RK3399 and SOC





RockPi WiKi

- Rock Pi Wiki
 - https://wiki.radxa.com/Rockpi4
- Rock Pi Hardware
 - https://wiki.radxa.com/Rockpi4/hardware/rockpi4
- Rock Pi Image
 - https://wiki.radxa.com/Rockpi4/downloads



CPU





CPU (1)

- > Two CPU clusters
 - Big cluster with dual-core Cortex-A72
 - high-performance
 - >> Little cluster with quad-core Cortex-A53
 - low power

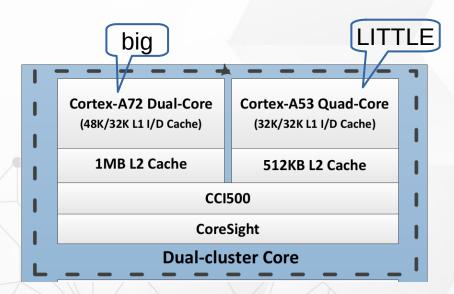
1 '			7			
	Cortex-A72 Dual-Core	Cortex-A53 Quad-Core	ı			
	(48K/32K L1 I/D Cache)	(32K/32K L1 I/D Cache)				
ı	484D 12 Cook o					
1	1MB L2 Cache	512KB L2 Cache				
1	CCI500					
1	CoreSight					
Dual-cluster Core						





CPU (2)

- Arm big.LITTLE technology
 - "LITTLE" processors are designed for maximum power efficiency
 - "big" processors are designed to provide maximum compute performance.





Memory





Memory (1)

- Internal ROM
 - Internal BootRom (Size : 32KB)
 - boot from
 - SPI, eMMC, SD/MMC
- External
 - DDR3/DDR3L/LPDDR3/LPDDR4
 - SPI NOR/NAND Flash
 - **EMMC5.1**
 - SD3.0/MMC4.51





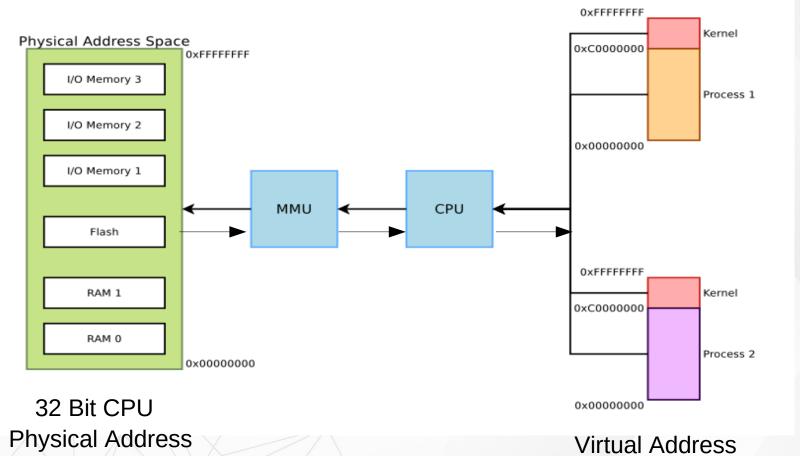
Memory (2)

- Internal ROM
 - Internal BootRom (Size : 32KB)
 - **>>** boot from
 - SPI, eMMC, SD/MMC
- Internal RAM
 - **200KB**





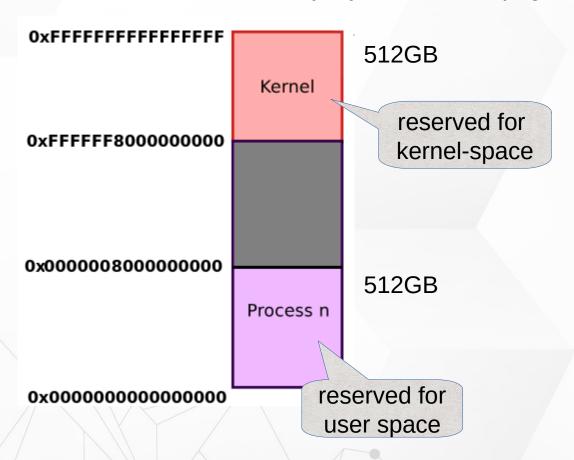
MMU (1)





