信息科学与技术学院
School of Information Science and Technology

CS 110
Computer Architecture
More I/O

Instructors: Siting Liu & Yuan Xiao
Course website: https://faculty.sist.shanghaitech.edu.cn/liust/courses/CS110.html
School of Information Science and Technology (SIST)
ShanghaiTech University

2026/6/9

Administratives

- Final exam, June 25th 8am-10am; you can bring **3**-page A4-sized double-sided cheat sheet, **handwritten** only! (Teaching center 102/202); the whole course will be covered. No electronic devices (no smart watches, no calculators, no phones, etc.)
- All the labs have been released! ✌️
- Project 3 ddl June 11th
- Project 4 released, ddl June 18th. **Will be checked during lab sessions June 18th/22nd/23rd.** ✌️
- HW 7 ddl June 11th.
- Discussion June 12th on *Final Review*.

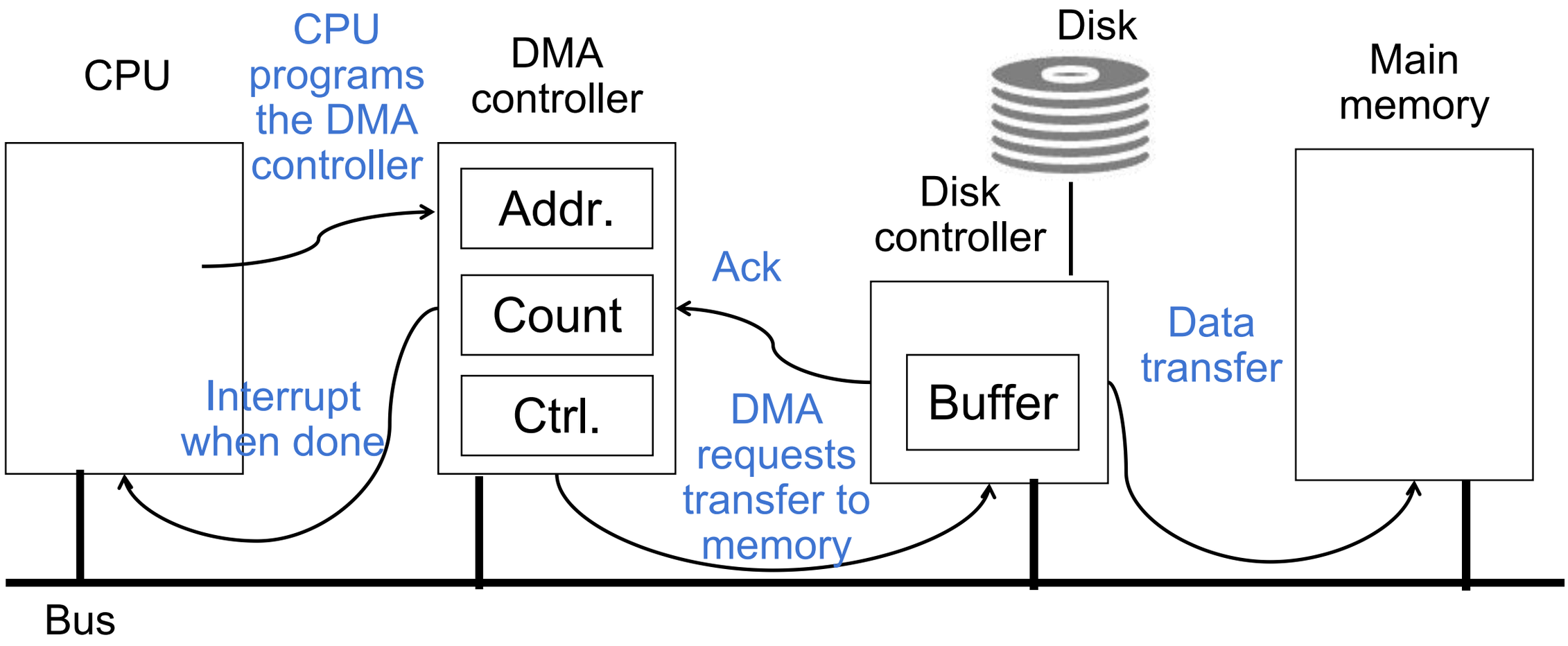
Review: I/O

- "Memory mapped I/O": Device control/data registers mapped to CPU address space
- CPU synchronizes with I/O device:
  - Polling: wastes processor resources;
  - Interrupts: Nothing to do with no I/O activity; high cost when lots of I/O - expensive saving states and thrashing caches;
    - e.g. mouse and keyboard; what about high data rate (e.g. network, disk)?
- "Programmed I/O":
  - CPU executes `lw/sw` instructions for all data movement to/from devices
  - CPU spends time doing 2 things:
    - Get data from device to main memory
    - Use data to compute
  - Not ideal, CPU can do something more important and complex

Direct Memory Access (DMA)

- ~~"Programmed I/O":~~ **DMA**
  - ~~CPU execs lw/sw instructions for all data movement to/from devices~~
  - CPU spends time doing ~~2 things~~:
    1. ~~Getting data from device to main memory~~
    2. Using data to compute
- Allow I/O devices to directly read/write main memory;
- New hardware: the DMA engine
- DMA engine contains registers written by CPU
  - Memory address to place data
  - # of bytes
  - I/O device #, direction of transfer
  - Unit of transfer, amount to transfer per burst

DMA Transfer



DMA: Incoming Data

- Receive interrupt from device
- CPU takes interrupt, begins transfer
  - Instructs DMA engine/device to place data @ certain address
- Device/DMA engine handle the transfer
  - CPU is free to execute other things
- Upon completion, Device/DMA engine interrupt the CPU again

DMA: Outgoing Data

- CPU decides to initiate transfer, confirms that external device is ready
- CPU begins transfer
  - Instructs DMA engine/device that data is available @ certain address
- Device/DMA engine handle the transfer
  - CPU is free to execute other things
- Device/DMA engine interrupt the CPU again to signal completion

