Lecture 27

Review

1. Compare two pipeline implementations options A and B with 4 and 7 stages, respectively.
   - The logic delays of the pipeline stages are follows:

	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Stage 6	Stage 7
Option A	250 ps	180 ps	400 ps	200 ps			
Option B	200 ps	150 ps	250 ps	250 ps	200 ps	150 ps	180 ps

   - What are the maximum clock rates for the two implementations?

Review

- Option A fs_max = _____1/0.4ns = 2.5GHz_____ (include the unit with your result)
- Option B fs_max = _____1/0.25ns = 4GHz_____ (include the unit with your result)

Review

- The table below states the operation of each pipeline stage:

	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Stage 6	Stage 7
Option A	IF/ID	EXE	MEM	WB			
Option B	IF	ID	EXE-1	EXE-2	EXE-3	MEM	WB

- Compared to the MIPS CPU, option A merges IF and ID in a single stage, while option B splits EXE over three pipeline stages. Registers and memory are written to in the first half of the cycle and read during the second half of the cycle (same as MIPS) but there are no forwarding paths. How many instructions are executed after an add and a lw instruction, respectively, before the new register values are available?

Review

	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Stage 6	Stage 7
Option A	IF/ID	EXE	MEM	WB			
Option B	IF	ID	EXE-1	EXE-2	EXE-3	MEM	WB

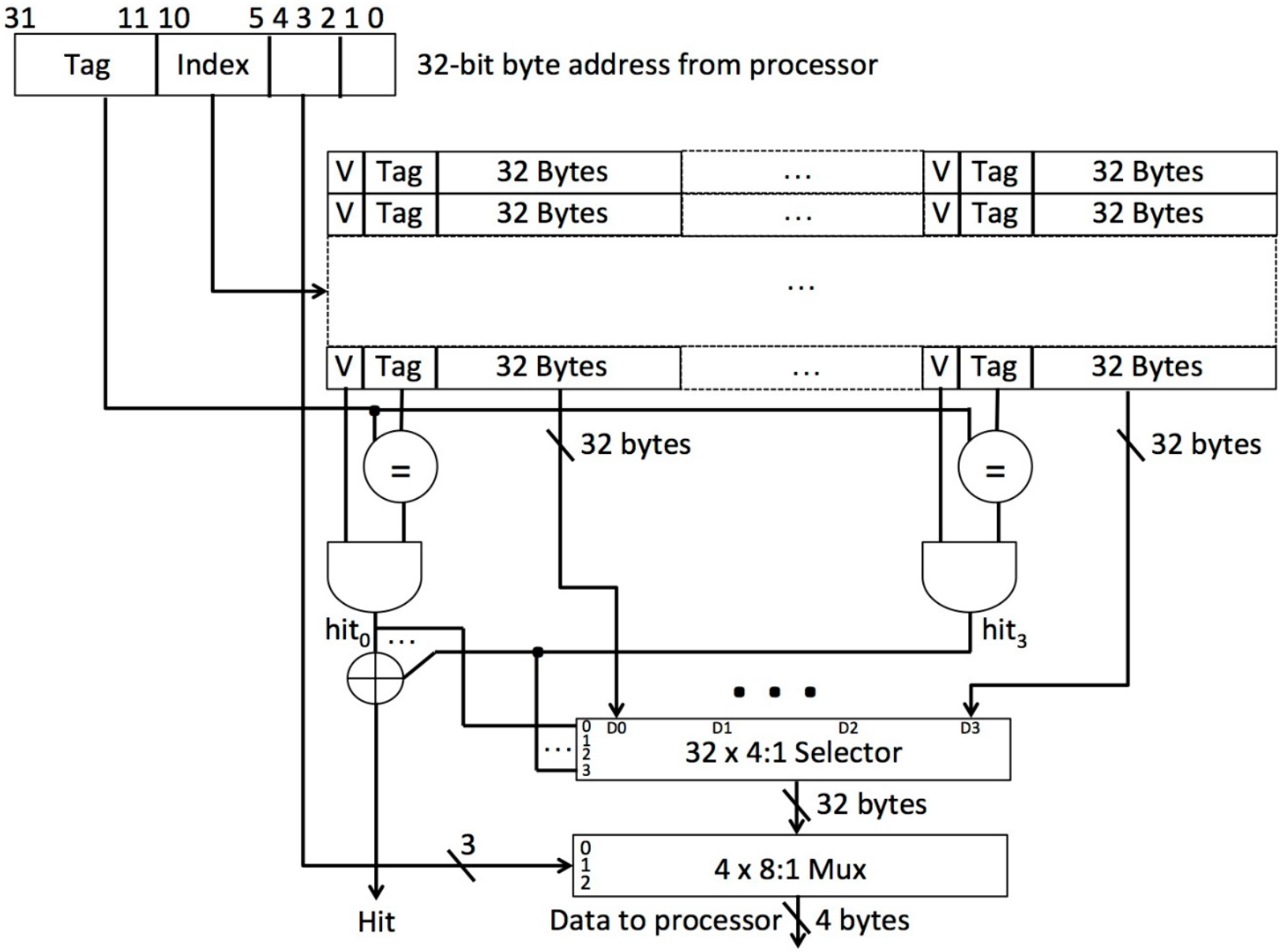
	# nops after add	# nops after lw
Option A	2	2
Option B	4	4