MMU (2)

64 Bit CPU AArch64 Linux memory layout with 4KB pages + 3 levels







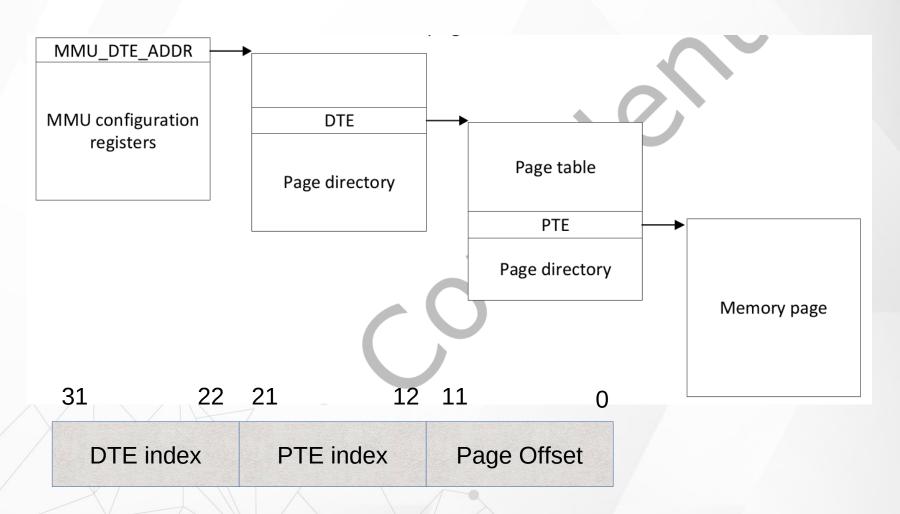
MMU (3)

- The MMU divides memory into **4KB** pages
- 2-level page table structure
 - The First level
 - Page Directory consists of 1024 Directory Table Entries (DTEs)
 - Each pointing to a Page Table.
 - The Second level
 - The Page Table consists of 1024 Page Table Entries (PTEs)
 - Each pointing to a page in memory





MMU (4)





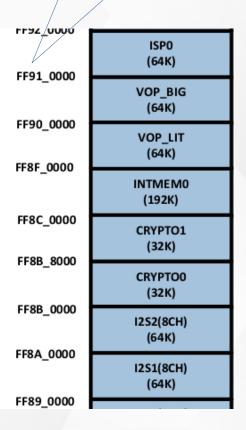


MMU (5)

I/O Address Mapping

FF3F_0000 FF79_8000 12C8 GPIO4 (64K) (32K) FF3E_0000 12C4 (64K) can't use FF3D_0000 physical memory 12C0 (64K) FF3C_0000 address direct INTMEM1 (64K) FF3B_0000 FF77_0000 Reserved CRU (64K) (64K) FF76_0000 FF3A_0000 MAILBOX1 **PMUCRU** (64K) (64K) FF39 0000 FF75_0000 WDT2 Reserved (64K) (64K) FF38_0000 FF74 0000 **UART4** GPIO1 (64K) (64K) FF37_0000 FF73_0000

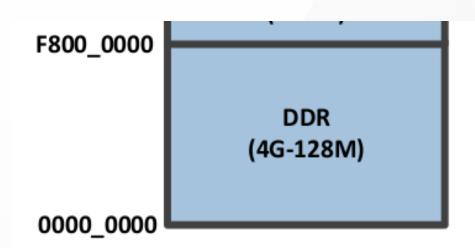
Physical memory address





MMU (6)

DDR SD RAM Memory





Interconnect Connect





Bus Architecture

- MBA: Advanced Microcontroller Bus Architecture
 - **MAXI**
 - **MAHB**
 - **MAPB**





GPU





Graphics Engine (1)

- Graphics Process Unit
- Mali-T860MP4 GPU
 - >> OpenGL ES1.1/2.0/3.0, OpenCL1.2,
 - 3DGraphics Engine
 - 2D Graphics Engine





