

# *Internetwork Communication via Router*

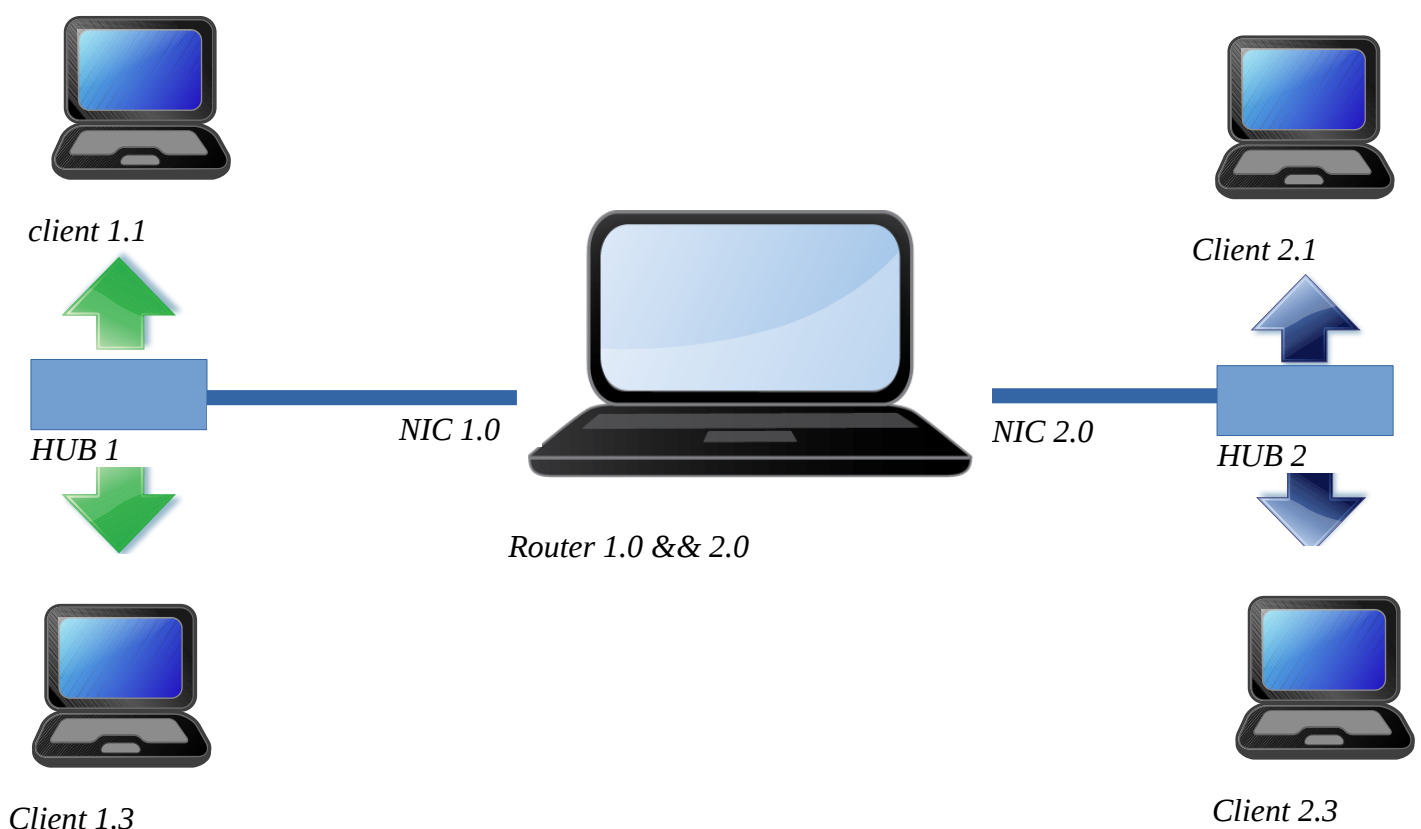
## **Objective:**

When any two or more users (belongs to different networks) want to communicate with each other, then their data/packets only can be delivered to the particular destination via Router. If all those users belong to the same network then they can directly send packets/frames via ethernet or wifi (Though in case of wifi they have to send the frames to the actual mac address of their wifi NIC / wlan interface -> that can be found in linux by “ifconfig”). But for different network they have to send those frames to the router and it is the responsibility of router to route the packet to the particular destination mac address . In this project I have implemented internetwork communication as well as within network communication.