Review

- Calculate the number of instructions executed per second for each implementation for the specified fs (these are not necessarily the correct results for part 1) and CPI.

	f_s	CPI	Instructions per second
Option A	3 GHz	1.5	1/CPI * Cycles / sec $(1/1.5)(3*10^{-9}))= 2*10^{9}$
Option B	5 GHz	2.0	$(1/2.0)(5*10^{-9})) = 2.5*10^{9}$

Review


31
11 10
5 4 3 2 1 0
Tag
Index
32-bit byte address from processor
V
Tag
32 Bytes
...
32 bytes
=
hit0
hit3
D0
D1
D2
D3
32 x 4:1 Selector
3
4 x 8:1 Mux
Hit
Data to processor
4 bytes

Review

- What is the block size of the cache in bytes?

  32 as shown directly from the figure

- What is the number of blocks in this cache?

  We have from bit number 5 to bit number 10 (6 bits) for indexing. The number of rows = 2^6. We also have a multiplexer that shows we have 4 different inputs, one for each block. The number of blocks = 2^6 x 4 = 256 Blocks

- What is the total data capacity of the cache in bytes?

  2^6 x 4 x 32 = 8192 Bytes

- What is the total number of valid bits in the cache?

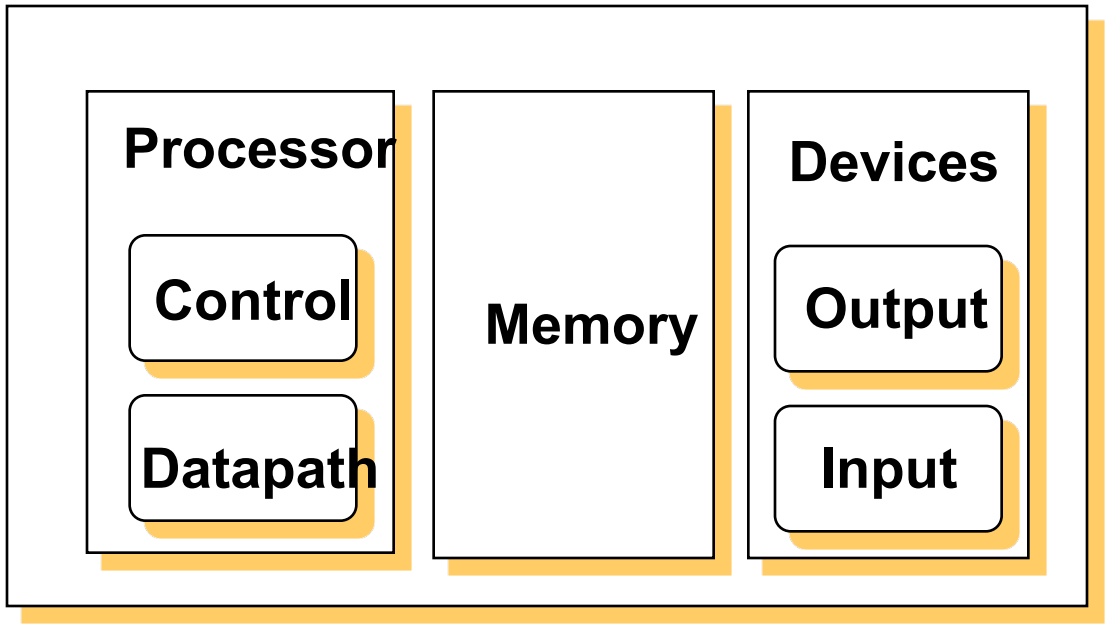
  This is equal to the total number of rows x 4= 2^6 x 4= 256

- What is the total number of tag bits in the cache?

