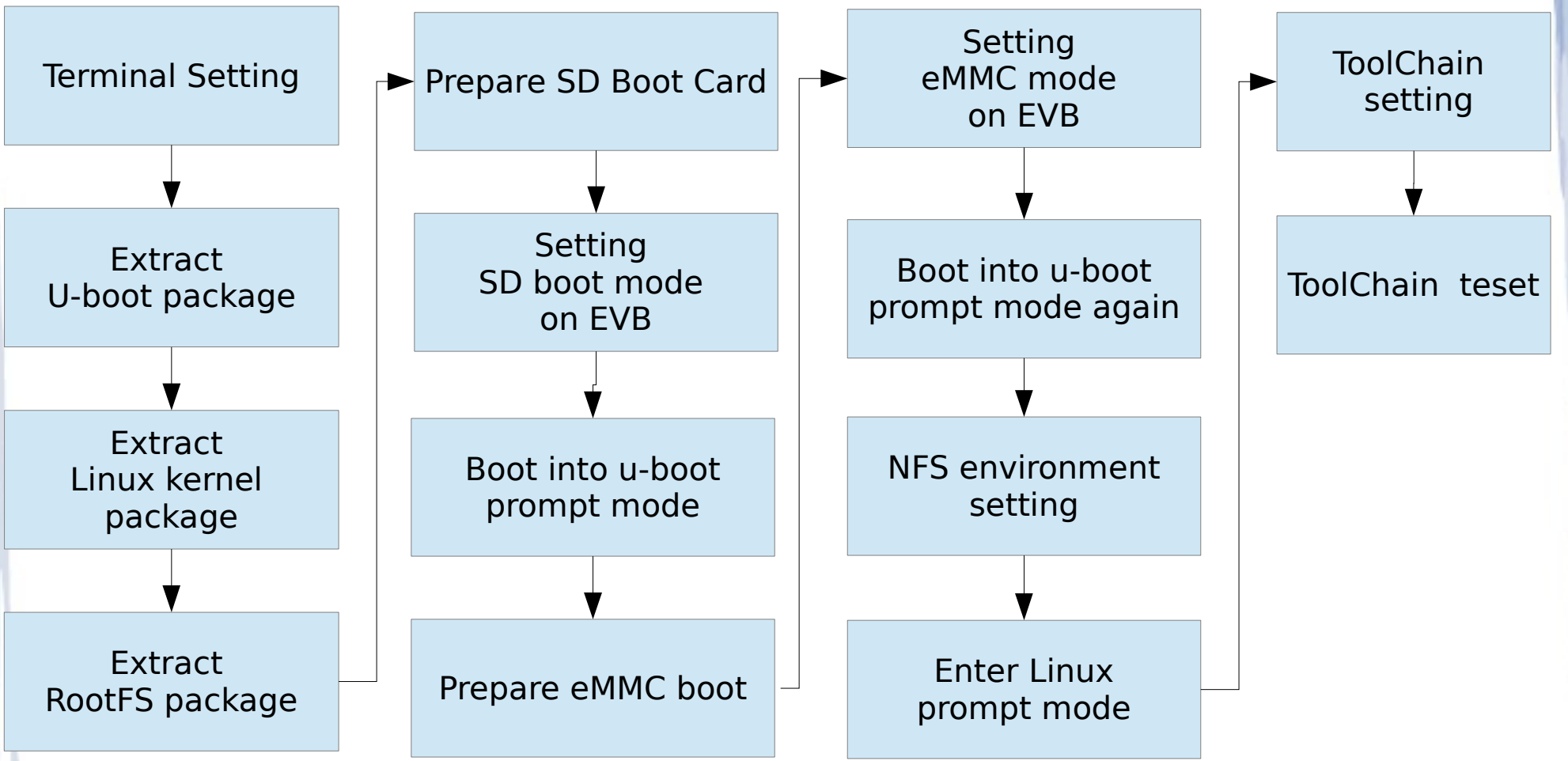
# Which architecture is my processor?