## **Environment Set-up:**

For this assignment first I have to setup the total environment properly ,so that any device can communicate with other (different network or same network) devices. So the **required equipments are:**

1. One laptop having two NIC ( both ethernet interface) -> In my case I have used an “**USB to LAN Clip**” converter for making two interface. So those interfaces are **PCI ethernet and USB ethernet**.
2. **Four laptops** -> (I have two networks and each network will have two devices).
3. **Two HUBs** -> Each network has a hub and all devices within that network communicates through the hub and one interface of the router is connected with the hub so that all those devices can send their data to the router if they want to communicate with different networks.
4. **Lan Cables** -> Three pairs of lan cable is used to make these above mentioned interface with the hub.



## Software Setup:

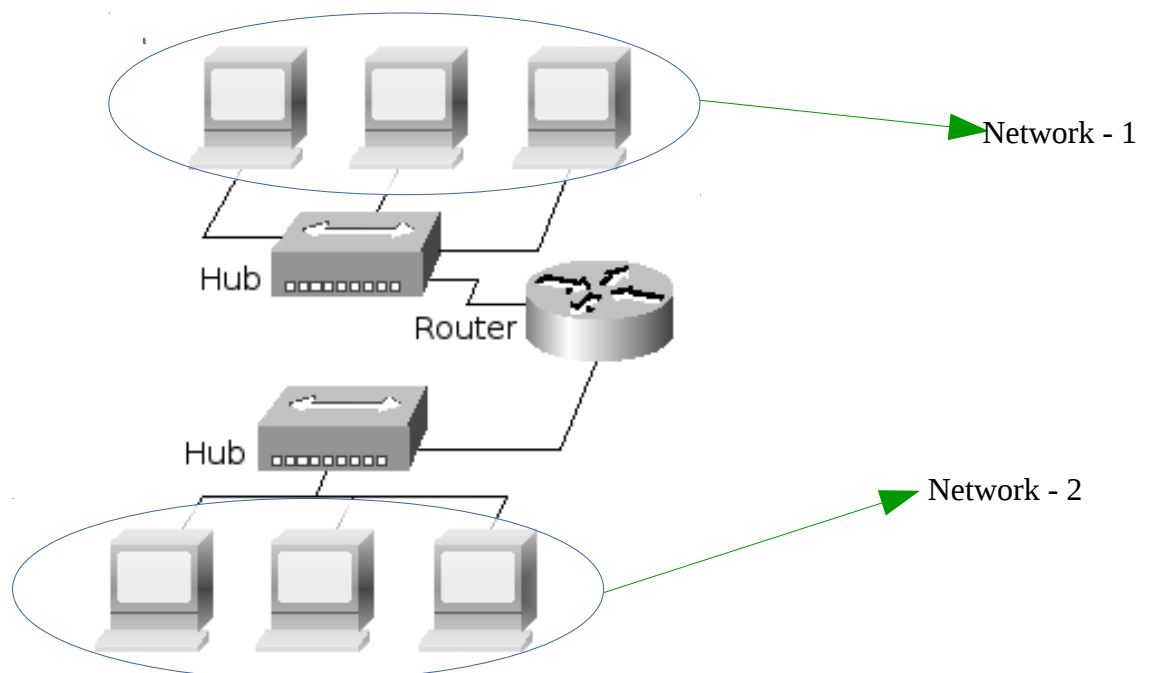
In the software part I have used the followings (**Host machine- LINUX 16.04 LTS**):

1. Oracle Virtual Box (Virtualization platform [Download Link](#))
2. MS-DOS (Operating System)
3. Turbo C++ compiler (For compiling C programmes in DOS)
4. Packet driver ( In my case I have used Pcnnet Fast-III as network adapter and its corresponding packet driver is “pcntpk.com” [Download Link](#) )
5. MTCP software package ( It includes FTP,DHCP, all those network application. Among them FTP can be used to transfer files between host machine and Vms).