  Tag size x the total number of tags = 21 x 256 = 5376

Input Output Systems

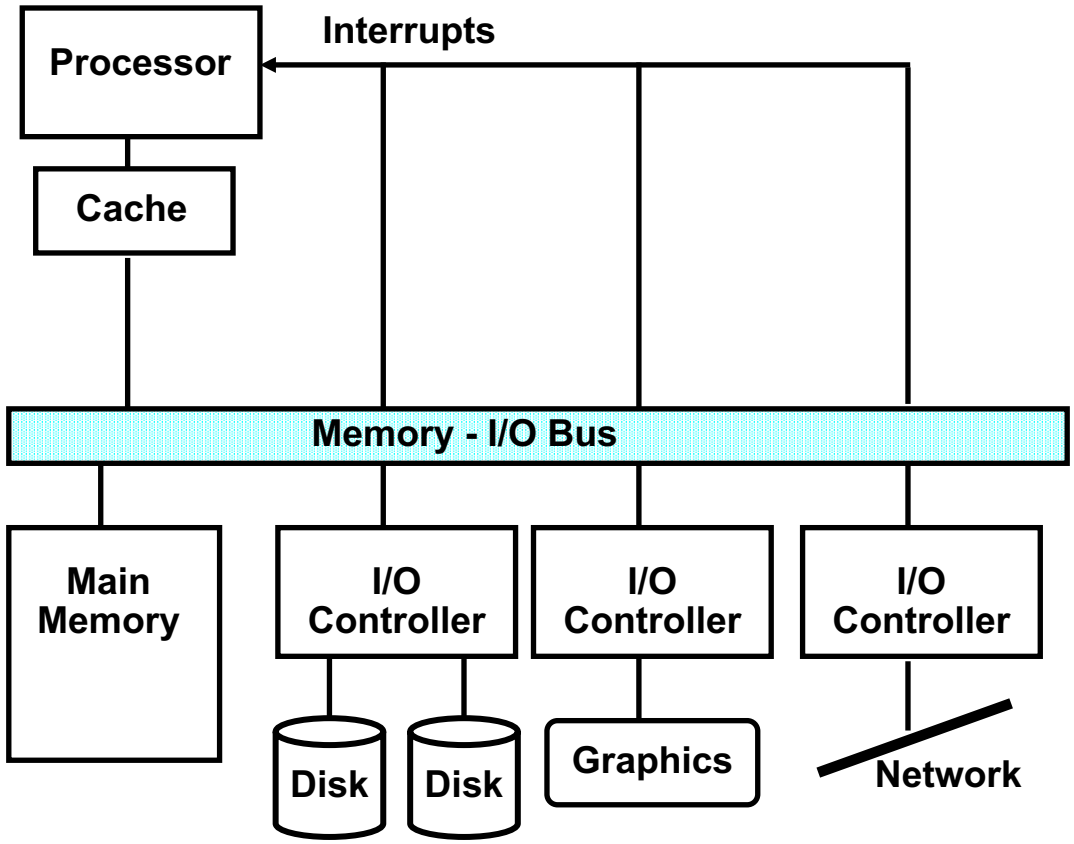
Major Components of a Computer



- Important metrics for an I/O system
  - Performance
  - Compatibility
  - Expandability and diversity
  - Dependability
  - Cost, size, weight

Review: A Typical I/O Interconnect System

- I/O devices can be characterized by
  - Behavior: input, output, storage
  - Partner: human or machine
  - Data rate: bytes/sec, transfers/sec
- Usually have more than one I/O device in the system connected to the processor via a bus
  - each I/O device is managed by an I/O Controller and the OS



