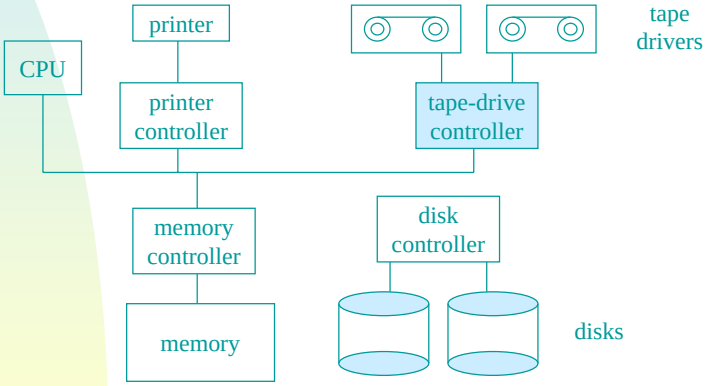
Contents

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Chapter 2
Computer-System Structure

Computer-System Structure

- Objective: General knowledge of the structure of a computer system.



- Device controllers: synchronize and manage access to devices.

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Booting

- Bootstrap program:
  - Initialize all aspects of the system, e.g., CPU registers, device controllers, memory, etc.
  - Load and run the OS
- Operating system: run *init* to initialize system processes, e.g., various daemons, login processes, after the kernel has been bootstrapped. (/etc/rc* & init or /sbin/rc* & init)

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

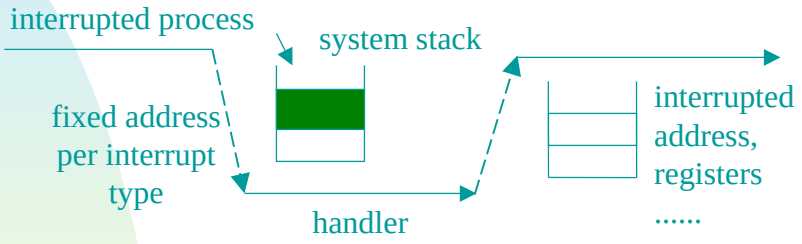
Interrupt

- Hardware interrupt, e.g. services requests of I/O devices
- Software interrupt, e.g. signals, invalid memory access, division by zero, system calls, etc – (trap)

- Procedures: generic handler or interrupt vector (MS-DOS,UNIX)

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Interrupt Handling Procedure



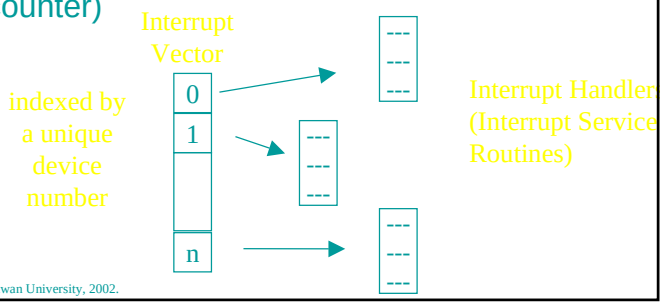
- Saving of the address of the interrupted instruction: fixed locations or stacks
- Interrupt disabling or enabling issues: lost interrupt?!

  prioritized interrupts → masking

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Interrupt Handling Procedure

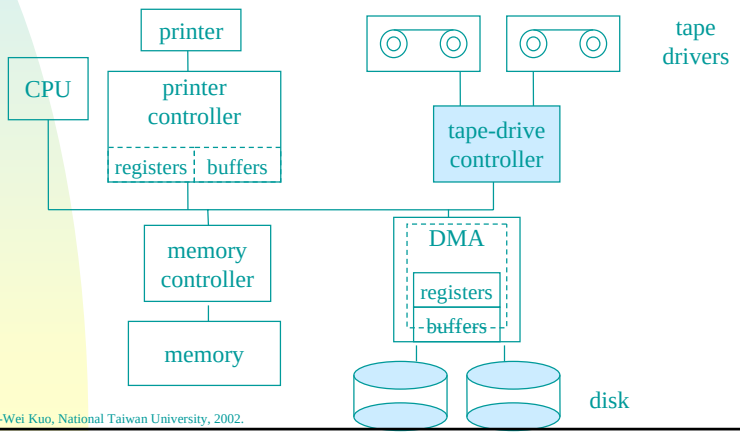
- Interrupt Handling
  - ➔ Save interrupt information
  - ➔ OS determine the interrupt type (by polling)
  - ➔ Call the corresponding handlers
  - ➔ Return to the interrupted job by the restoring important information (e.g., saved return addr. ➔ program counter)



* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

I/O Structure

- The device drivers are responsible of moving data between the peripheral devices and their local buffer storages.



* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

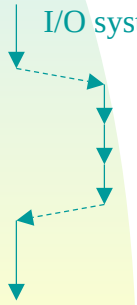
I/O Structure

- I/O operation
  a. CPU sets up specific controller registers within the controller.
  b. Read: devices → controller buffers → memory
     Write: memory → controller buffers → devices
  c. Notify the completion of the operation by triggering an interrupt

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

I/O Types

a. Synchronous I/O
   - Issues: overlapping of computations and IO activities, concurrent I/O activities, etc.



or
- wait instruction (idle till interrupted)
- looping

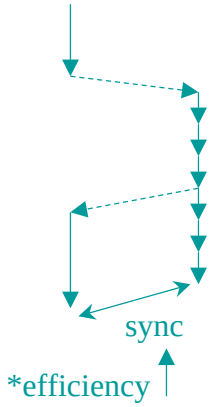
or
- polling
- wait for an interrupt

Loop: jmp *Loop*

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

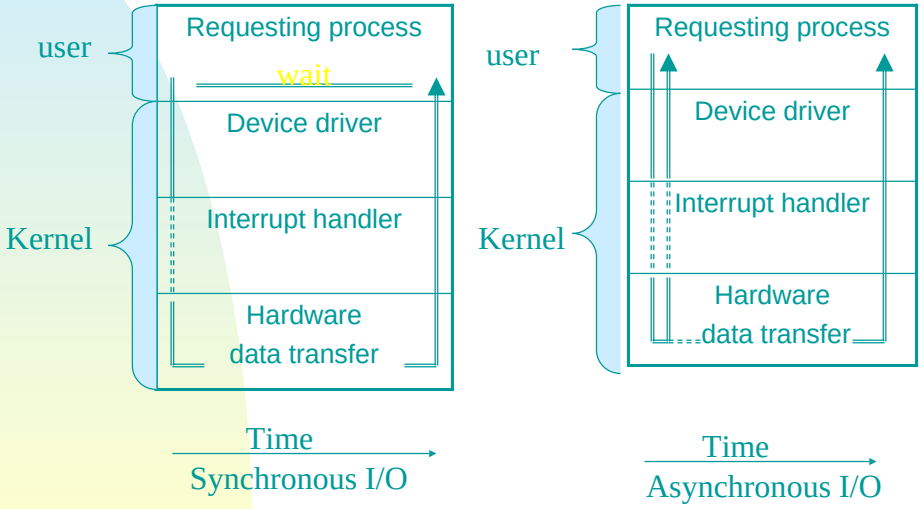
I/O types

b. Asynchronous I/O



* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

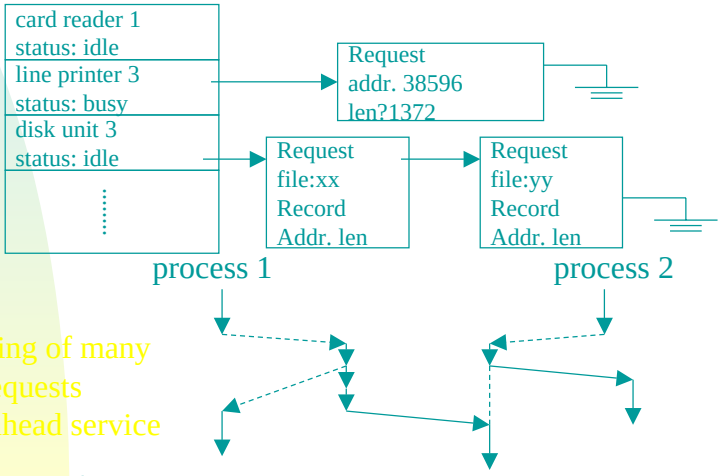
I/O Types



* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

I/O Types

- A Device-Status Table Approach



- Tracking of many I/O requests
- type-ahead service

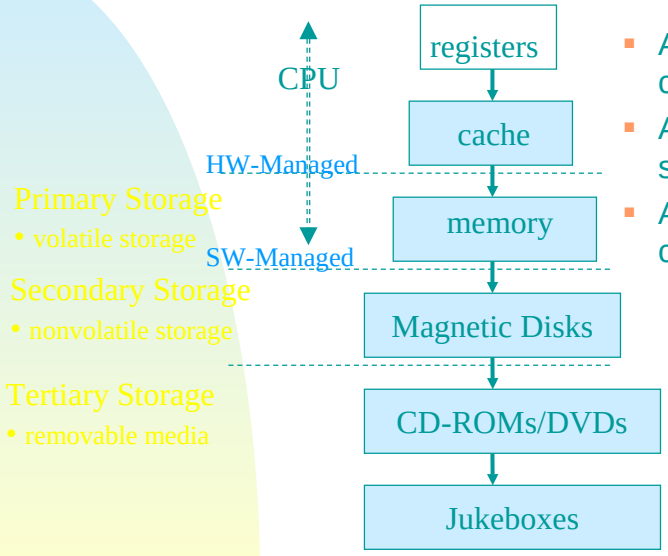
* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

DMA

- Goal: Release CPU from handling excessive interrupts!
  - E.g. 9600-baud terminal
    2-microsecond service / 1000 microseconds
    High-speed device:
    2-microsecond service / 4 microseconds
- Procedure
  - Execute the device driver to set up the registers of the DMA controller.
  - DMA moves blocks of data between the memory and its own buffers.
  - Transfer from its buffers to its devices.
  - Interrupt the CPU when the job is done.

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Storage Structure



- Access time: a cycle
- Access time: several cycles
- Access time: many cycles

Primary Storage
- volatile storage

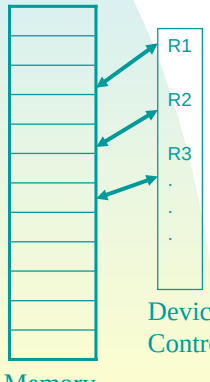
Secondary Storage
- nonvolatile storage

Tertiary Storage
- removable media

* Differences: Size, Cost, Speed, Volatility

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

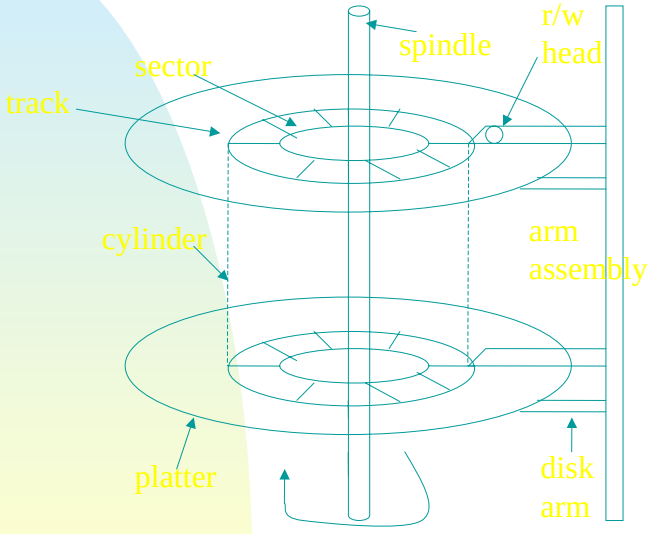
Memory



- Processor can have direct access!
- Intermediate storage for data in the registers of device controllers
- Memory-Mapped I/O (PC & Mac)
  (1) Frequently used devices
  (2) Devices must be fast, such as video controller, or special I/O instructions is used to move data between memory & device controller registers
- Programmed I/O – polling
  - or interrupt-driven handling