DMA: New Problems 1

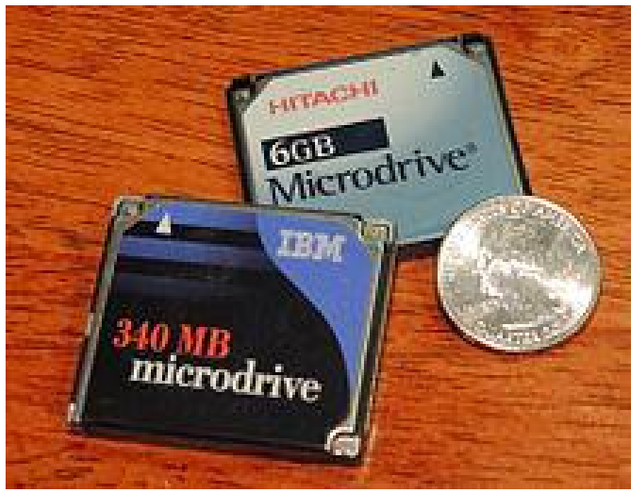
- Where in the memory hierarchy do we plug in the DMA engine?
- Two extremes:
  - Between CPU and L1:
    - Pro: Free coherency
    - Con: Thrash the CPU's working set with transferred data
  - Between Last-level cache and main memory:
    - Pro: Don't mess with caches
    - Con: Need to explicitly manage coherency

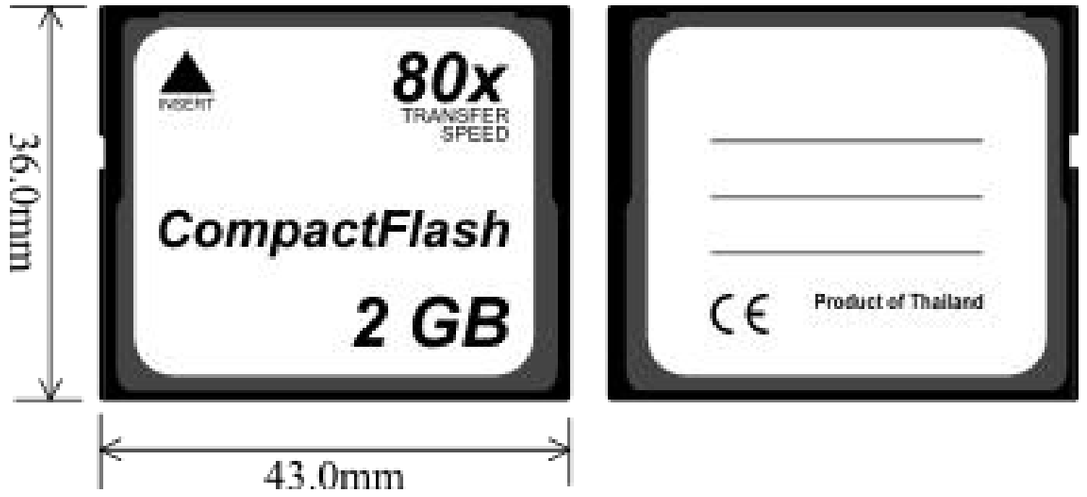
DMA: New Problems 2

- How do we arbitrate between CPU and DMA Engine/Device access to memory?
- Three options:
  - Burst Mode
    - Start transfer of data block, CPU cannot access memory in the meantime
  - Cycle Stealing Mode
    - DMA engine transfers a byte, releases control, then repeats - interleaves processor/DMA engine accesses
  - Transparent Mode
    - DMA transfer only occurs when CPU is not using the system bus

Common I/O device: SSD (Flash Memory)

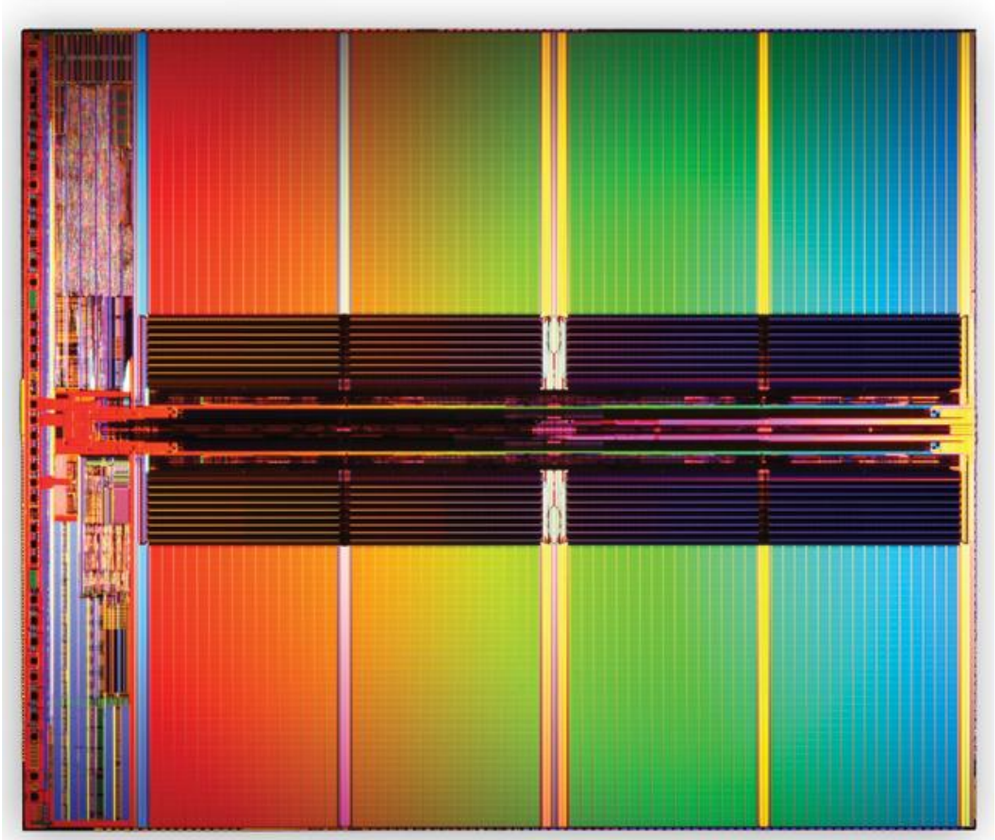
- >15 years ago: Microdrives and Flash memory (e.g., CompactFlash) went head-to-head
  - Both non-volatile (retains contents without power supply)
  - Flash benefits: lower power, seldom crashes (no moving parts, need to spin μdrives up/down)
  - Disk cost = fixed cost of motor + arm mechanics, but actual magnetic media cost very low
  - Flash cost = most cost/bit of flash chips
  - Over time, cost/bit of flash came down, became cost-competitive



