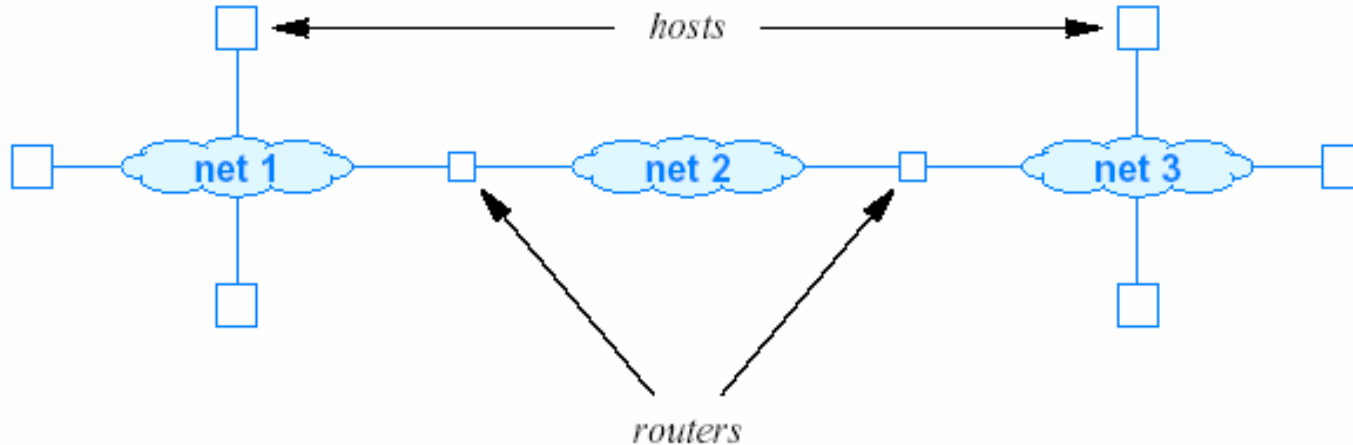

ECE 697J – Advanced Topics in Computer Networks

Packet Processing on End-Systems
9/11/03

Network Systems

- The obvious: hosts and routers



- Hosts can be variety of devices:
 - Workstations, servers, wireless PDAs, cell phones, etc.
- But there is more on different layers

Layer 2 Devices

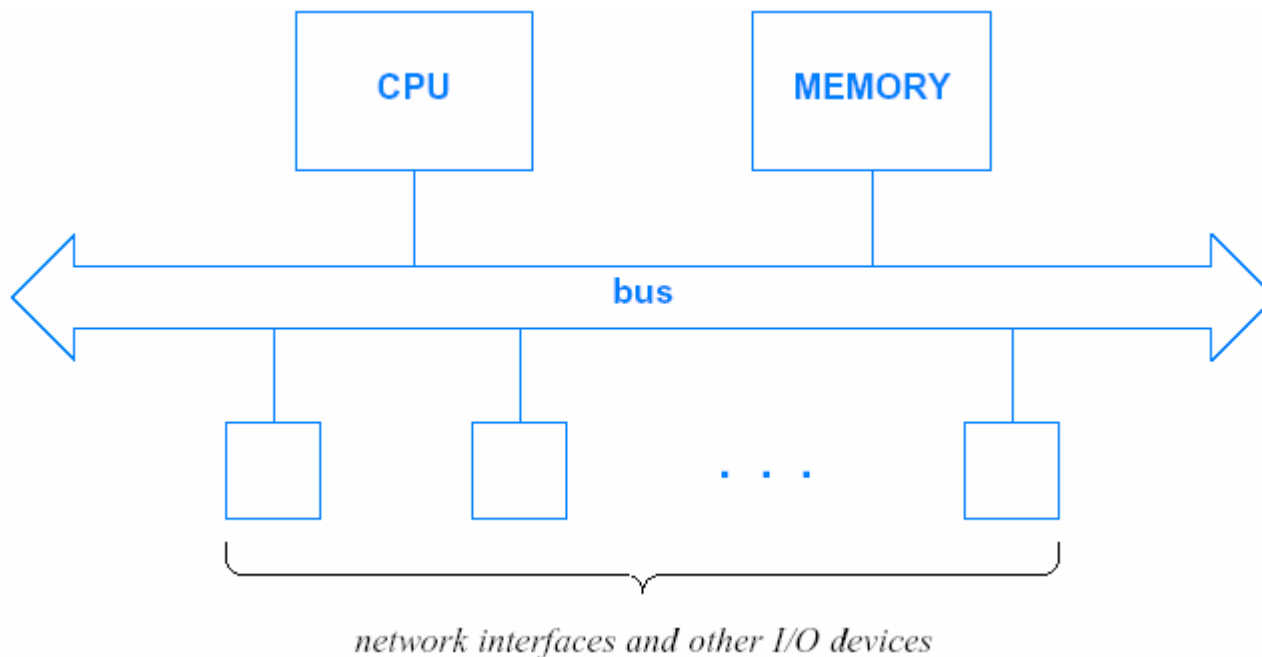
- Bridges:
 - Connection between two networks on data link level
 - Isolation of Ethernet collision domains
- Layer 2 switch:
 - Similar to bridge
 - Often with point-to-point connections on each port
 - High-throughput
- VLAN switch:
 - Supports several Virtual LANs
 - Layer 2 switch that emulates several smaller switches

