

# LINUX NETWORK DEVICE DRIVER PROGRAMMING

## PREREQUISITE : C and Basic Device Driver

### CH1: PCI DEVICE DRIVER AND ITS ROLE

- Understanding the x86 processor bus: PCI
- PCI Core & Programming the PCI
- Finding & Interacting with a PCI Device
- PCI Bus, Device and Function numbers
- Registering & Finding a PCI device
- Mapping & Accessing the PCI device regions
- Accessing the Configuration Space
- Accessing the I/O and Memory Regions
- Enabling the PCI Device

#### Hands-On Assignments

Lab1: Registering the driver with the PCI subsystem.

Lab2: Write a module that scans your PCI devices, and gathers information about them. For each found device, read some information from its configuration register. Fields you may wish to obtain could include:  
 PCI\_VENDOR\_ID, PCI\_DEVICE\_ID, PCI\_REVISION\_ID,

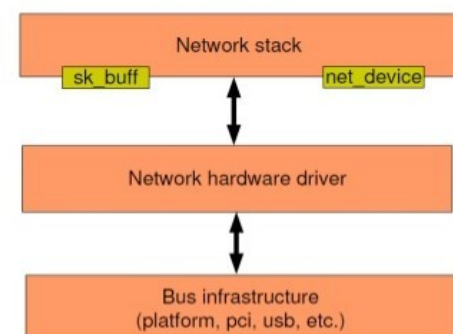
Lab3: Write a Character based PCI driver to find Information about IRQ Line, Memory region, I/O region, configuration region, prefetchable and non-prefetchable region in BAR.

31		16 15		0			
Device ID		Vendor ID				00h	
Status		Command				04h	
Class Code			Revision ID			08h	
BIST	Header Type	Latency Timer	Cache Line Size			0Ch	
Base Address Registers (BAR) 0						10h	
Base Address Registers (BAR) 1						14h	
Secondary Latency Timer	Subordinate Bus Number	Secondary Bus Number	Primary Bus Number			18h	
Secondary Status		I/O Limit	I/O Base			1Ch	
Memory Limit		Memory Base				20h	
Prefetchable Memory Limit		Prefetchable Memory Base				24h	
Prefetchable Base Upper 32 Bits						28h	
Prefetchable Base Limit 32 Bits						2Ch	
I/O Limit Upper 16 Bits			I/O Base Upper 16 Bits			30h	
Reserved					Capabilities Pointer		34h
Expansion ROM Base Address Register (XROMBAR)						38h	
Bridge Control		Interrupt Pin	Interrupt Line			3Ch	

### CH2: BASIC NETWORK DEVICE DRIVER

- Registering the Network Driver
- Buffer Management with skbuffs
- Packet Transmission & Reception
- Reception using interrupt and poll
- Start the network interface's transmit Queue
- Other network operations including statistics

#### Network Device Model



#### Hands-On Assignments

Lab1: Building a Transmitting Network Driver, Module to include a transmission function.

Lab2: Adding Reception, Extend your transmitting device driver to include a reception function.

# LINUX NETWORK DEVICE DRIVER PROGRAMMING

## PREREQUISITE : C and Basic Device Driver

### CH3: ADVANVED NETWORK DEVICE DRIVER

- Registering with the Linux low level bus interface subsystem
- Allocating interface descriptor block (net\_device)
- Device specific structure and initializing media specific fields
- Getting device specific structure object pointer
- Enabling Network interface card
- Getting the Device resources (MMIO and PMIO)
- Getting device MAC address
- Initialization of device methods in the net\_device
- Registering net\_device object with the kernel
- Registering the interrupt handler (ISR)
- Allocating Rxring and Txring
- Initializing the hardware (Network Interface Card)

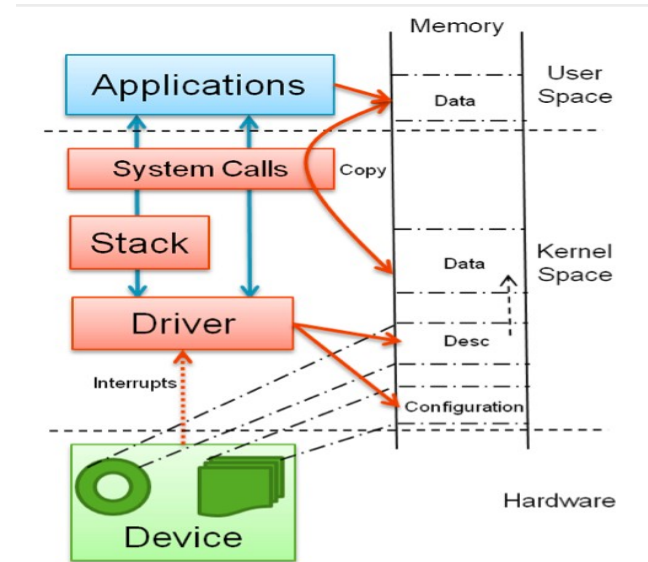


Figure 1: Kernel space network driver

### Hands-On Assignments

Lab3: Writing the PCI based Network Driver for NIC(Network Interface Card) .

Programming the Network Device Registers, Implementing the Transmission and Reception with the actual device(NIC) and Setting up the network across Computers.

**Total duration = One Weekend SAT - SUN(9.30 AM to 5.30 AM)**

**Embisylabs @ Bangalore**  
**info@embisylabs.com**  
**+91-88848 67053**