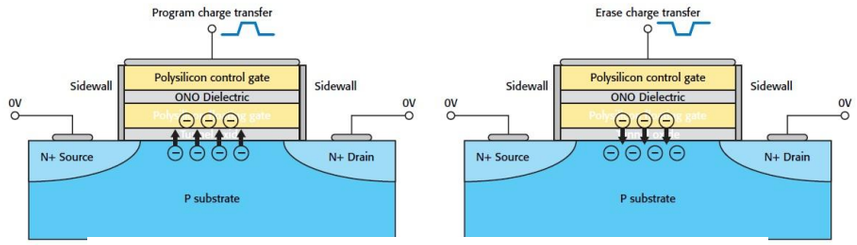


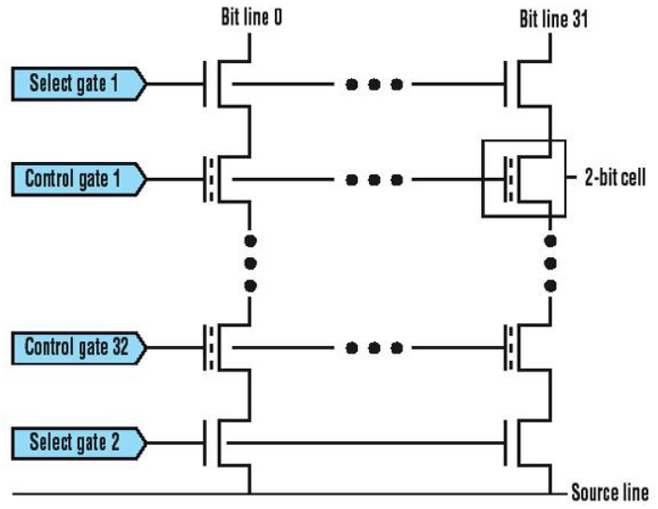
Flash Memory, SSD Technology

- NMOS transistor with an additional conductor between gate and source/drain which “traps” electrons. The presence/absence is a 1 or 0;
- Memory cells can withstand a limited number of program-erase cycles. Controllers use a technique called wear leveling to distribute writes as evenly as possible across all the flash blocks in the solid-state drive (SSD);
- Even compute using flash memory, more in EE219.



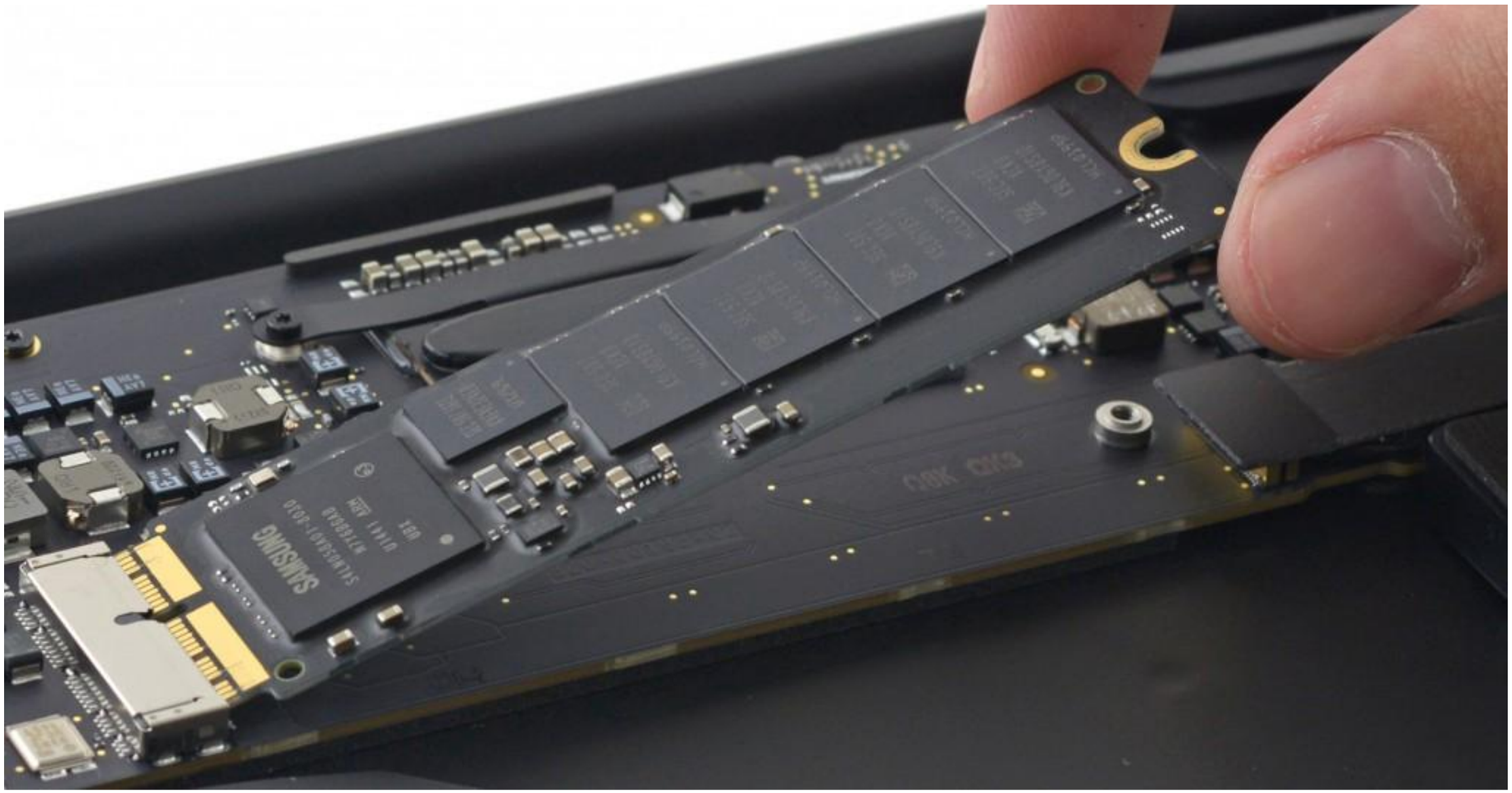
2. Micron’s triple-level cell (TLC) flash memory stores 3 bits of data in each transistor.