Layer 3 & 4 Devices

- IP Router
 - Packet forwarding
 - IP destination address lookup, simple packet header processing
- Firewall
 - Blocks packets to certain internal addresses and ports
 - Maintains list of currently active connections
- Network Address Translator (NAT)
 - “Hides” subnet behind single external IP address
 - Rewrites packets to change IP address and port numbers
- Load Balancer
 - Distributes web requests to server farm
 - Uses Layer 4+ (or Layer 7) classification and TCP splicing
- Set-Top Box
 - Decrypts content for service subscribers
- Other devices: Monitor, Policer, Shaper, Analyzer

Packet Processing on Host

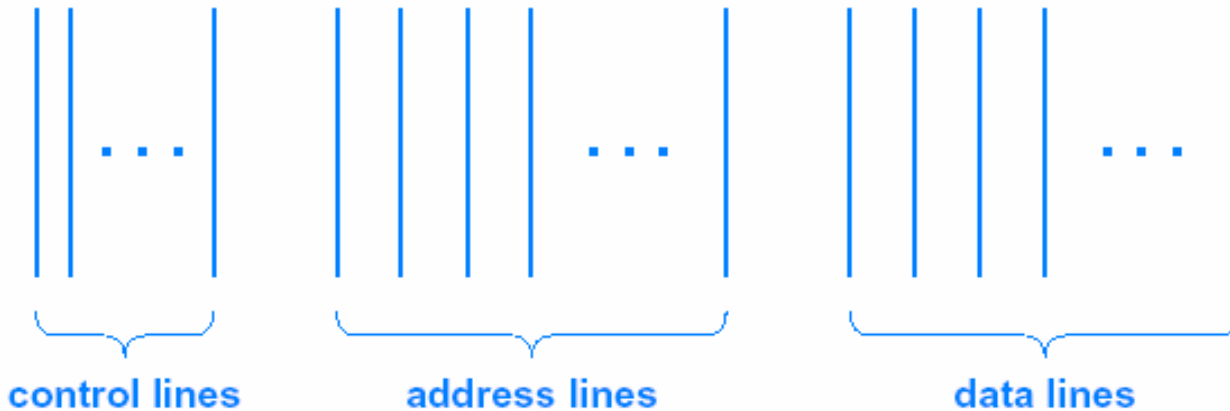
- “Conventional Computer System”:
 - Single CPU, memory, 1+ I/O devices, bus interconnect