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Magnetic disks



- Transfer Rate
- Random-Access Time
  - Seek time in *x* ms
  - Rotational latency in *y* ms
    - 60~200 times/sec

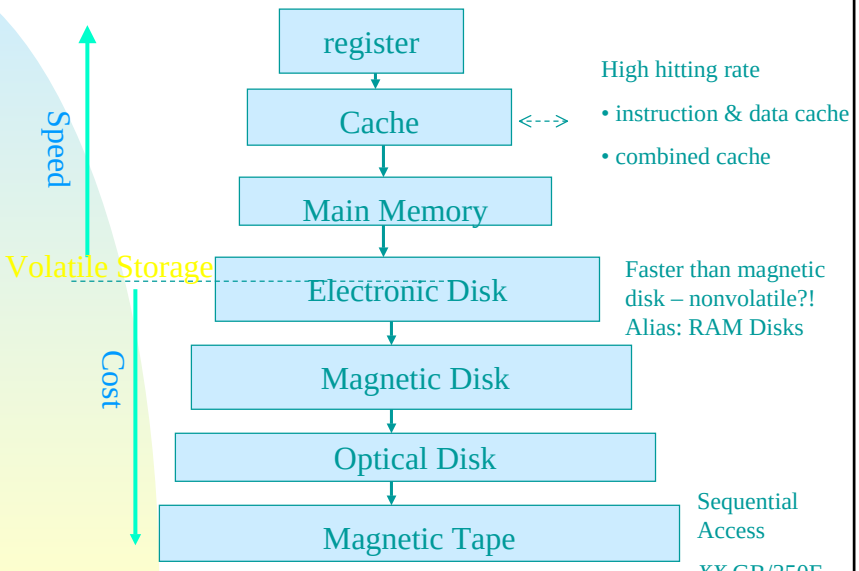
* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Magnetic Disks

- Disks
  - Fixed-head disks:
    - More r/w heads v.s. fast track switching
  - Moving-head disks (hard disk)
  - Primary concerns:
    - Cost, Size, Speed
  - Computer → host controller → disk controller → disk drives (cache ←→ disks)
- Floppy disk
  - slow rotation, low capacity, low density, but less expensive
- Tapes: backup or data transfer bet machines

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Storage Hierarchy



* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Storage Hierarchy

- Caching
  - Information is copied to a faster storage system on a temporary basis
  - Assumption: Data will be used again soon.
    - Programmable registers, instr. Cache, etc.
- Cache Management
  - Cache Size and the Replacement Policy
- Movement of Information Between Hierarchy
  - Hardware Design & Controlling Operating Systems

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Storage Hierarchy

- Coherency and Consistency
  - Among several storage levels (vertical)
    - Multitasking vs unitasking
  - Among units of the same storage level , (horizontal), e.g. cache coherency
    - Multiprocessor or distributed systems

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Hardware Protection

- Goal:
  - Prevent errors and misuse!
    - E.g., input errors of a program in a simple batch operating system
    - E.g., the modifications of data and code segments of another process or OS
- Dual-Mode Operations – a mode bit
  - User-mode executions except those after a trap or an interrupt occurs.
  - Monitor-mode (system mode, privileged mode, supervisor mode)
    - Privileged instruction:machine instructions that may cause harm

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

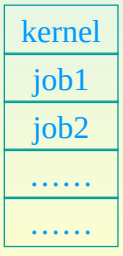
Hardware Protection

- System Calls – trap to OS for executing privileged instructions.
- Resources to protect
  - I/O devices, Memory, CPU
- I/O Protection (I/O devices are scare resources!)
  - I/O instructions are privileged.
    - User programs must issue I/O through OS
    - User programs can never gain control over the computer in the system mode.

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Hardware Protection

- Memory Protection
  - Goal: Prevent a user program from modifying the code or data structures of either the OS or other users!
  - Instructions to modify the memory space for a process are privileged.