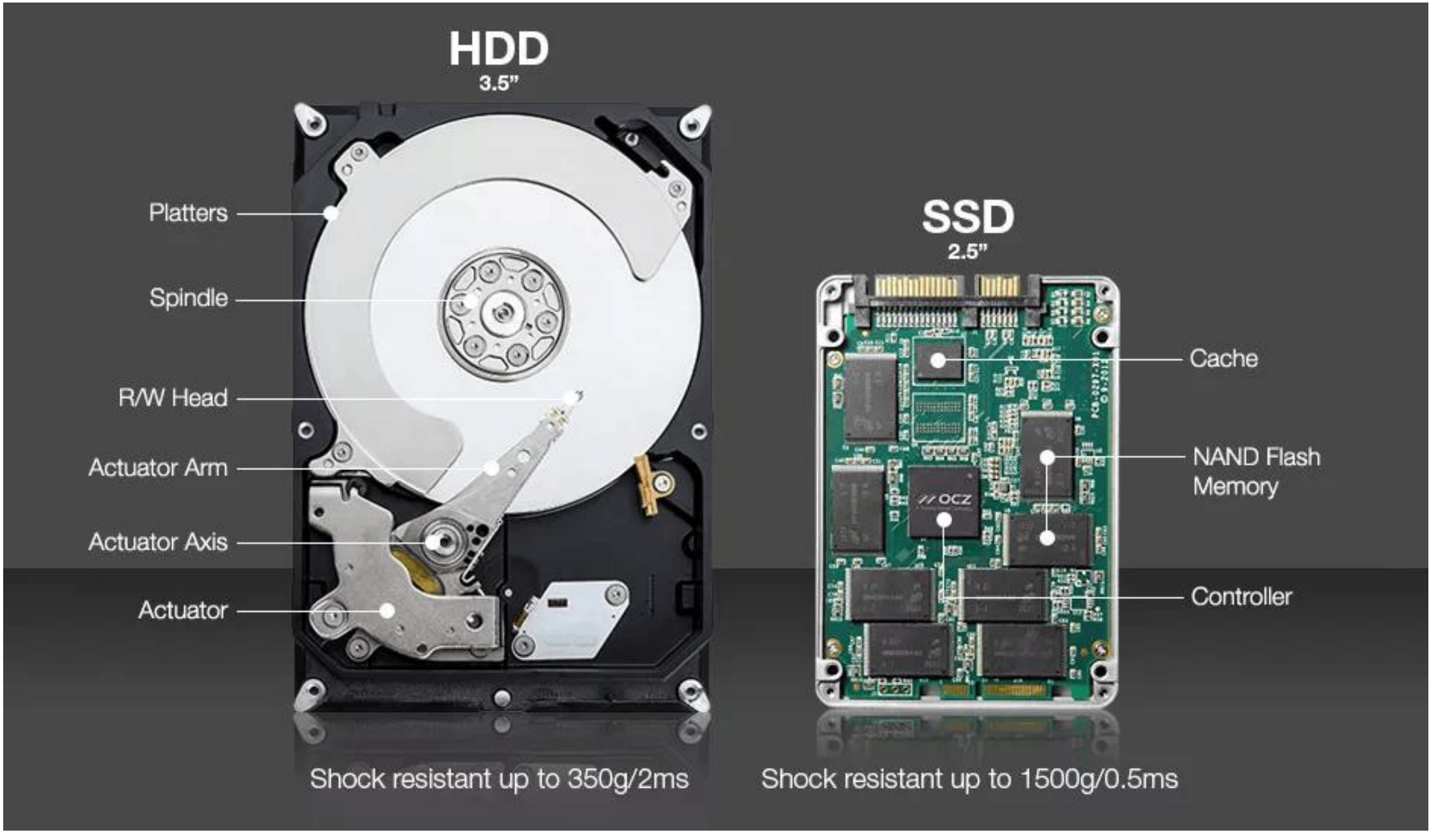


In the basic functional block used in multilevel NAND flash memories, 32 rows of bit lines and 32 control-gate lines form a building block that’s repeated many times to form the memory array. The select gate lines are used with the control gate lines to control access to the array.

SSD in Real Computing Systems



HDD vs. SSD


HDD
3.5"
Platters
Spindle
R/W Head
Actuator Arm
Actuator Axis
Actuator
SSD
2.5"
Cache
NAND Flash Memory
Controller
Shock resistant up to 350g/2ms
Shock resistant up to 1500g/0.5ms

HDD vs. SSD

	HDD	SSD
Cost per bit	Cheaper	
Capacity	Larger	
Durable		More durable (shock-resistant)
Performance		Faster
Power consumption		Lower
Size		More compact
Endurance	Better	

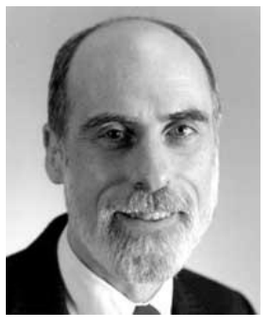
Common I/O Devices 2: Networking

- Originally sharing I/O devices between computers
  - E.g., printers
- Then communicating between computers
  - E.g., file transfer protocol (FTP)
- Then communicating between people
  - E.g., e-mail
- Then communicating between networks of computers
  - E.g., file sharing, www, social network…

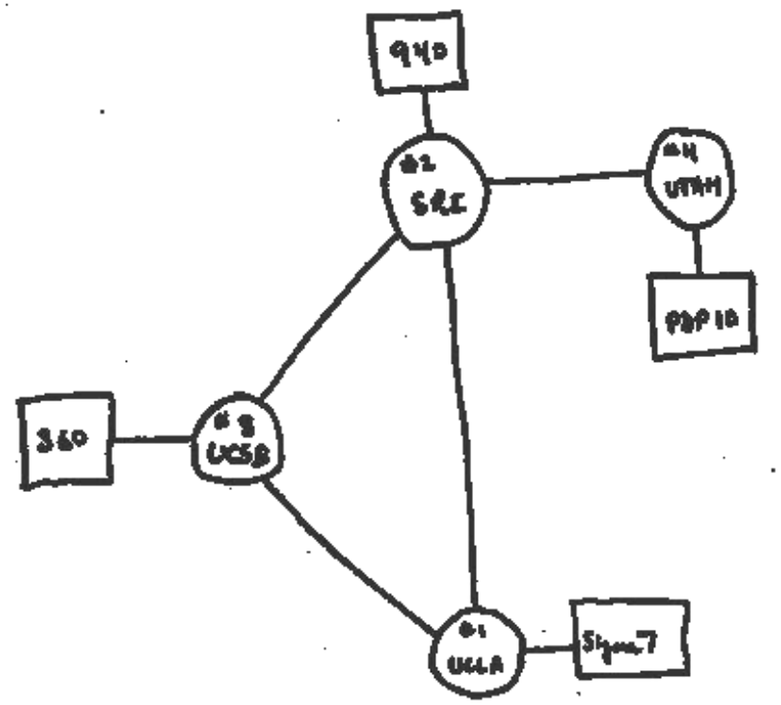
The Internet (1962)

- History
- 1963: JCR Licklider, while at DoD's ARPA, writes a memo describing desire to connect the computers at various research universities: Stanford, Berkeley, UCLA, ...
- 1969 : ARPA deploys 4 "nodes" @ UCLA, SRI, Utah, & UCSB
- 1973 Robert Kahn & Vint Cerf invent TCP, now part of the Internet Protocol Suite
- Internet growth rates
  - Exponential since start!