Input and Output Devices

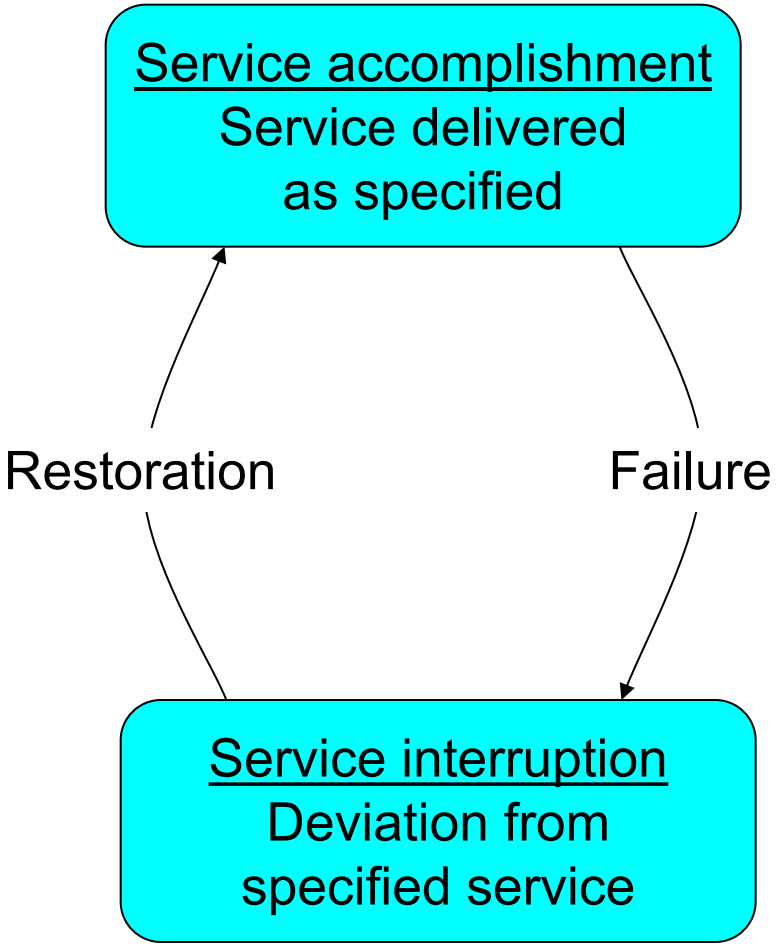
- I/O devices are incredibly diverse with respect to
  - Behavior – input, output or storage
  - Partner – human or machine
  - Data rate – the peak rate at which data can be transferred between the I/O device and the main memory or processor

Device	Behavior	Partner	Data rate (Mb/s)
Keyboard	input	human	0.0001
Mouse	input	human	0.0038
Laser printer	output	human	3.2
Magnetic disk	storage	machine	800 - 3000
Graphics display	output	human	800 - 8000
Network/LAN	input or output	machine	100 - 10000

I/O System Characteristics

- Dependability is important
  - Particularly for storage devices
- Performance measures
  - Latency (response time)
  - Throughput (bandwidth)
- Desktops & embedded systems
    - Mainly interested in response time & diversity of devices
- Servers
    - Mainly interested in throughput & expandability of devices

Dependability



- Fault: failure of a component
  - May or may not lead to system failure

Dependability Measures

- Reliability: mean time to failure (MTTF)
- Service interruption: mean time to repair (MTTR)
- Mean time between failures
  - MTBF = MTTF + MTTR
- Availability = MTTF / (MTTF + MTTR)
- Improving Availability
  - Increase MTTF: fault avoidance, fault tolerance, fault forecasting
  - Reduce MTTR: improved tools and processes for diagnosis and repair

Lecture 28

RAID

- Redundant Array of Inexpensive (Independent) Disks
  - Use multiple smaller disks (one large disk)
  - Parallelism improves performance
  - Plus extra disk(s) for redundant data storage
- Provides fault tolerant storage system
  - Especially if failed disks can be “hot swapped”
- RAID 0
  - No redundancy (“AID”?)
    - Just stripe data over multiple disks
  - But it does improve performance

RAID 1 & 2

- RAID 1: Mirroring
  - N + N disks, replicate data
    - Write data to both data disk and mirror disk
    - On disk failure, read from mirror
- RAID 2: Error correcting code (ECC)
  - N + E disks (e.g., 10 + 4)
  - Split data at bit level across N disks
  - Generate E-bit ECC
  - Too complex, not used in practice

