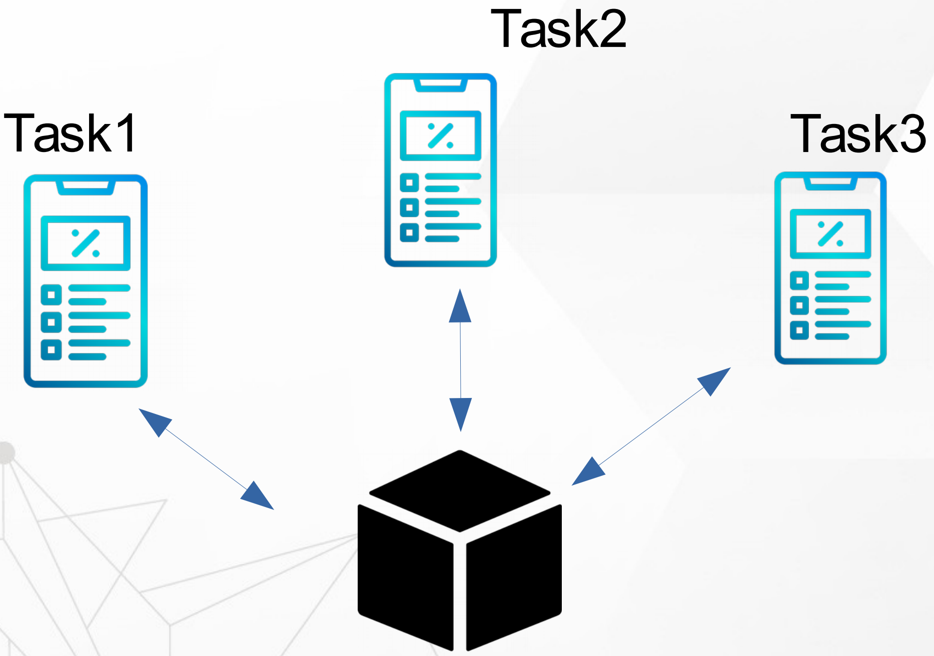
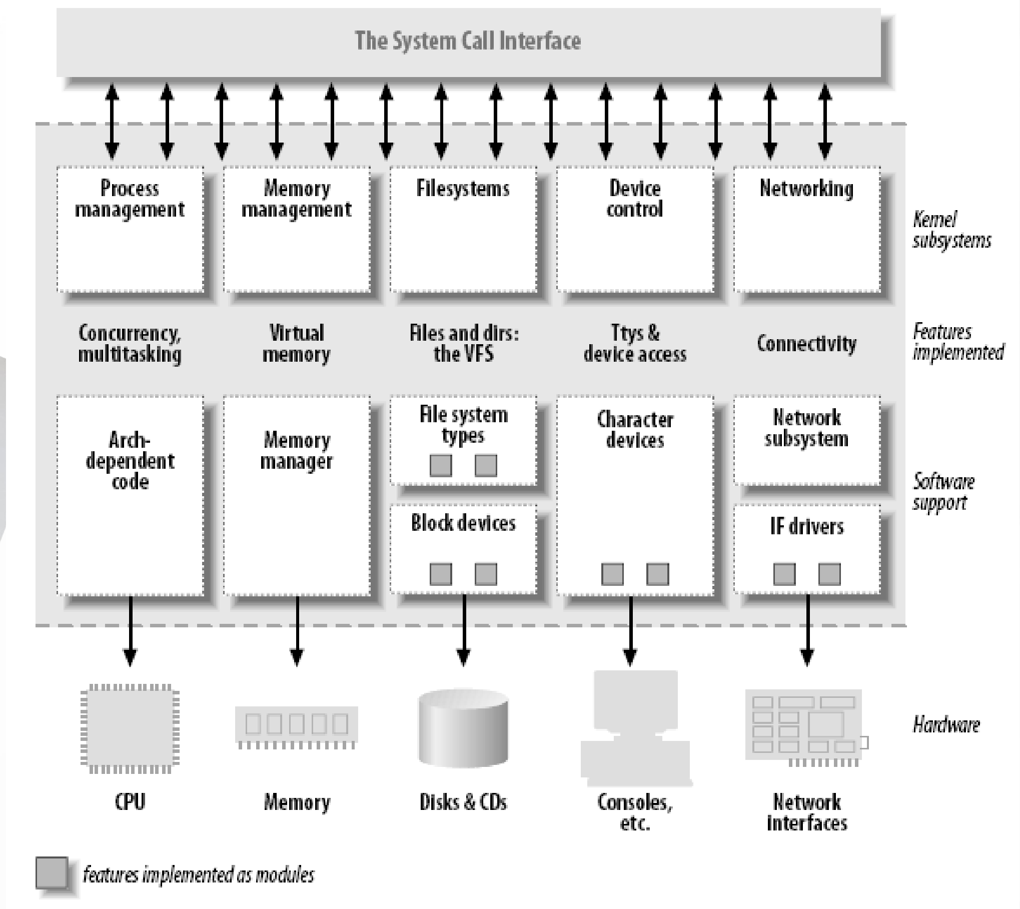
# CH9 Linux Device Driver Module