"Lick"



Vint Cerf



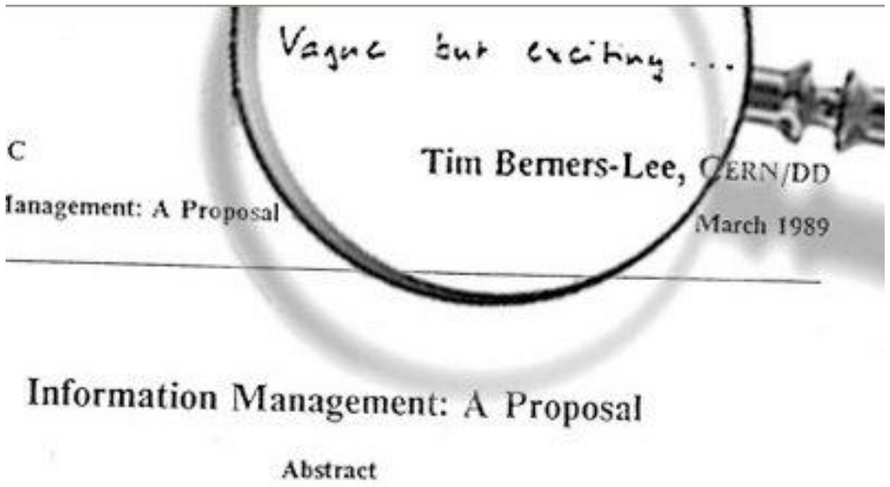
en.wikipedia.org/wiki/Internet_Protocol_Suite

The World Wide Web (1989)

- "System of interlinked hypertext documents on the Internet"
- History
  - 1945: Vannevar Bush describes hypertext system called "memex" in article
  - 1989: Sir Tim Berners-Lee proposed and implemented the first successful communication between a Hypertext Transfer Protocol (HTTP) client and server using the internet.
  - ~2000 Dot-com entrepreneurs rushed in, 2001 bubble burst
- Today : Access anywhere! https (security)!