## Frame Structure:

Dest mac (6 byte)	Source Mac (6 byte)	Type (2 byte)	Dest ip (2 byte)	Source ip (2 byte)	Data
-------------------	---------------------	---------------	------------------	--------------------	------

**Router:** A **router** is a networking device that forwards data packets between computer networks. Routers perform the traffic directing functions on the Internet. A data packet is typically forwarded from one router to another router through the networks that constitute the internetwork until it reaches its destination node.



## Routing procedure:

Router maintains a table and that table contains all those devices' ip and their corresponding mac address. I haven't implemented ARP protocol so the table will be populated manually. I have created a file and in that file I have stored all those mac address and their corresponding ip and when the router will be run , it will read from the file and populate the table. If router will receive a packet then it will perform following tasks according to the scenario :

**Step 1:** It will compare 1<sup>st</sup> 6 byte with its two NICs' mac address , if one of the mac address will match with the 1<sup>st</sup> 6 byte of the coming frame then it will store the frame into a buffer and **perform tasks according to the following cases** .If 1<sup>st</sup> 6 byte does not match then it will not do any thing and reject the frame.

**Case 2: After Step 1** it will examine the 14<sup>th</sup> and 16<sup>th</sup> byte .If they are same then it will check for 15<sup>th</sup> byte if it belongs to router's host id then the packet will be accepted by router and router will show proper messege but no routing will be done. If 15<sup>th</sup> byte is not router's host id then the packet will be rejected and router will show error message.

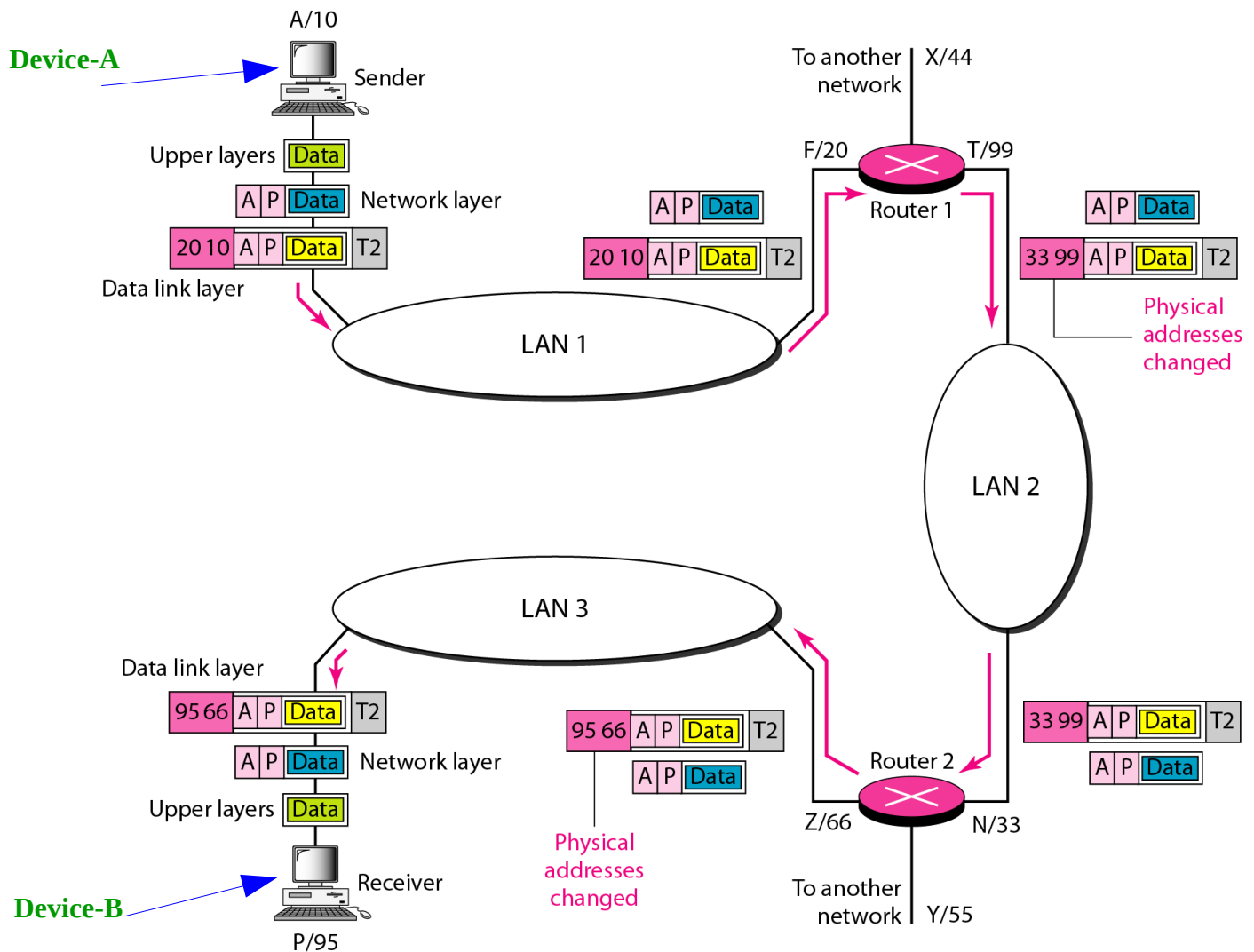
**Case 3:** If 14<sup>th</sup> and 16<sup>th</sup> byte are different then router will search the sub table which only contains all those devises' mac address whose netID = 14<sup>th</sup> byte of the packet. Then from that subtable router will take that unique mac address whose client's hostID = 15<sup>th</sup> byte of the packet. After getting that mac address router will reframe the coming packet by changing the destination mac address with the retrieved mac address from the sub table and source mac address with the mac address of the NIC by which the packet will be send to the network (net ID = 14<sup>th</sup> byte) . After reframing the packet router will send that frame with corresponding software interrupt (In my case either 0x60 or 0x70).

