

DEVICE DRIVERS AND INTERRUPTS SERVICE MECHANISM

Lesson-11: Device types, Physical and Virtual device functions

Device Types

- For each type of device, there is a set of the generic commands.
- For example, for char device one set of commands and for block device there can be another set.

Device Types

- Types of Physical and Virtual devices in a system
- char, block, loop back device, file, pipe, socket, RAM disk, sound, video, media and net.

Virtual device driver

- Definition : A virtual-device driver is the component of a device driver that communicates directly between an application and memory or a physical device.
- Virtual device driver controls the flow of data
- Allows more than one application to access the same memory or physical device without conflict.

Char Device

- Char Device: For example, a device to which one character is sent at one time or is read from it at one time. For example, mouse, keyboard, keypad, timer .

Block Device

- **Block Device:** For example, a device to which one block of characters is sent at one time or is read from it at one time. For example, printer, disk.

Block Device configuration as Char Device

- Block as well as Char device: For example, a device to which one block of characters or a single character is sent at one time or is read from it at one time. For, example, LCD display unit. A device can be configured as char or block as per the need by a generic command.

Configuration as loop-back Device

- Loop-back Device: A device to which one character or set of characters are sent, and those are echoed back to same.

Configuration as copy Device

- **Copy Device:** A device using which a set of characters are sent, and those are returned to another device. For example, disk_copy device when characters are copied from one disk to another or a keyboard-cum-display device. Keyboard input is sent to a buffer and display unit uses that buffer for display.

Virtual Devices

- Besides the physical devices of a system, drivers are also used in a systems for virtual devices.
- Physical device drivers and virtual device drivers have analogies.
- Like physical device, virtual device drivers may also have functions for device connect or open, read, write and close.

Driver

- A memory block can have data buffers for input and output in analogy to buffers at an IO device and can be accessed from a *char* driver or *block* or *pipe* or *socket* driver.

Virtual Device Examples

- Pipe device: A device from to which the blocks of characters are send from one end and accessed from another ends in FIFO mode (first-in first-out) after a connect function is executed to connect two ends.

Virtual Device Examples ...

- Socket device: A device from to which (a) the blocks of characters are send from one end with a set of the port (application) and sender addresses, (b) accessed from another end port (application) and receiver addresses, (c) access is in FIFO mode (first-in first-out) only after a connect function is executed to connect two sockets.

Virtual Device Examples...

- File device: A device from which the blocks of characters are accessed similar to a disk in a tree like format (folder, subfolder,...). For example, a named file at the memory stick.

Virtual Device Examples

- RAM disk Device: A set of RAM memory blocks used like a disk, which is accessed by defining addresses of directory, subdirectory, second level subdirectory, folder and subfolder

Difference between various types of virtual devices

- Pipe needs one address at an end,
- Socket one addresses and one port number at an end, and
- File and disk can have multiple addresses. Reading and writing into a file is from or to current cursor address in the currently open folder.

- Just as a file is sent *read* call, a device must be sent a driver command when its input buffer(s) is to be read.
- Just as a file is sent *write* call, a device needs to be sent a driver command when its output buffer is to be written.

Virtual device example for Remote System access

- A **virtual device** example is a device description that is used to form a connection between a user and a physical system networked or connected to a remote system.

Virtual device driver File name (VxD)

- Driver filename in Windows OS is used where the V stands for virtual and D stands for device. The “d” can be replaced with other characters; for example, VdD means a display driver.

Linux Internals and Device Drivers and Linux Network Functions

- Linux has internal functions called *Internals*. Internals exist for the device-drivers and network-management functions.
- Useful *Linux drivers* for the embedded system and gives the uses of each.

Linux OS Internals

- Linux operating system internals
- Large number of readily available device drivers in Linux
- Linux drivers for most common physical and virtual devices
- Linux functions for the network sockets and protocols

Linux drivers

- Char (For driving a character)
- Block (For driving a block of char)
- Input (For standard IO devices)
- Media (For standard media device functions)
- Video (For standard video device functions)
- Sound (For standard audio device functions)

Linux drivers in the *net* directory

The Linux internal functions exist for

- Sockets,
- Handling of Socket buffers,
- firewalls,
- network Protocols (for examples, NFS, IP, IPv6 and Ethernet) and
- bridges.
- They work separately as drivers and also form a part of the network management function of the operating system.

Summary

We learnt

- Physical and virtual devices
- Drivers for virtual devices are also written similar to the physical device drivers
- Use the same generic commands.

We learnt

- Device Types : char, block, loop back, file device, pipe, socket, RAM disk, sound, video and media
- Linux operating system internals
- Large number of readily available device drivers in Linux.

End of Lesson 14 of Chapter 4