Tim Berners-Lee

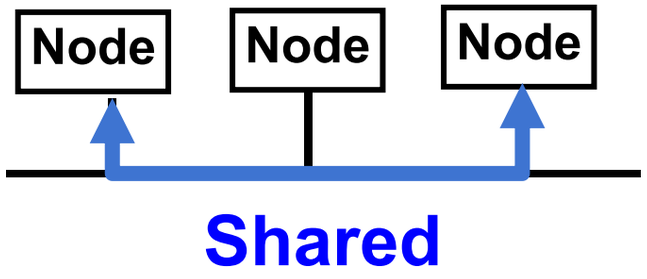
World's First web server in 1990

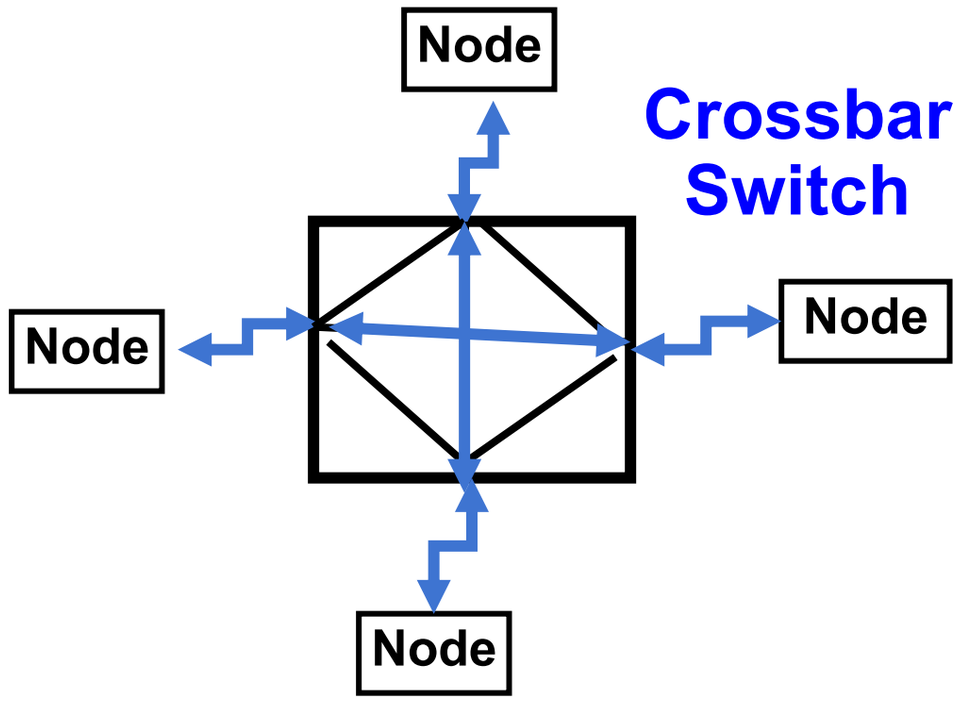


en.wikipedia.org/wiki/History_of_the_World_Wide_Web

Shared vs. Switch-Based Networks

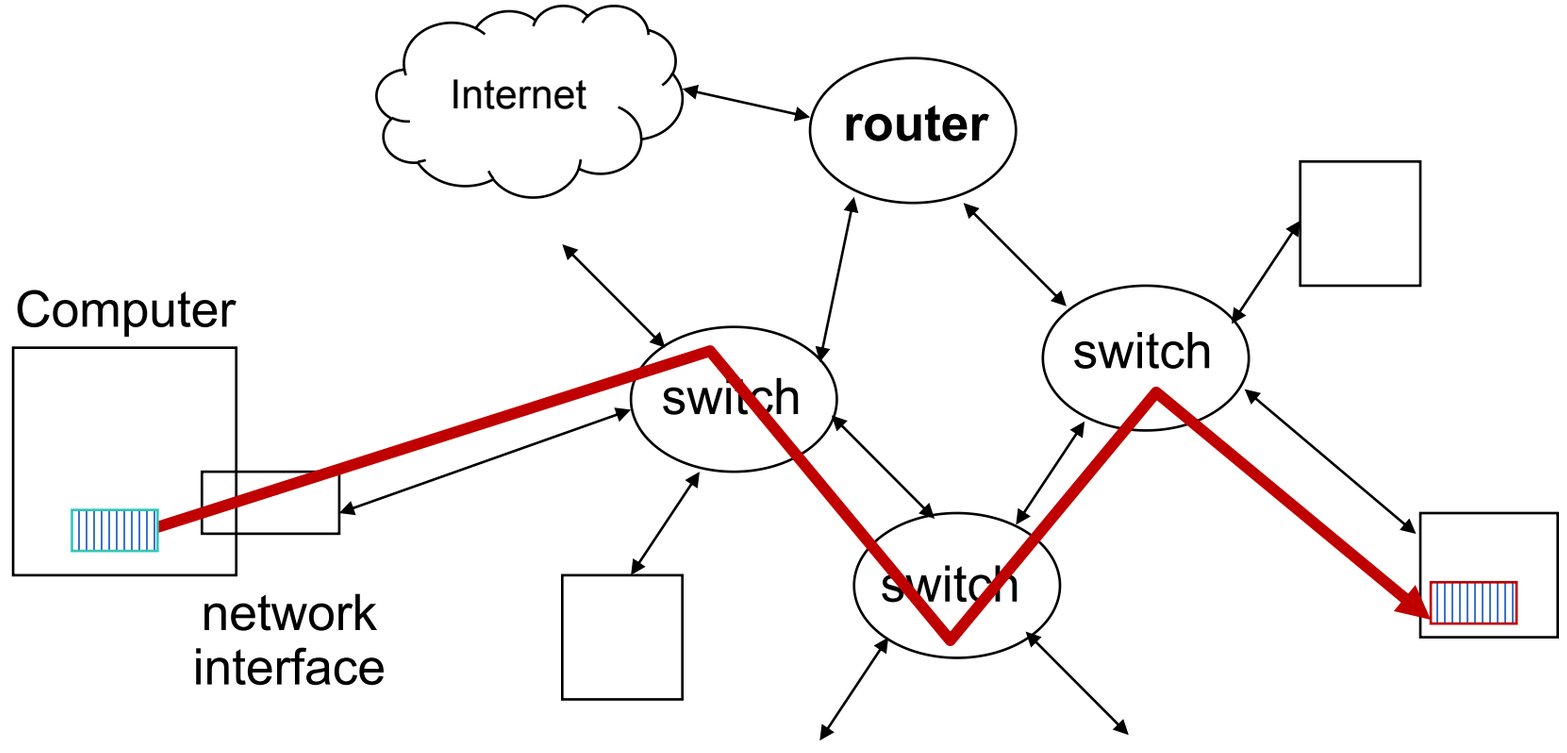
- Shared vs. Switched:
  - Shared: 1 at a time
  - Switched: pairs (“point-to-point” connections) communicate at same time
- Aggregate bandwidth (BW) in switched network is many times that of shared
  - point-to-point faster since no arbitration, simpler interface





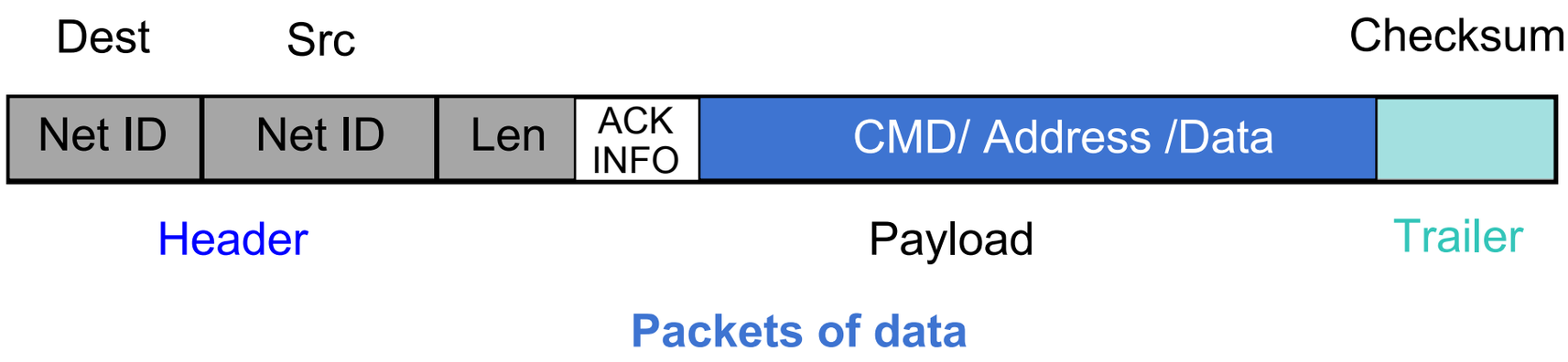
What Makes Networks Work?