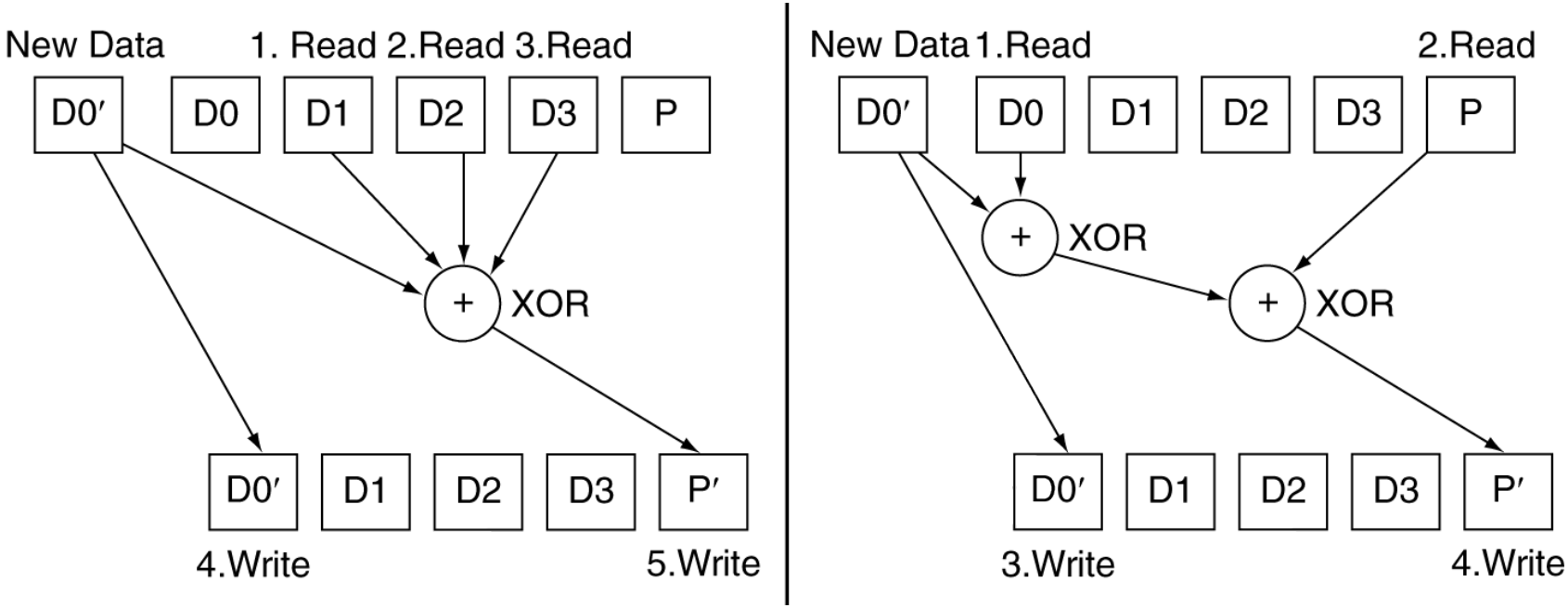
RAID 3: Bit-Interleaved Parity

- N + 1 disks
  - Data striped across N disks at byte level
  - Redundant disk stores parity
  - Read access
    - Read all disks
  - Write access
    - Generate new parity and update all disks
  - On failure
    - Use parity to reconstruct missing data
- Not widely used

RAID 4: Block-Interleaved Parity

- N + 1 disks
  - Data striped across N disks at block level
  - Redundant disk stores parity for a group of blocks
  - Read access
    - Read only the disk holding the required block
  - Write access
    - Just read disk containing modified block, and parity disk
    - Calculate new parity, update data disk and parity disk
  - On failure
    - Use parity to reconstruct missing data
- Not widely used