Base register

Limit register

⇔ Check for every memory address by hardware

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Hardware Protection

- CPU Protection
  - Goal
    - Prevent user programs from sucking up CPU power!
  - Use a timer to implement time-sharing or to compute the current time.
    - Instructions that modify timers are privileged.
  - Computer control is turned over to OS for every time-slice of time!
    - Terms: time-sharing, context switch

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Network Structure

- Local-Area Network (LAN)
  - Characteristics:
    - Geographically distributed in a small area, e.g., an office with different computers and peripheral devices.
    - More reliable and better speed
      - High-quality cables, e.g., twisted pair cables for 10BaseT Ethernet or fiber optic cables for 100BaseT Ethernet
  - Started in 1970s
  - Configurations: multiaccess bus, ring, star networks (with gateways)

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

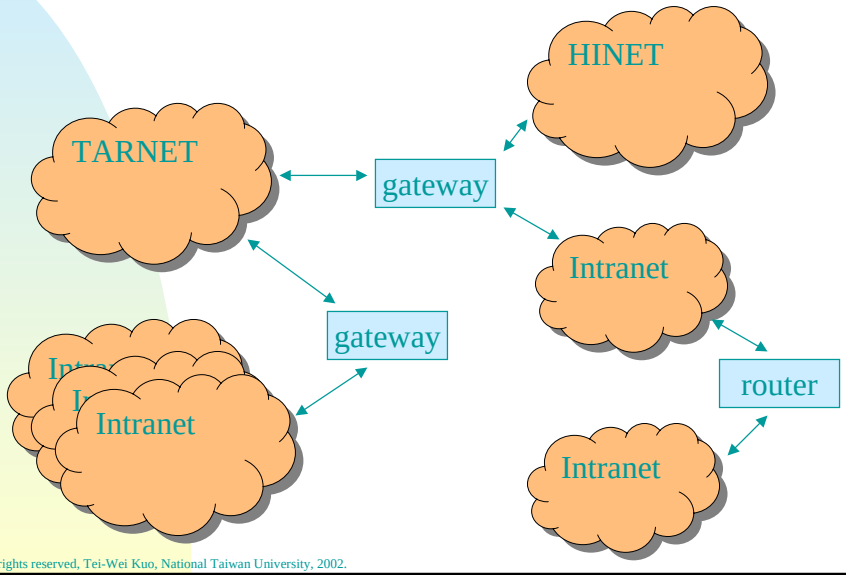
Network Structure

- Wide-Area Network (WAN)
  - Emerged in late 1960s (Arpanet in 1968)
- World Wide Web (WWW)
  - Utilize TCP/IP over ARPANET/Internet.

• Definition of “Intranet”: roughly speaking for any network under one authorization, e.g., a company or a school.
  • Often in a Local Area Network (LAN), or connected LAN’s.
  • Having one (or several) gateway with the outside world.
  • In general, it has a higher bandwidth because of a LAN.

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Network Structure – WAN



* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Network Structure – WAN

- Router
  - With a Routing table
    - Use some routing protocol, e.g., to maintain network topology by broadcasting.
  - Connecting several subnets (of the same IP-or-higher-layer protocols) for forwarding packets to proper subnets.
- Gateway
  - Functionality containing that of routers.
  - Connecting several subnets (of different or the same networks, e.g., Bitnet and Internet)for forwarding packets to proper subnets.