- Links connect switches and/or routers to each other and to computers or devices
- Ability to name the components and to route packets of information - messages - from a source to a destination
- Layering, redundancy, protocols, and encapsulation as means of abstraction (big idea in Computer Architecture)



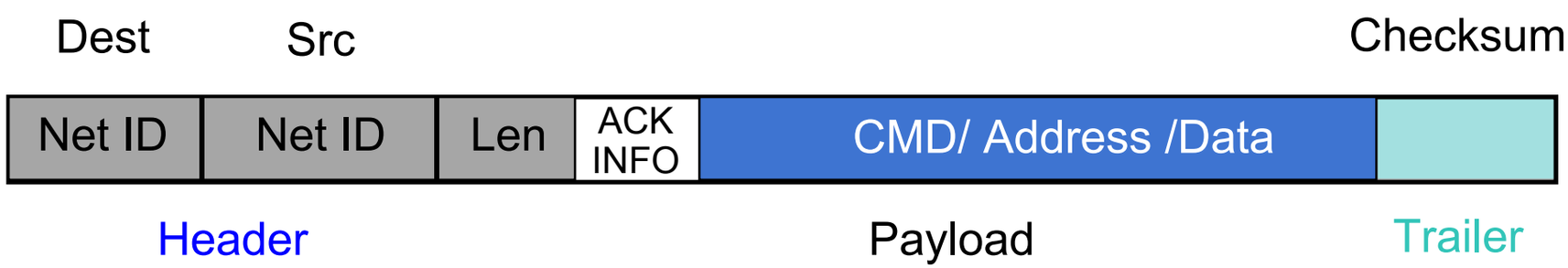
Software Protocol to Send and Receive

- SW Send steps
  - 1: Application copies data to OS buffer
  - 2: OS calculates checksum, starts timer
  - 3: OS sends data to network interface HW and says start
- SW Receive steps
  - 3: OS copies data from network interface HW to OS buffer
  - 2: OS calculates checksum, if OK, send ACK; if not, delete message (sender resends when timer expires)
  - 1: If OK, OS copies data to user address space, & signals application to continue



Protocols for Networks of Networks?

- What does it take to send packets across the globe?
  - Bits on wire or air
  - Packets on wire or air
  - Delivery packets within a single physical network
  - Deliver packets across multiple networks
  - Ensure the destination received the data
  - Create data at the sender and make use of the data at the receiver



Packets of data

Protocols for Networks of Networks?

- Lots to do and at multiple levels!
- Use abstraction to cope with complexity of communication
- Hierarchy of layers:
  - Applications (chat client, game -> websocket), (WWW -> http), (email -> SMTP/pop3), (ftp -> ftp), etc.
  - Transport (TCP, UDP)
  - Network (IP)
  - Data Link Layer (Ethernet)
  - Physical Link (copper wires, wireless, etc.)

Protocol Family Concept

- Protocol: packet structure and control commands to manage communication
- **Protocol families (suites)**: a set of cooperating protocols that implement the network stack

- Key to protocol families is that communication occurs logically at the same level of the protocol, called peer-to-peer…but is implemented via services at the next lower level

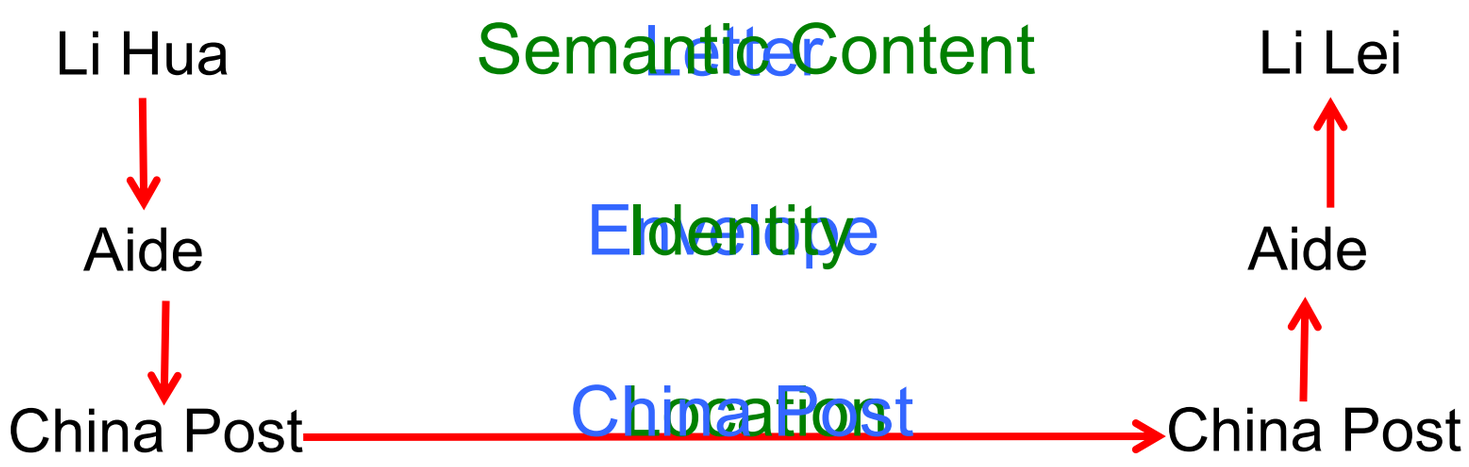
- **Encapsulation**: carry higher level information within lower level “envelope”

Analogy: Send a Letter

- Li Hua writes letter to Li Lei
  - Folds letter and hands it to assistant
- Assistant:
  - Puts letter in envelope with Li Lei's full name
  - Takes to China Post
- China Post Office
  - Puts letter in larger envelope
  - Puts name and street address on China Post envelope
  - Puts package on China Post delivery truck
- China Post delivers to other company