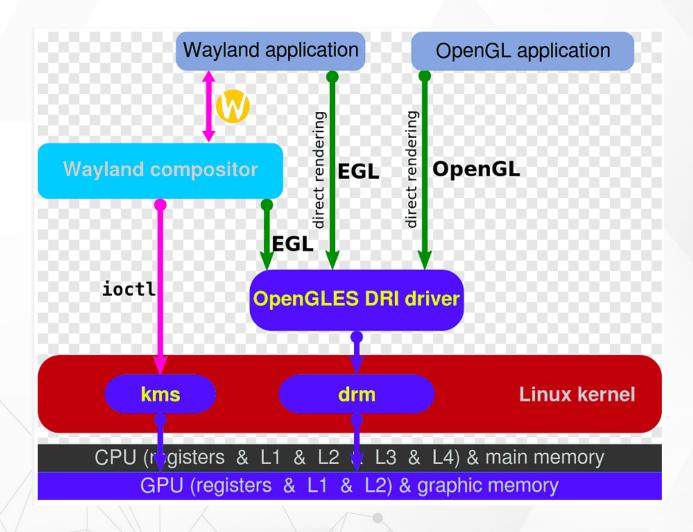
Graphics Engine (2)

- OpenGL
 - Open Graphics Library
- OpenGL ES
 - OpenGL for Embedded Systems
- EGL
 - Native Platform Graphics Interface





Graphics Engine (3)





Connect





USB (1)

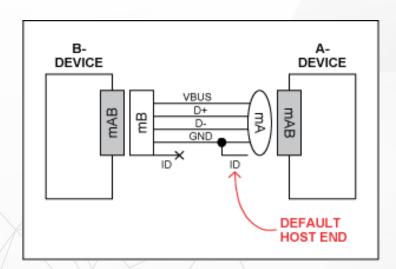
- **USB** Host
 - **>>** RK3399
 - >> OHCI : 1.1. Hardware Complex
 - > UHCI: 1.0, 1.1 Software Complex
 - **>>** EHCI : 2.0
 - >> XHCI : 3.0
- USB Device
 - >> USB Storage





USB (2)

- USB OTG
 - >> USB_ID 信號爲低時,該設備應作爲 Host
 - >> USB_ID 信號爲高時,該設備作爲 Slave

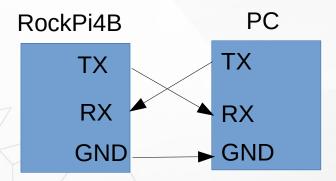






UART

- The Universal Asynchronous Receiver/Transmitter
 - Write Data
 - CPU → Data → APB → UART
 - Nead Data
 - Data → UART → APB → CPU

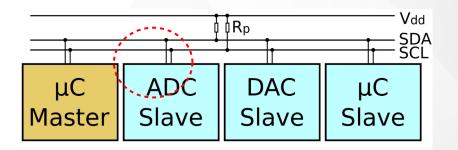




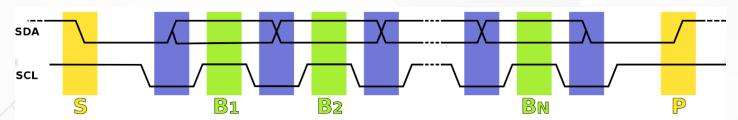


12C (1)

- Serial bus
 - >>SDA data line
 - SCL clock line



Protocol

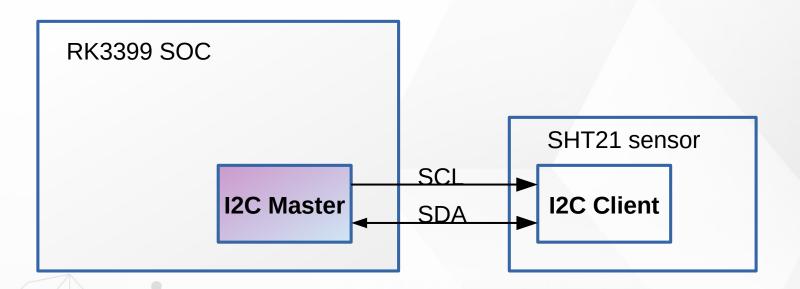






12C (2)