# Introduction

- Device drivers
  - Black boxes to hide details of hardware devices
  - Use standardized calls



# Kernel Modularization



# Example

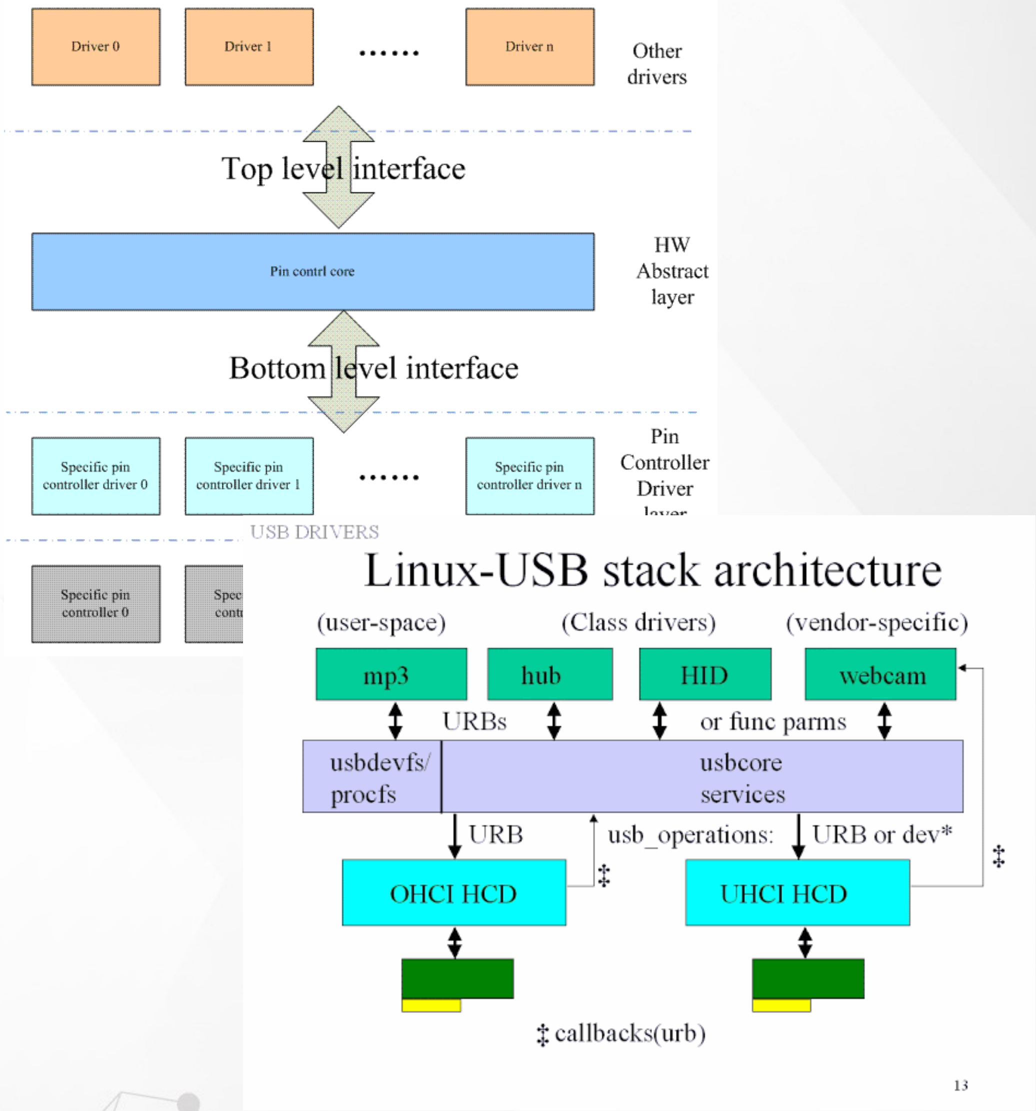
```
$ make
$ sudo insmod simple.ko
$ dmesg | tail
$ lsmod | grep simple
$ sudo rmmod simple
```

# Classes of Devices Driver

- Char module
  - simple
  - access stream of bytes
- Block module
  - block and char devices differ only in the way data is managed internally by the kernel
- Network module
  - Manage network data packets