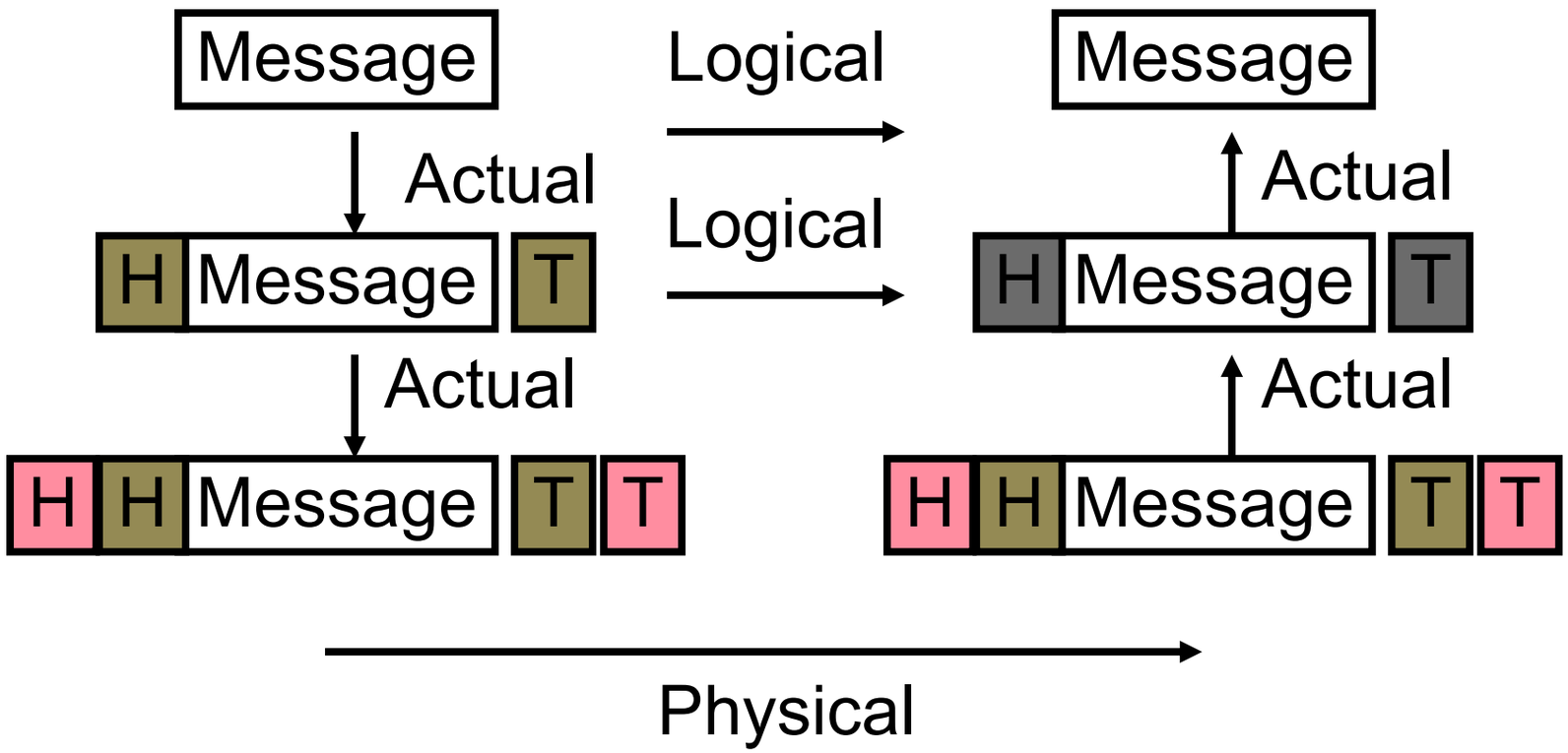
The Path of the Letter

- “Peers” on each side understand the same things
- Lowest level has most packaging



Protocol Family Concept

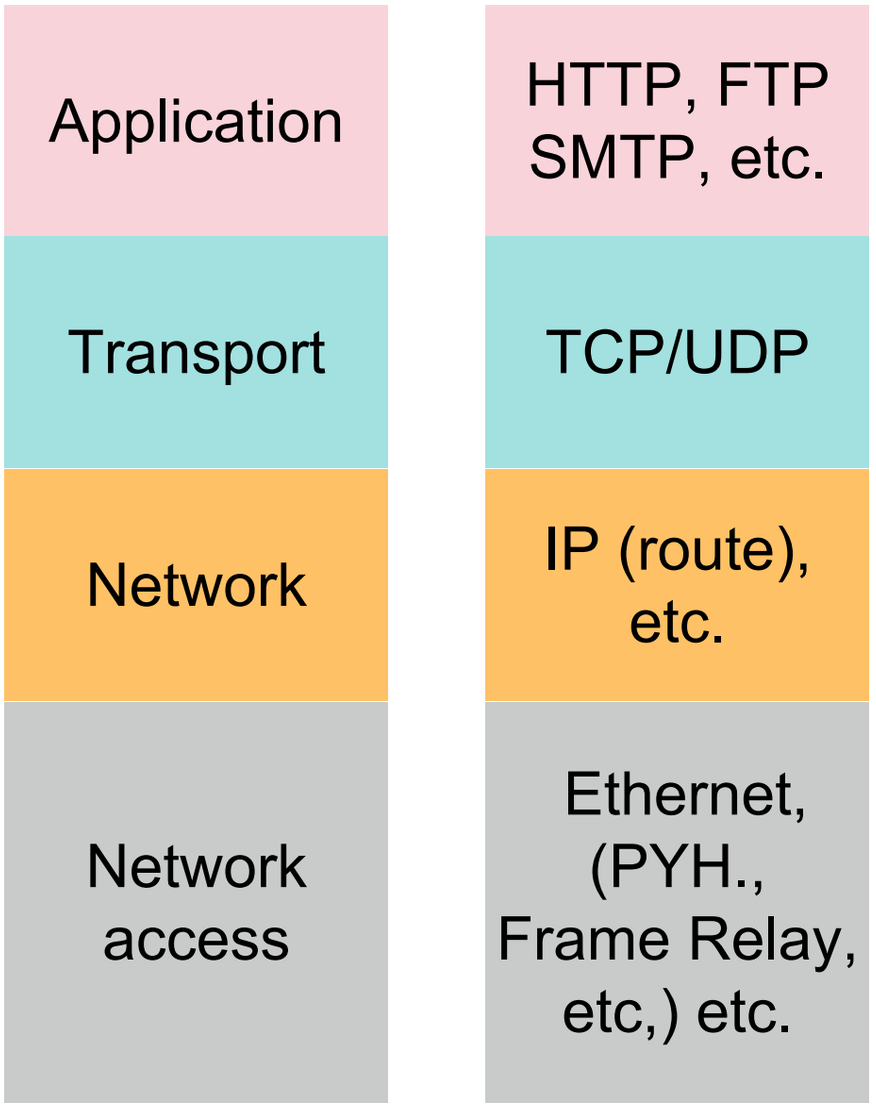
- Each lower level of stack “encapsulates” information from layer above by adding header and trailer.



Most Popular Protocol for Network of Networks

- Transmission Control Protocol/Internet Protocol (TCP/IP) stack

- This protocol family is the basis of the Internet, a WAN (wide area network) protocol
  - IP makes best effort to deliver
  - Packets can be lost, corrupted
- TCP guarantees delivery
- TCP/IP so popular it is used even when communicating locally: even across homogeneous LAN (local area network)
- UDP/IP: video or sound streaming; video call….
- More in CS120



Conclusion

- I/O speed range is 100-million to one
- Polling vs. Interrupts
- DMA to avoid wasting CPU time on data transfers
- Disks for persistent storage, replaced by flash memory
- Networks: computer-to-computer I/O
- Protocol suites allow networking of heterogeneous components. Abstraction!!!