- Network Interface Card (NIC) used for communication

Bus Interconnect

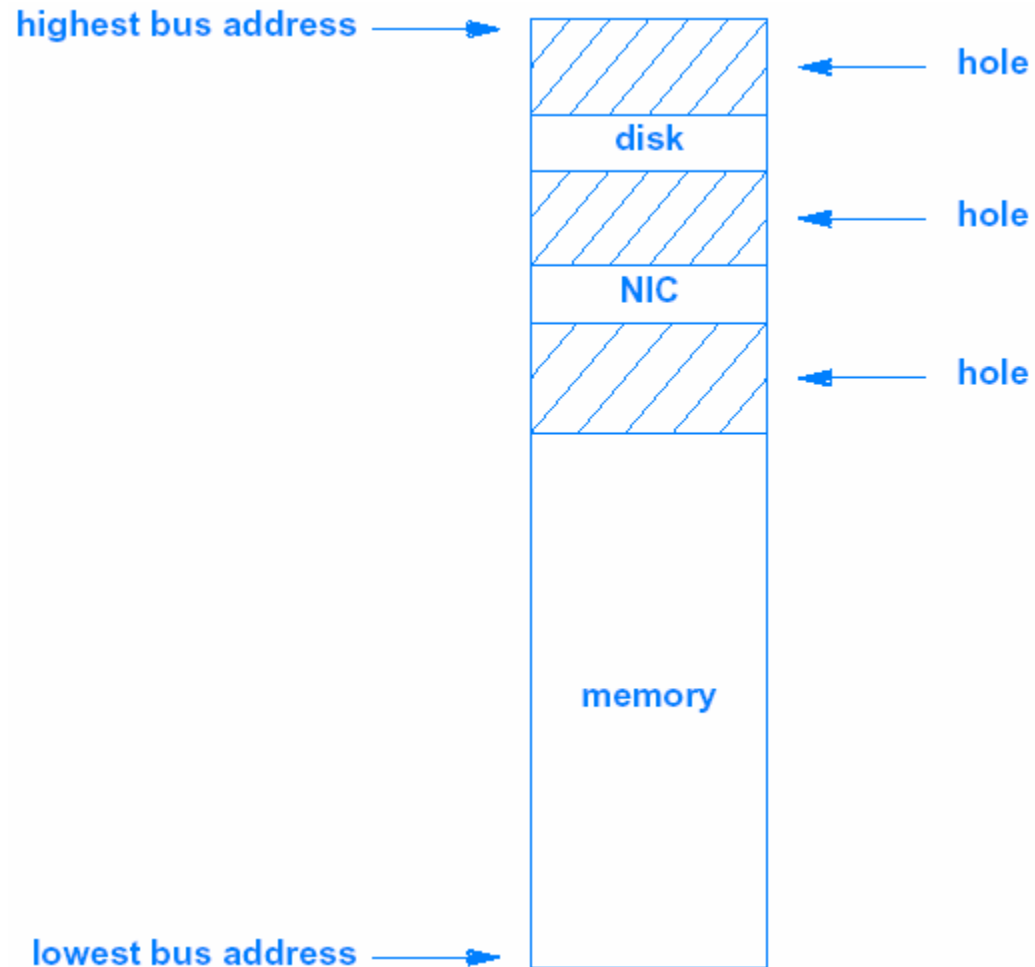
- Bus is parallel set of wires to which devices connect:



- Address is location of data
- Control indicates valid data, read or write, etc.
- Bus bandwidth determined by width and bus frequency
 - Bus BW = width * bus frequency
- Example: PCI bus on PC: 32 bits 66MHz
- How to distinguish reads and writes to different devices?