# Subsystem

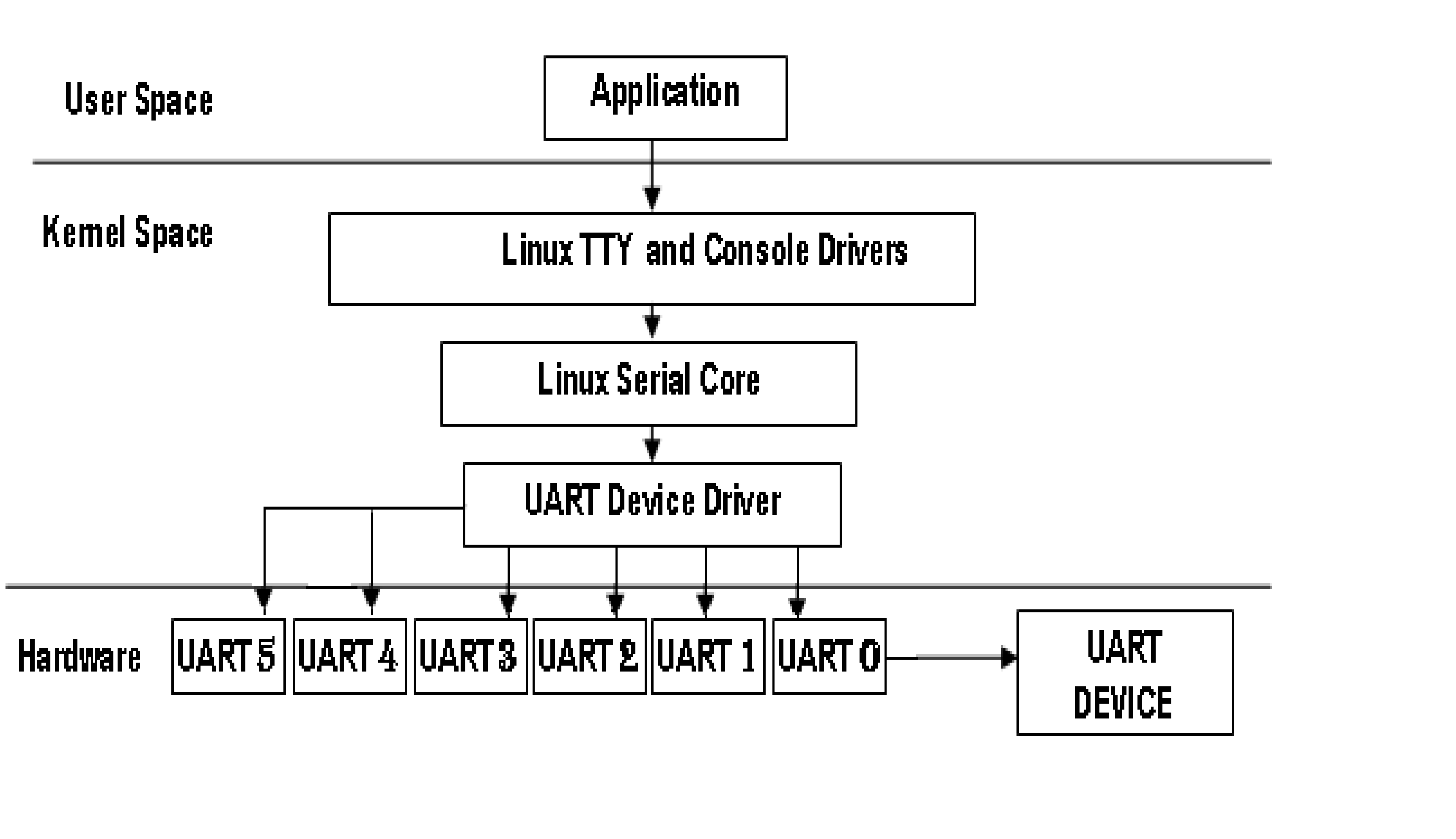
- DRM Subsystem
- GPIO Subsystem
- I2C Subsystem
- SPI Subsystem
- MTD Subsystem



http://abhishekkd.blogspot.com/2013/03/usb-and-mmc.html

https://0uyangsheng.github.io/2018/06/08/Linux-GPIO-Pinctrl-Subsystem/

# Sub-system


User Space
Application
Kemel Space
Linux TTY and Console Drivers
Linux Serial Core
UART Device Driver
Hardware
UART 5
UART 4
UART 3
UART 2
UART 1
UART 0
UART DEVICE

# Where are Modules in Kernel

- ${KERNEL}/drivers

  → ${KERNEL}/drivers/chars

  → ${KERNEL}/drivers/i2c

  → ${KERNEL}/drivers/gpio

  → Module aliases for module loading utilities.

- **Kernel build configure**

  → **${KERNEL}/.config**

- Kconfig

  → ${KERNEL}/drivers/chars/Kconfig

- $make menuconfig

# Build Modules

- Build modules

  → $ make modules

- Add install patch

  → $ export **INSTALL_MOD_PATH**=../modules

- Install module to INSTALL_MOD_PATH

  → $ make modules_install

  → Installs all modules in /lib/modules/<version>

# Module Deploy

- modules_install

→ modules.alias : Module aliases for module loading utilities.

→ modules.dep : Module dependencies

→ modules.symbols : Tells which module a given symbol

# Install Module

- Install module

  → $ modprob ${module_name}

  → $ insmode  ${module_name}

- Remove moudle

  → $ modprob -r ${module_name}

  → $ rmmod

# modprobe depmod

- modprobe

  → /lib/modules/'uname -r'

- Depmod

  → creates a list of module dependencies /lib/modules/version

# Exercise

- Deploy Linux modules to target board form nanopi-m4 Linux kernel
- Use depmod
- Use modprobe, insmod, rmmod