Master and Client

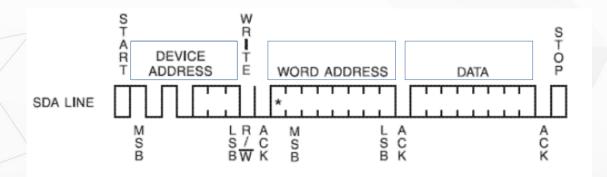






I2C protocol

- Write
 - byte write
 - page write
- Device address
- >> Read/write bit: 0
- ACK

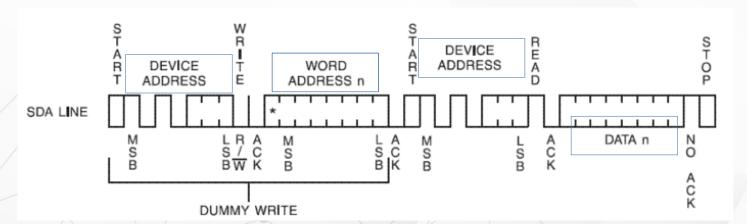






I2C protocol

- Read
 - byte read
 - page read
- Device address
- Read/write bit : 1
- ACK

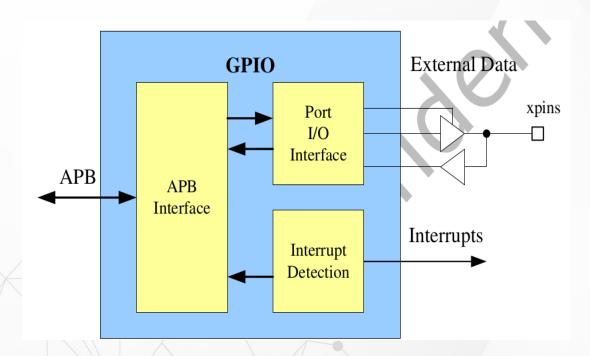






GPIO

- General Purpose Programming I/O
- GPIO controls the output data and direction of external I/O pads





WiFi Basic





WIFI - 802.11(1)

▶ IEEE 802.11 - 無線區域網路

ISO OSI 7-layer model

Application Presentation Session **Transport** Network Data Link

Physical

IEEE 802 standards Logical Link Control Medium Access (MAC) Physical (PHY)





WIFI - 802.11(2)

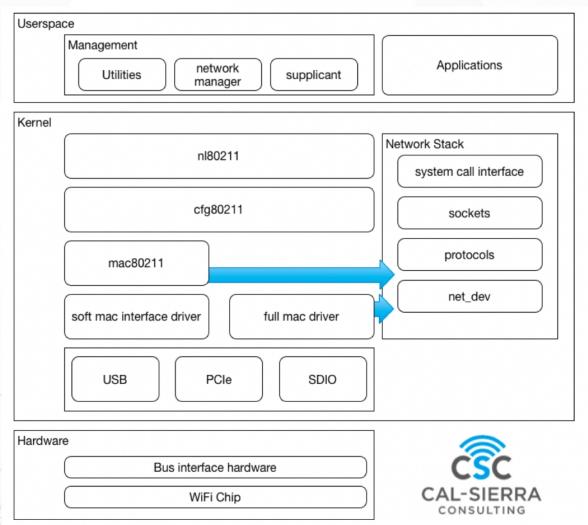
世代名稱 ^[註 1] ◆	IEEE標準		最大速率	頻率	
□10台件(一寸	名稱 ◆	發布年份 ♦	(Mbit/s)	(GHz)	
Wi-Fi 7	802.11be	(2024) ^[註 2]	1376~46120	2.4/5/6	
Wi-Fi 6E	802.11ax	2020	574~9608 ^[1]	6 ^[2]	
Wi-Fi 6			2019	2.4/5	
Wi-Fi 5	802.11ac	2014	433~6933	5 ^[3]	
Wi-Fi 4	802.11n	2008	72~600	2.4/5	
Wi-Fi 3 ^[註 3]	802.11g	2003	6~54	2.4	
Wi-Fi 2 ^[註 3]	802.11a	1999	6~54	5	
Wi-Fi 1 ^[註 3]	802.11b	1999	1~11	2.4	
Wi-Fi 0 ^[註 3]	802.11	1997	1~2	2.4	