**Case 4 :** If 14<sup>th</sup> and 16<sup>th</sup> byte are different then router will search the sub table which only contains all those devises' mac address whose netID = 14<sup>th</sup> byte of the packet. After getting the sub table if router is unable to find any entry corresponding to the destination host id ,then router will show proper error message and cancel the routing.

### Table that the Router Maintains :

NET-ID	HOST-ID	MAC ADDRESS					
1	1 (PCI NIC)	08	00	27	93	D6	34
	2	08	00	27	D6	F4	4A
	3	08	00	27	E6	25	Eb
2	1 (USB NIC)	08	00	27	E6	2E	EF
	2	08	00	27	00	2B	B3
	3	08	00	27	A3	Bb	93

## How packets are being routed in a small network :



Here device-A wants to send packet to device-B . Schematic diagram of the entire routing process in which packet reframing has taken place. While packets are being routed generally the destination and Source ip remain unchanged but the destination mac and source mac will be changed accordingly.

### Example :

Device-A has been assigned an iip -> 1.2 and MAC -> 08:00:27:D6:F4:4A

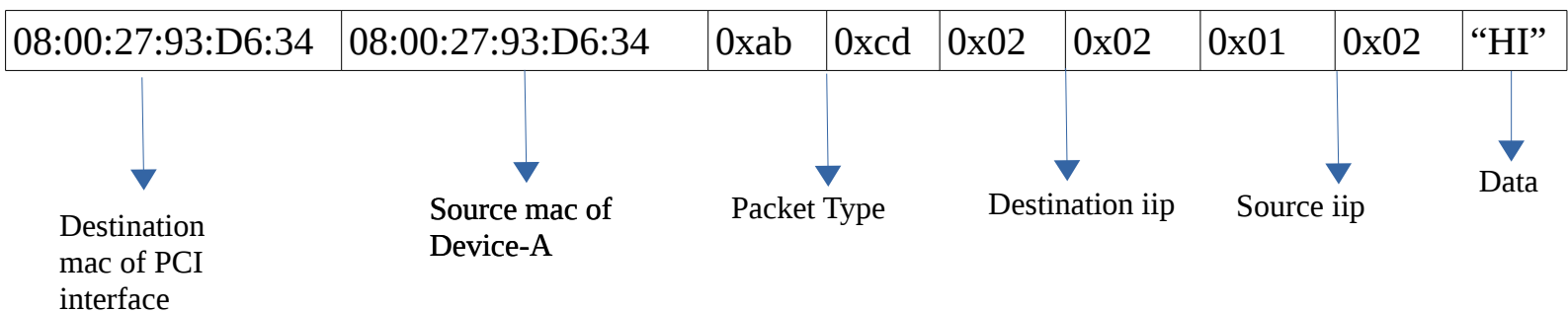
Device-B has been assigned an iip -> 2.2 and MAC -> 08:00:27:00:2B:B3