Bus Address Space

- Addresses “code” device information
 - Each device gets a unique set of addresses
 - Address space depends on application
 - Not entire address space needs to be allocated



Other Bus Issues

- Busses implement “fetch-store paradigm”
 - A bus operation is either a load (fetch) or a store – nothing else
- Control operations can be encoded as load/store ops
 - How?
- Real busses are more complicated
 - Bus arbiter implements access rules (e.g., priorities)
 - Some busses allow split-transaction
 - Some busses transfer data on each edge of the clock
 - Etc.
- For us: bus is necessary to communicate between CPU and NIC

NIC Functionality

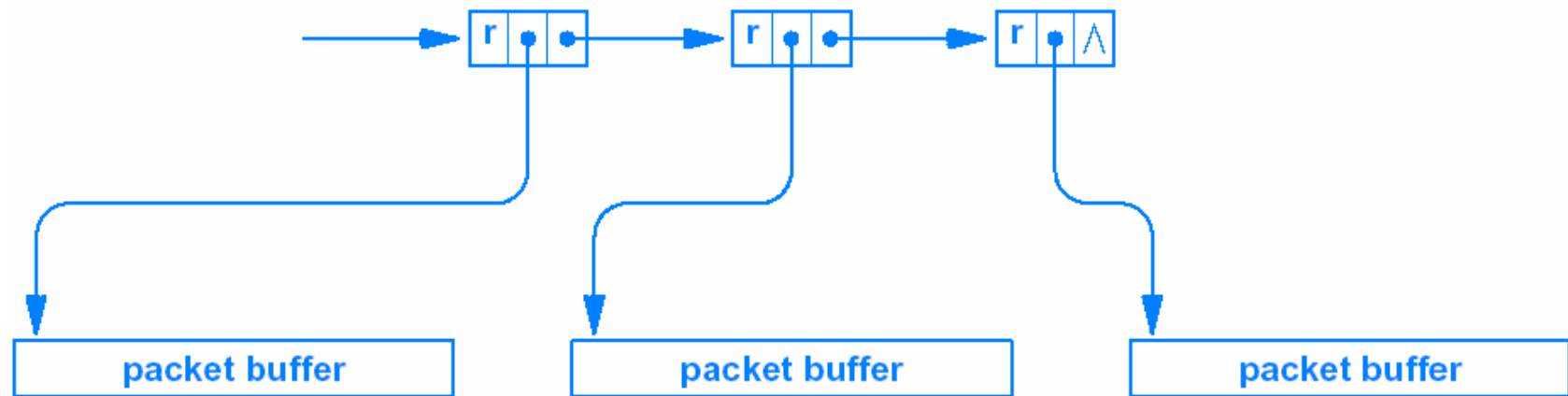
- NIC implements Layer 1 and 2 functionality
 - Sends and receives frames correctly
- Packet transmission:
 - CPU assembles packet in memory (typically including layer 2 header)
 - CPU transmits packet in chunks over bus to NIC
 - NIC buffers packet and sends it into the network
- Packet reception:
 - NIC has assigned buffer space
 - On packet arrival, packet is stored in that buffer
 - NIC informs CPU about packet
- Several inefficiencies!

NIC Optimization

- Onboard address recognition and filtering
 - Recognition of unicast and broadcast addresses
 - Multicast addresses more complex, why?
 - Multicast addresses are configured by CPU and limited
- Onboard packet buffering
 - NIC has memory to buffer packets, why?
 - Bursty traffic and contention on bus interconnect can require buffering
 - NIC can receive packets while transferring others to CPU
- Direct Memory Access (DMA)
 - Transfer of large amounts of data directly to/from memory
 - No CPU involvement
 - CPU tells NIC the location of buffer in memory