- 1. ^ Wi-Fi是Wi-Fi联盟的商標
- 2. ^ 預定的發布年份
- 3. ^ 3.0 3.1 3.2 3.3 Wi-Fi联盟未定義 Wi-Fi 0/1/2/3 的世代名稱^{[4][5]}





WIFI – Linux & 802.11 (1)







WIFI – Linux & 802.11 (2)

- Mac80211
 - most associated to hardware offloading
 - >>> the 802.11 protocol state machine lives here
- cfg80211
 - middle-layer **Handles Everything Configurable**
- nl80211
 - The API between user-land and kernel-land
 - Relies on the **netlink** protocol to exchange messages between the two worlds





WIFI - Tool

- > Show / manipulate wireless devices and their configuration
 - wi 🜊
- >> For connecting to a WPA/WPA2 network
 - wpa_supplicant



Audio





ALSA Overview

- Advanced Linux Sound Architecture
 - >>Linux kernel
 - Software framework
- Sound Servers
 - PulseAudio, JACK ...
- > ALSA stream is a data flow representing sound
 - PCM (Pulse-code modulation)







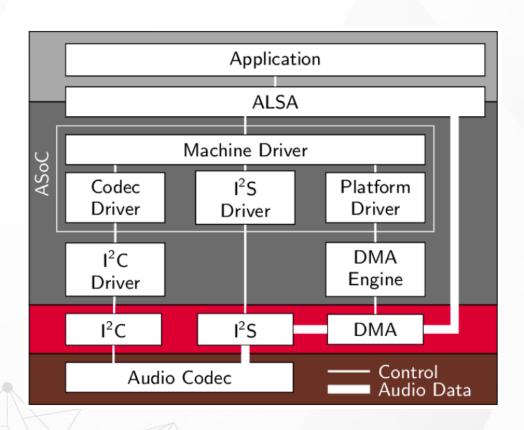
ALSA Overview

- Audio Codec
 - Audio
 Codec 就是音樂訊號(Audio)壓縮 / 解壓縮 (Compress/DECompress)
 的演算法或程式,前後加起來就是 Audio Codec.
- Parameters of the hardware
 - sampling rate : 44100 Hz
 - sample width : 8 bit, 16 bit, 24 bit
 - sample encoding: endianness
 - number of channels : 1 channel, 2 channel ...



ALSA Overview





https://www.researchgate.net/figure/Structure-of-ASoC-and-the-embedment-into-the-Linux-audio-framework_fig2_262112720





Sound Card in Linux (1)

Is /proc/asound/ -I

```
lrwxrwxrwx
               1 root root 5 Apr 30 06:51 HDMICODEC -> card1
               4 root root 0 Apr 30 06:51 card0
dr-xr-xr-x
               3 root root 0 Apr 30 06:51 card1
dr-xr-xr-x
               1 root root 0 Apr 30 06:51 cards
-r--r--r--
               1 root root 0 Apr 30 06:51 devices
-r--r--r--
               1 root root 0 Apr 30 06:51 hwdep
-r--r--r--
               1 root root 0 Apr 30 06:51 pcm
-r--r--r
               1 root root 5 Apr 30 06:51 rockchipes8316c -> card0
Irwxrwxrwx
dr-xr-xr-x 2 root root 0 Apr 30 06:51 seq
               1 root root 0 Apr 30 06:51 timers
-r--r--r--
               1 root root 0 Apr 30 06:51 version
-r--r--r--
```





Sound Card in Linux (2)

cat /proc/asound/cards

Sound card 0

0 [rockchipes8316c]: rockchip_es8316 - rockchip,es8316-codec rockchip,es8316-codec

1 [HDMICODEC]: HDMI-CODEC - HDMI-CODEC HDMI-CODEC

Sound card 1





Sound Card in Linux (3)

rockchipes8316c

Is -I /proc/asound/card0/

-r--r-- 1 root root 0 Apr 30 06:59 id capture
dr-xr-xr-x 3 root root 0 Apr 30 06:59 pcm0c
dr-xr-xr-x 3 root root 0 Apr 30 06:59 pcm0p
playback

ls -l /proc/asound/card1/

-r--r-- 1 root root 0 Apr 30 07:01 id dr-xr-xr-x 3 root root 0 Apr 30 07:01 pcm0p