# Just do it!

- Understand tiny-4412 EVB
- Build develop environment
  - Terminal Setting
    - Gtkterm, minicom ..
  - Prepare Tiny4412 BSP
    - U-boot, Linux kernel, RootFS
  - Setting toolchain
  - Build NFS Environment

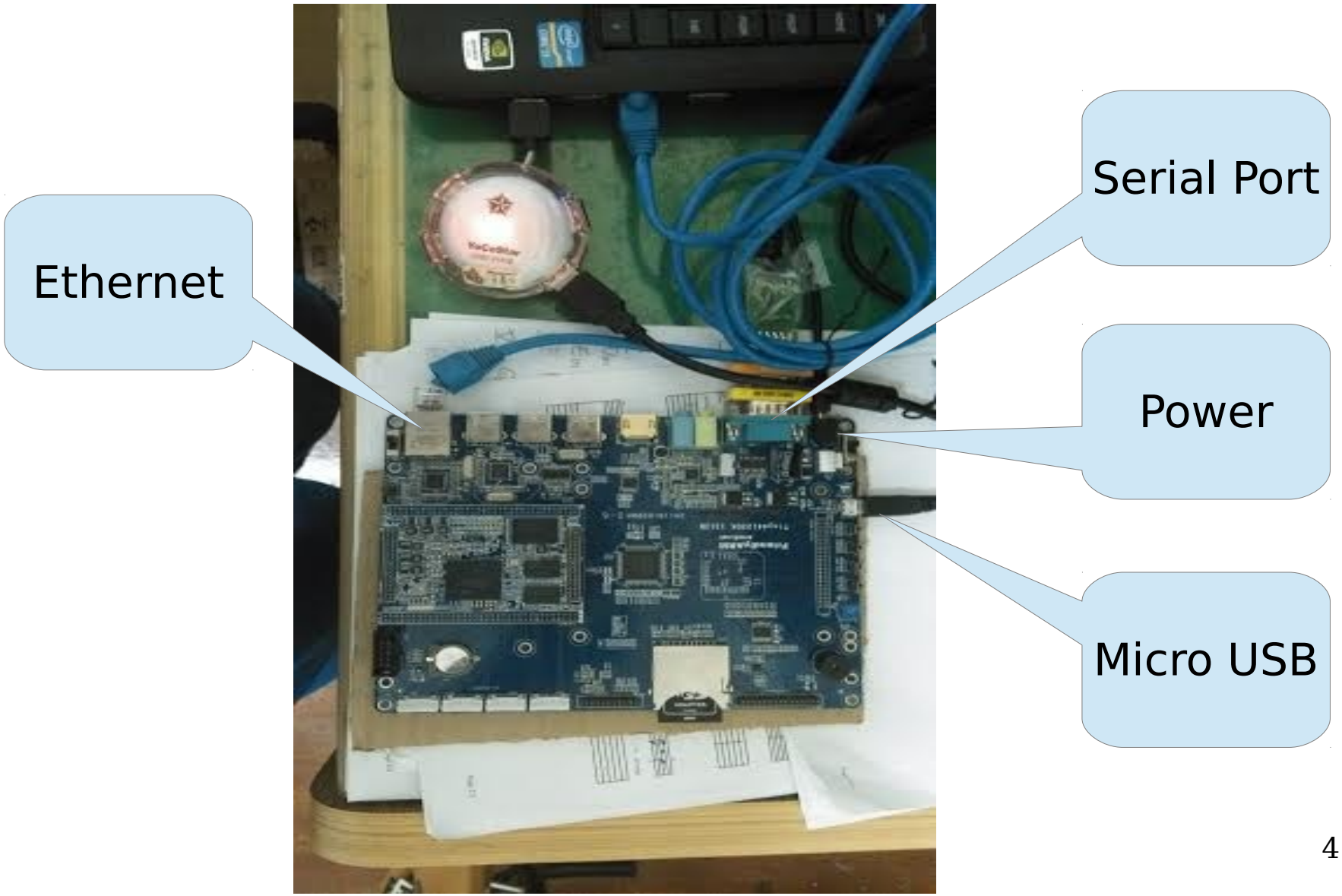
# Exercise Step



# Terminal Setting

- sudo apt-get install gtkterm
- sudo gtkterm
- Connect serial port to Host PC
- Connect network line to Host PC
- Connect micro USB line to Host PC

# EVB Setting



# Tiny-4412 EVB Boot Select

- Boot method
  - SD boot – switch down
  - eMMC boot - switch up