RAID 3 vs RAID 4

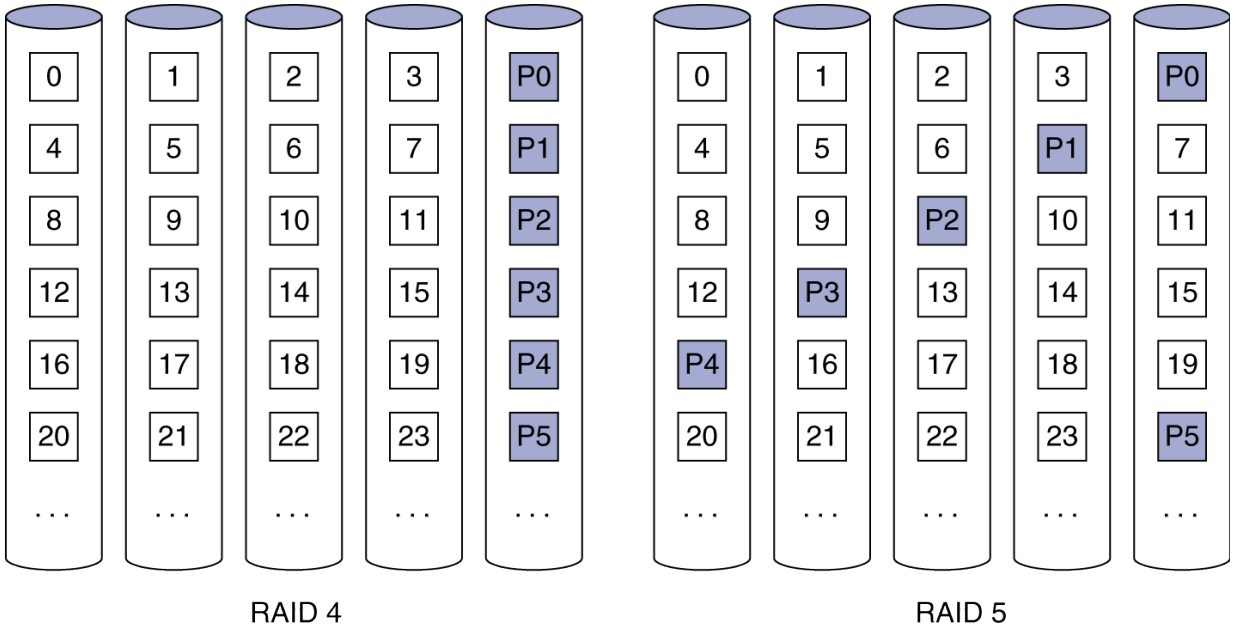


RAID 3

RAID 4

RAID 5: Distributed Parity

- N + 1 disks
  - Like RAID 4, but parity blocks distributed across disks
    - Avoids parity disk being a bottleneck
- Widely used



RAID Summary

- RAID can improve performance and availability
  - High availability requires hot swapping
- Assumes independent disk failures
  - Too bad if the building burns down!
- See “Hard Disk Performance, Quality and Reliability”
  - http://www.pcguide.com/ref/hdd/perf/index.htm

I/O Performance Measures

- I/O bandwidth (throughput) – amount of data that can be input (output) and communicated across an interconnect (e.g., a bus) to the processor/memory (I/O device) per unit time
  1. How much data can we move through the system in a certain time?
  2. How many I/O operations can we do per unit time?

- I/O response time (latency) – the total elapsed time to accomplish an input or output operation
  - An especially important performance metric in real-time systems

- Many applications require *both* high throughput and short response times

I/O System (Bus) Interconnect Issues

- A bus is a shared communication link (a single set of wires used to connect multiple subsystems) that needs to support a range of devices with widely varying latencies and data transfer rates
  - Advantages
    - Versatile – new devices can be added easily and can be moved between computer systems that use the same bus standard
    - Low cost – a single set of wires is shared in multiple ways
  - Disadvantages
    - Creates a communication bottleneck – bus bandwidth limits the maximum I/O throughput

- The maximum bus speed is largely limited by
  - The length of the bus
  - The number of devices on the bus

Types of Buses

- Processor-memory bus ("Front Side Bus", proprietary)
  - Short and high speed
  - Matched to the memory system to maximize the memory-processor bandwidth
  - Optimized for cache block transfers
- I/O bus (industry standard, e.g., SCSI, USB)
  - Usually is lengthy and slower
  - Needs to accommodate a wide range of I/O devices
  - Use either the processor-memory bus or a backplane bus to connect to memory
- Backplane bus (industry standard, e.g., ATA, PCIexpress)
  - Allow processor, memory and I/O devices to coexist on a single bus
  - Used as an intermediary bus connecting I/O busses to the processor-memory bus

I/O Transactions

- An I/O transaction is a sequence of operations over the interconnect that includes a request and may include a response either of which may carry data. A transaction is initiated by a single request and may take *many* individual bus operations. An I/O transaction typically includes two parts,
  1. Sending the address
  2. Receiving or sending the data