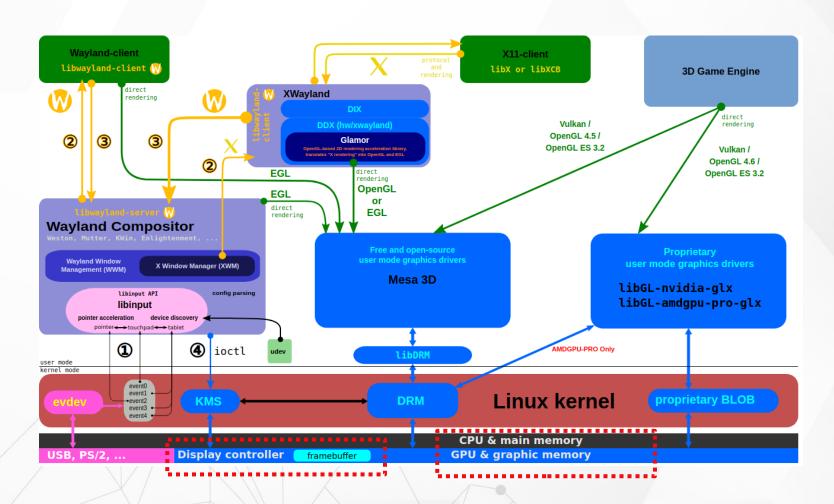


Linux Display Subsystem





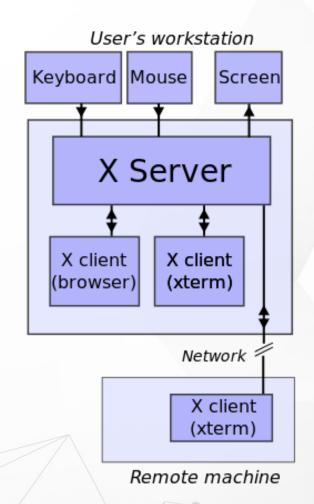
X Server and Wayland Compositor







Linux Windows System (1)

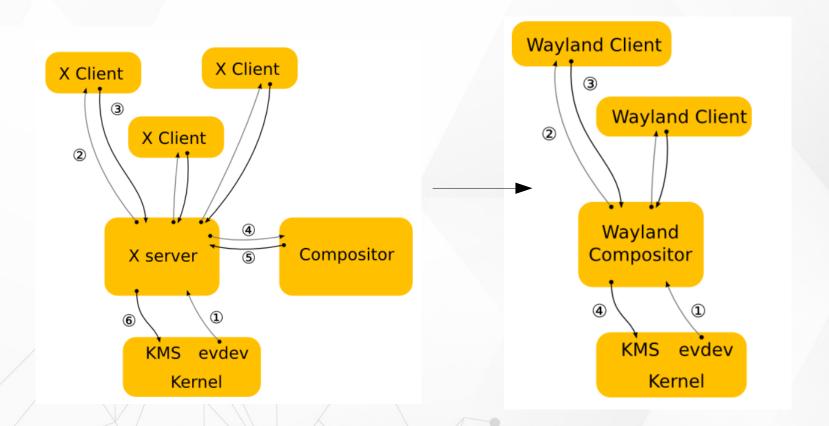






Linux Windows System (2)

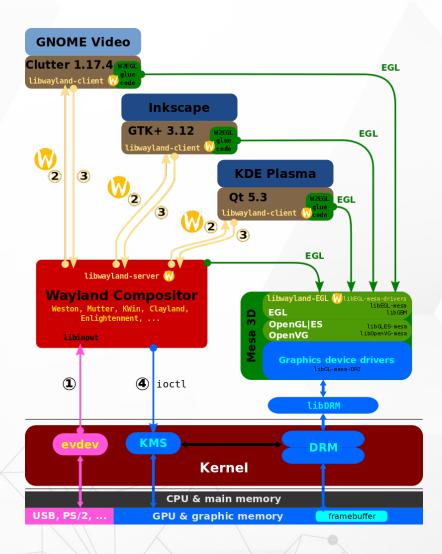
Wayland is a replacement for the X11 window system protocol





Wayland Display Server and EGL











GTK

- GTK (formerly GTK+) is a free and open-source cross-platform widget toolkit for creating graphical user interfaces (GUIs).