### **Router has two interfaces:**

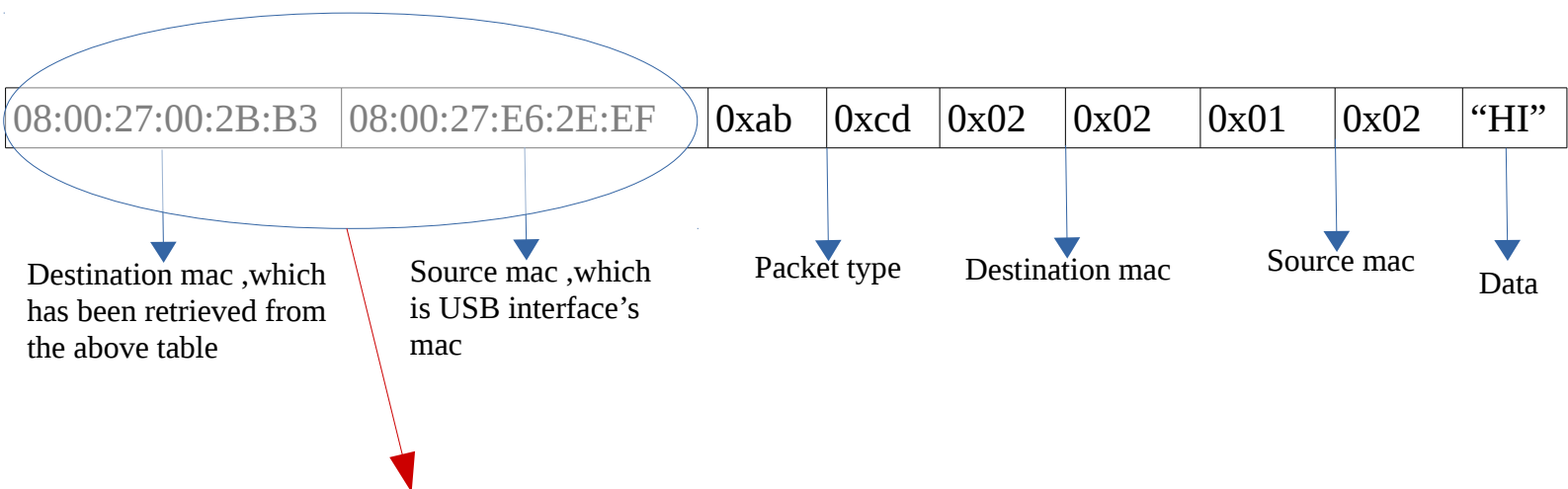
1. PCI Ethernet (Default LAN interface) -> 1.1 and MAC ->08:00:27:93:D6:34
2. USB Ethernet (LAN to USB converter) -> 2.1 and MAC -> 08:00:27:e6:2e:ef

***In my experiment PCI interface has been connected with netID = 1 and USB interface has been connected with netID = 2.***

if Device-A wants to send data to Device-B then device-A will create packet as :



After creating the packet Device-A will drop this packet into the network (LAN) and as destination has been given as router's PCI interface so router will accept the packet and all devices in the network will reject the packet and then according to the routing procedures described above router will reframe the packet and the updated packet will look like :



**These 12 bytes only have been changed because of routing.**

After updating the packet's 1<sup>st</sup> 12 bytes router will again send the packet to another network, with which router's USB interface has been connected . In my case that is network ID = 2. As destination of the packet has been changed to Device-B's mac address so Device-B will accept the packet and all other device in that network will reject this packet and Device-B will get the data that A has sent.

**Source Code :** [Link for Source Code](#)

**Compilation :** **For compiling the code the command :**

**tcc -ml “filename.c”**