Operation and Data Chaining

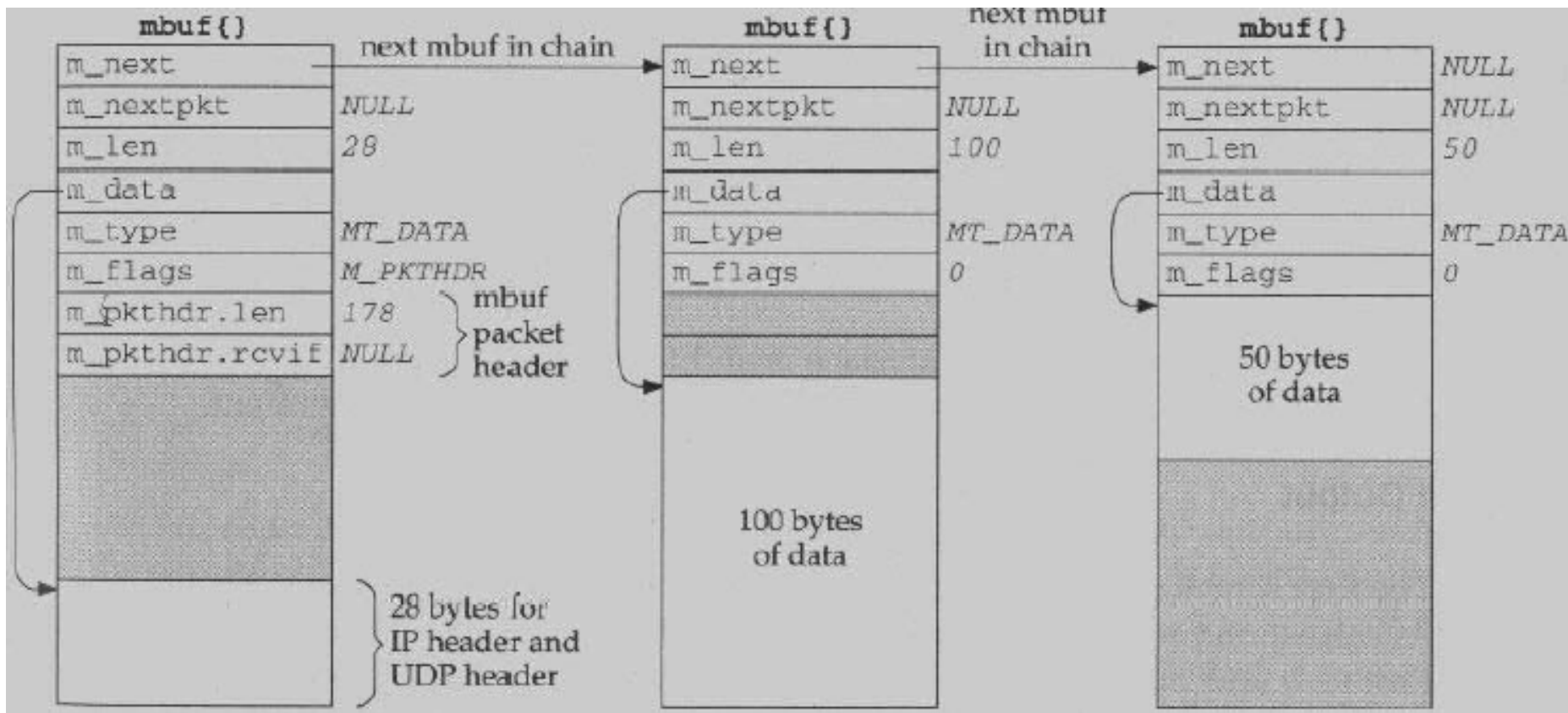
- CPU sets up linked list of buffers
 - NIC fills buffers as data comes in:



- “Command” in each buffer specifies receive or transmit
- Extra bit indicates if NIC has completed transfer