GNOME

- GNOME is the default desktop environment of many major Linux distributions
- originally an acronym for GNU Network Object Model Environment
- free and open-source desktop environment for Linux and other Unix-like[10] operating systems

https://en.wikipedia.org/wiki/GNOME https://en.wikipedia.org/wiki/GTK



OpenGL and EGL



EGL

- EGL Native Platform Graphics Interface is an interface portable layer for graphics resource management.
- works between rendering APIs such as OpenGL ES or OpenVG and the underlying native platform window system

OpenGL

 OpenGL (Open Graphics Library) is a crosslanguage, cross-platform application programming interface (API) for rendering 2D and 3D vector graphics

Mesa

- Mesa, also called Mesa3D and The Mesa 3D
 Graphics Library
- it is an open source implementation of OpenGL, Vulkan, and other graphics API specifications

https://www.khronos.org/egl/





Direct Rendering Manager (DRM)

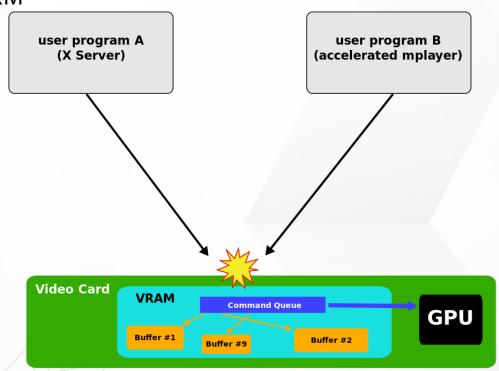
- Direct Rendering Manager
 - Management of buffers and free space within that memory.
 - Solve Frame buffer driver cannot be used GPU and multi-user process.
- DRM consists of
 - libdrm
 - libdrm provides a user space library for accessing the DRM
 - KMS : Kernel Mode Setting
 - Change resolution and depth
 - DRI : Direct Rendering Infrastructure
 - Interfaces to access hardware directly
 - GEM: Graphics Execution Manager
 - Buffer management
 - DRM Driver in kernel side
 - Access hardware







If no use DRM

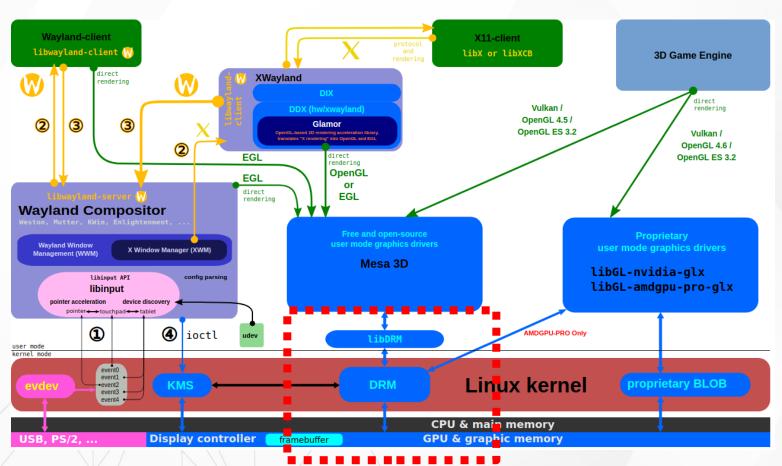


© 2014 Javier Cantero - this work is under the Creative Commons Attribution ShareAlike 4.0 license



Direct Rendering Manager (DRM)



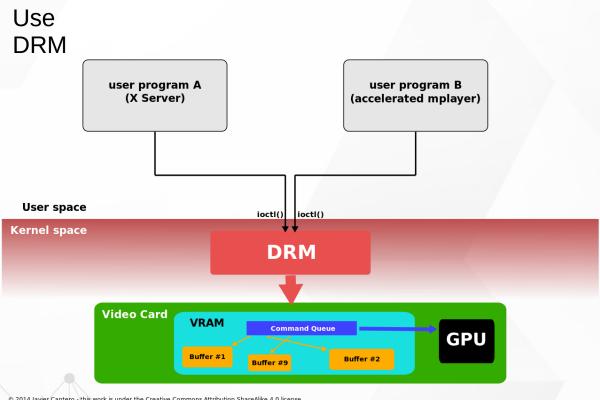


https://upload.wikimedia.org/wikipedia/commons/2/2d/ The_Linux_Graphics_Stack_and_glamor.svg