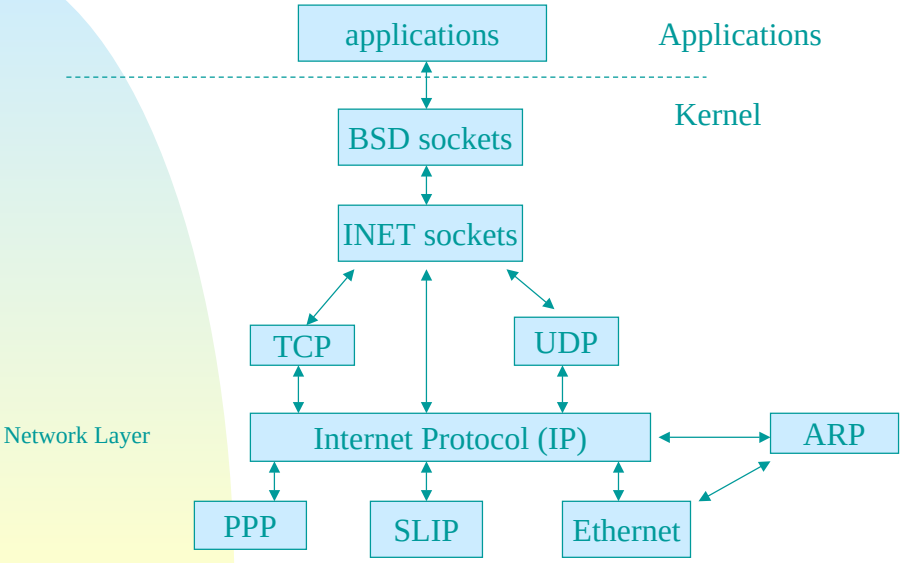
* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Network Structure – WAN

- Connections between networks
  - T1: 1.544 mbps, T3: 45mbps (28T1)
    - Telephone-system services over T1
- Modems
  - Conversion of the analog signal and digital signal

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

Network Layers in Linux



* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

TCP/IP

- IP Address:
  - 140.123.101.1
    - 256*256*256*256 combinations
      - 140.123 -> Network Address
      - 101.1 -> Host Address
    - Subnet:
      - 140.123.101 and 140.123.102
  - Mapping of IP addresses and host names
    - Static assignments: /etc/hosts
    - Dynamic acquisition: DNS (Domain Name Server)
      - /etc/resolv.confg
    - If /etc/hosts is out-of-date, re-check it up with DNS!
  - Domain name: cs.ccu.edu.tw as a domain name for 140.123.100, 140.123. 101, and 140.123.103

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

TCP/IP

- Transmission Control Protocol (TCP)
  - Reliable point-to-point packet transmissions.
  - Applications which communicate over TCP/IP with each another must provide IP addresses and port numbers.
    - /etc/services
    - Port# 80 for a web server.
- User Datagram Protocol (UDP)
  - Unreliable point-to-point services.
- Both are over IP.

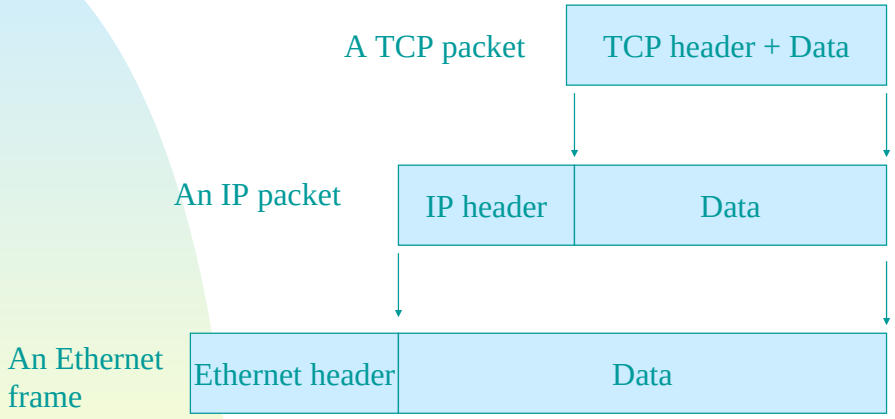
* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

TCP/IP

- Mapping of Ethernet physical addresses and IP addresses
  - Each Ethernet card has a built-in Ethernet physical address, e.g., 08-01-2b-00-50-A6.
  - Ethernet cards only recognize frames with their physical addresses.
  - Linux uses ARP (Address Resolution Protocol) to know and maintain the mapping.
    - Broadcast requests over Ethernet for IP address resolution over ARP.
    - Machines with the indicated IP addresses reply with their Ethernet physical addresses.

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.

TCP/IP



- Each IP packet has an indicator of which protocol used, e.g., TCP or UDP

* All rights reserved, Tei-Wei Kuo, National Taiwan University, 2002.