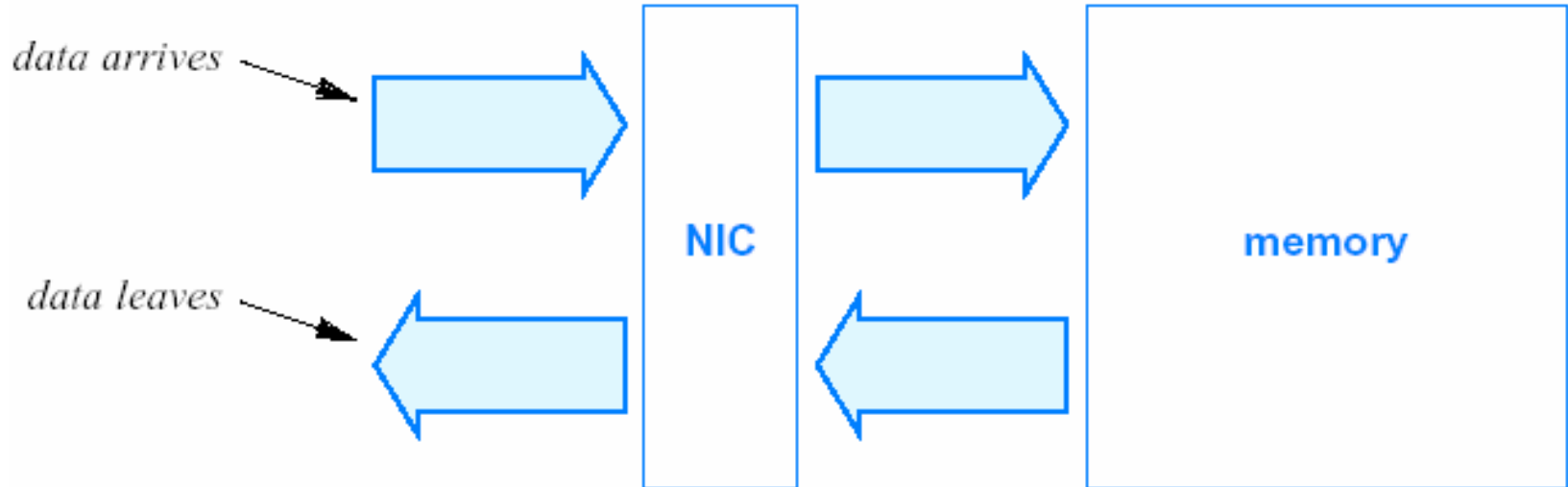
Data Chaining

- Data chaining used also in operating systems
- Unix BSD packet mbufs:



Data Flow Diagrams

- Illustration of data path:



- Side note:
 - In PC systems, the PCI bus is the bottleneck of the system

Next Class

- Packet processing functions
- Various data structures and algorithms
 - Table lookups and hashing
 - IP fragmentation and reassembly
 - IP forwarding
 - TCP connection recognition
 - TCP splicing
- Paper assignment
 - Who wants what?

Papers

- **IP lookup:** Marcel Waldvogel, George Varghese, Jon Turner, Bernhard Plattner. Scalable High Speed IP Lookups. In Proc. of ACM SIGCOMM 97, pages 25-36, Cannes, France, September 1997.
- **Router design:** S. Keshav and Rosen Sharma. Issues and Trends in Router Design. IEEE Communications Magazine, 36(5):144-151, May 1998.
- **Network applications (1):** George Apostolopoulos, David Aubespain, Vinod Peris, Prashant Pradhan, Debanjan Saha. Design, Implementation and Performance of a Content-Based Switch. In Proc. of IEEE INFOCOM 2000, pages 1117-1126, Tel Aviv, Israel, March 2000.
- **Network applications (2):** Li-wei Lehman, Stephen J. Garland, and David L. Tennenhouse. Active reliable multicast. In Proc. of IEEE INFOCOM 98, pages 581-589, San Francisco, CA, April 1998.
- **Active networking:** David L. Tennenhouse and David J. Wetherall. Towards an active network architecture. Computer Communication Review, 26(2):5-18, April 1996.
- **Scheduling:** M. Shreedhar and George Varghese. Efficient fair queuing using deficit round-robin. IEEE/ACM Transactions on Networking, 4(3): 375-385, June 1996.