Direct Rendering Manager (DRM)





© 2014 Javier Cantero - this work is under the Creative Commons Attribution ShareAlike 4.0 license



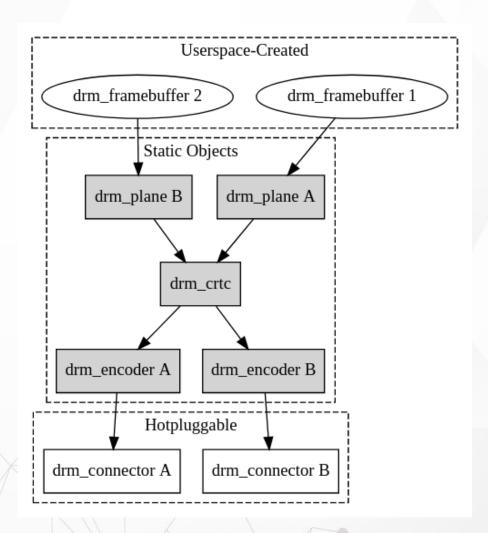




- > KMS device model
 - >CRTCs
 - >> Connectors
 - >> Encoders
 - Planes
- Kernel Mode Setting
 - screen resolution
 - color depth and
 - refresh rate

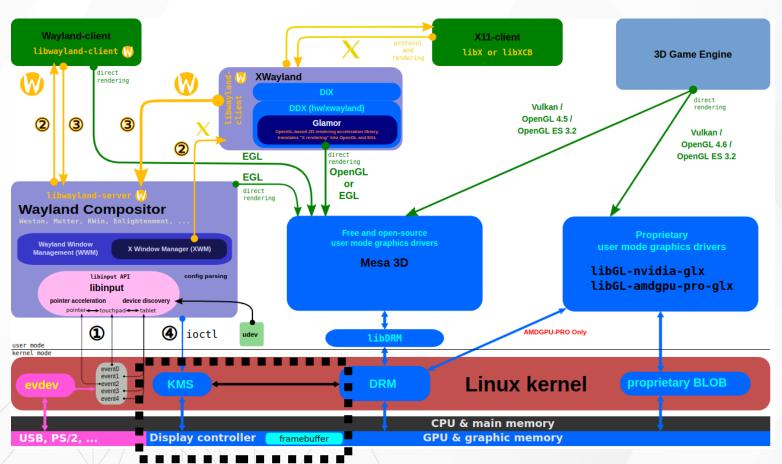








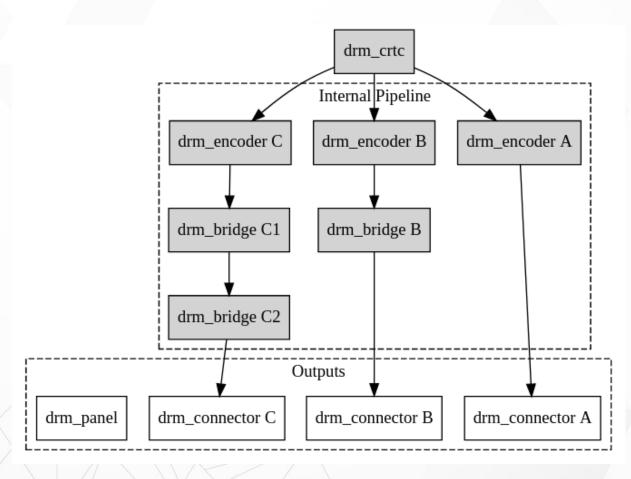




https://upload.wikimedia.org/wikipedia/commons/2/2d/ The Linux Graphics Stack and glamor.svg











Video Frame Buffer

The frame buffer device provides an abstraction for the graphics hardware.