- Bus transactions are defined by what they do to memory
  - A read transaction reads data from memory (to either the processor or an I/O device) (output)
  - A write transaction writes data to the memory (from either the processor or an I/O device) (input)

Bus Bandwidth Determinates

- The bandwidth of a bus is determined by
  - Whether it is synchronous or asynchronous and the timing characteristics of the protocol used
  - The bus width (i.e., number of data lines)
  - Whether the bus supports block transfers or only word-at-a-time transfers

	Firewire	USB 2.0
Type	I/O	I/O
Data lines	4	2
Clocking	Asynchronous	Synchronous
Max # devices	63	127
Max length	4.5 meters	5 meters
Peak bandwidth	50 MB/s (400 Mbps)	0.2 MB/s (low) 1.5 MB/s (full) 60 MB/s (high)

Synchronous and Asynchronous Buses

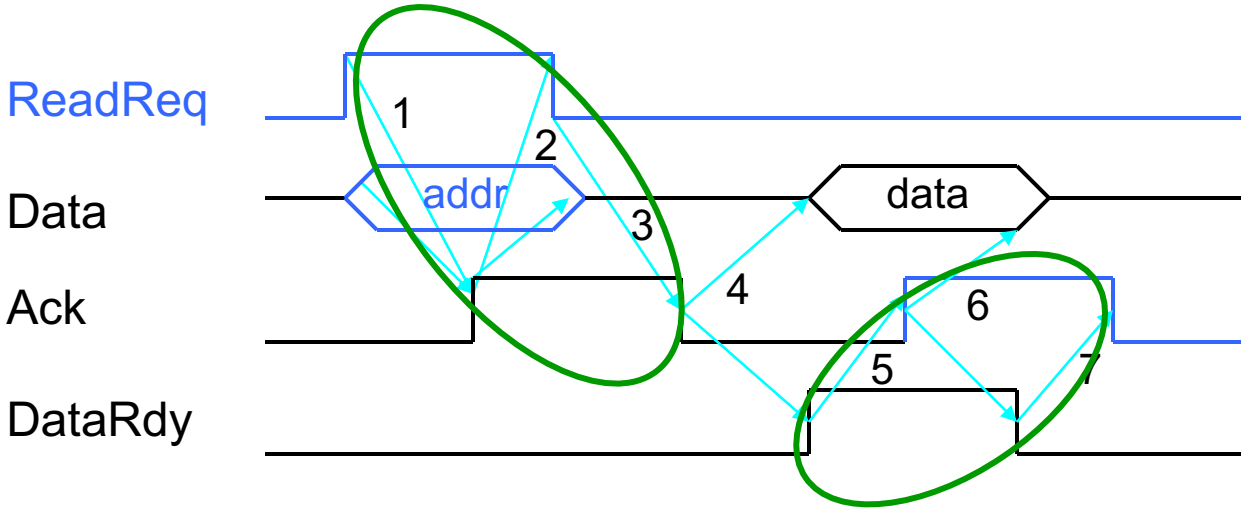
- Synchronous bus (e.g., processor-memory buses)
  - Includes a clock in the control lines and has a fixed protocol for communication that is relative to the clock
  - Advantage: involves very little logic and can run very fast
  - Disadvantages:
    - Every device communicating on the bus must use same clock rate
    - To avoid clock skew, they cannot be long if they are fast
- Asynchronous bus (e.g., I/O buses)
  - It is not clocked, so requires a handshaking protocol and additional control lines (ReadReq, Ack, DataRdy)
  - Advantages:
    - Can accommodate a wide range of devices and device speeds
    - Can be lengthened without worrying about clock skew or synchronization problems
  - Disadvantage: slow(er)

ATA Cable Sizes

- Companies have transitioned from synchronous, parallel wide buses to asynchronous narrow buses
  - Serial ATA cables (red) are much thinner than parallel ATA cables (green)

Asynchronous Bus Handshaking Protocol

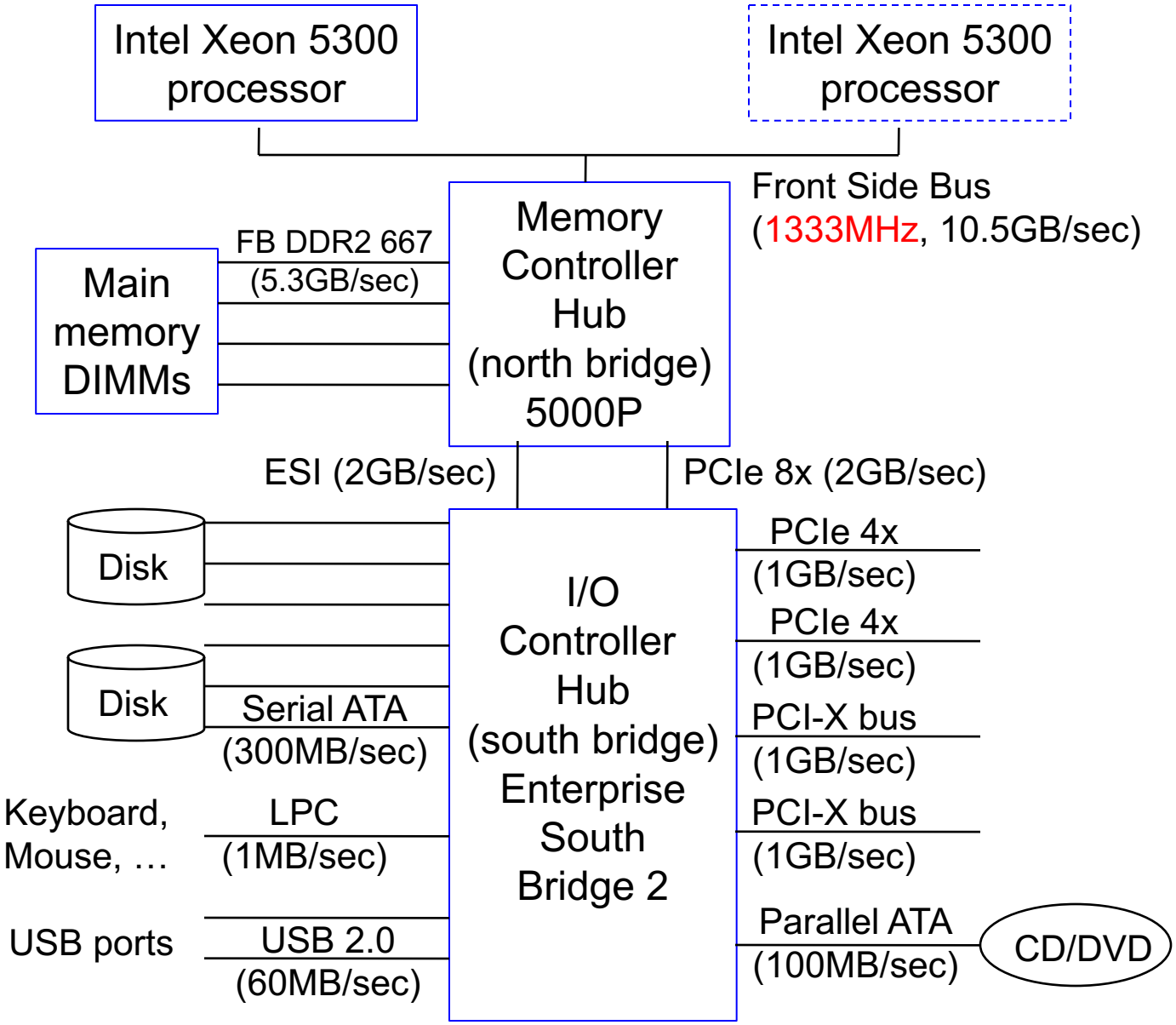
- Output (read) data from memory to an I/O device



I/O device signals a request by raising ReadReq and putting the addr on the data lines

1. Memory sees ReadReq, reads addr from data lines, and raises Ack
2. I/O device sees Ack and releases the ReadReq and data lines
3. Memory sees ReadReq go low and drops Ack
4. When memory has data ready, it places it on data lines and raises DataRdy
5. I/O device sees DataRdy, reads the data from data lines, and raises Ack
6. Memory sees Ack, releases the data lines, and drops DataRdy
7. I/O device sees DataRdy go low and drops Ack

A Typical I/O System



“Real” I/O in SPIM

- SPIM (MIPS processor simulator) supports one memory-mapped I/O device – a terminal with two independent units
  - Transmitter writes characters to the display (console window)
  - Receiver reads characters from the keyboard