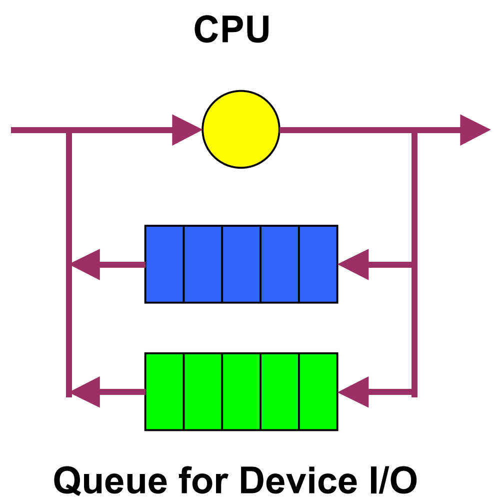
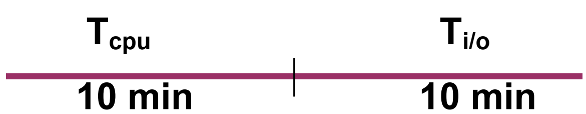
Storage and I/O Devices

What is the point behind building fast CPUs, if the speed of I/O does not improve?



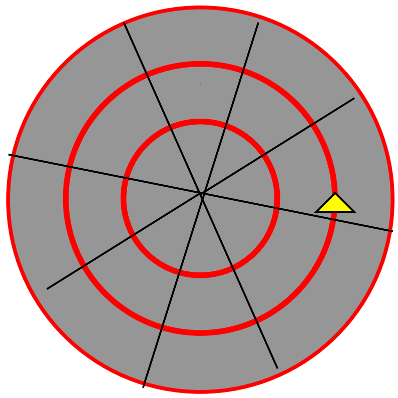
Life of a process



Example

Per **Amdahl Law**, regardless of the CPU speedup, the overall speedup never exceeds 100%!

Disk



Recognize

Tracks, sectors,

Platter/surfaces

Cylinders etc.

Time to read or write from a disk =

Seek time +

Rotational latency +

Controller time +

Queuing delay +

Data transfer time

Disk Characteristics in 2001

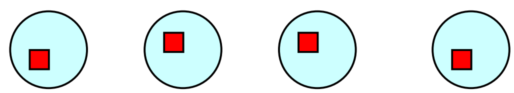
	Seagate Cheetah Ultra160 SCSI	**IBM Travelstar**	**IBM 1 GB Microdrive**
Diameter	3.5”	2.5”	1.0”
Formatted capacity	73.4 GB	32.0 GB	1.0 GB
Cylinders	14,100	21,664	7,167
Disks	12	4	1
Heads	24	8	2
Bytes / sector	512-4096	512	512
Sectors / track	~424	~512	~512
RPM	10,033	5,411	3,600
Seek time *	6.0 ms	12.0 ms	12.0 ms
Data transfer rate	27-40 MB/s	11-21 MB/s	2.6-4.2 MB/s

Continued advance in capacity of 60% per year, and bandwidth of 40% per year.

Disk Array: RAID

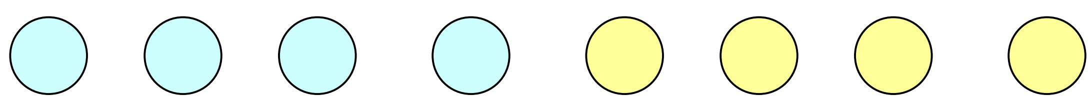
Disk access time and availability can be improved using disk cache and RAID (Redundant Array of Inexpensive Disks). Available at various levels, 0-6.

RAID 0: No redundancy



Striping is used to distribute data over several disks. The access time improves due to the possibility of parallel access.

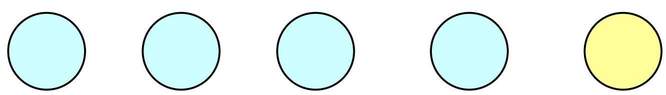
RAID 1: Mirroring



There is 100% redundancy. If one crashes, then data is available from the other

RAID 3: Bit-interleaved parity

Each bit of an N-bit word is written on a separate disk. There is one extra disk (disk N+1), called the parity disk.

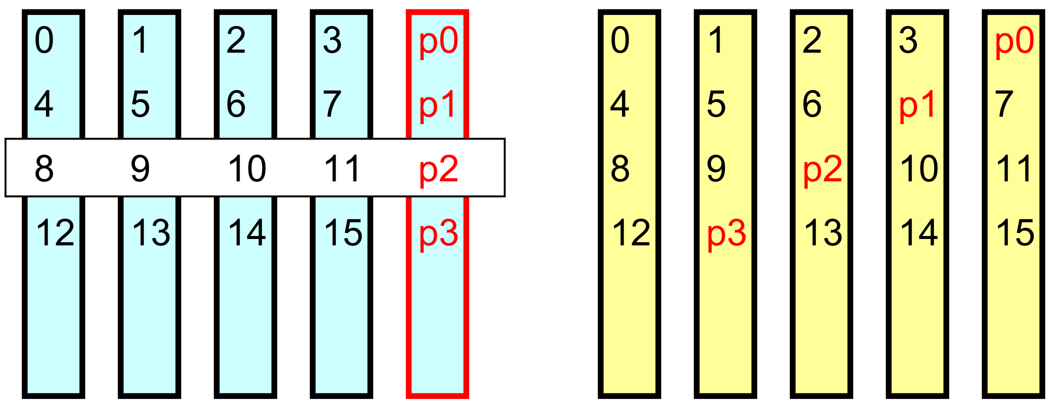


	D0	D1	D2	D3	Parity(even)
Word 0	0	1	1	1	1
Word 1	0	1	0	1	0
Word 2	1	0	1	0	0

If one disk crashes, then data can be reconstructed from the others.

RAID 5: Block Interleaved Distributed Parity

Unlike RAID 4, where all the parity blocks are stored on a separate parity disk, in RAID 5, the parity blocks are distributed. The CRC bits appended to each block handle small read errors



RAID 4 **RAID 5**

Both RAID 4 and RAID 5 allow simultaneous reads, but RAID 5 leads to better parallelism during write operations.

Types of Data Transfer

Programmed I/O

CPU controls everything. CPU polls devices. CPU busy waits when the I/O is slow or not ready. Inefficient, but cheap.

Interrupt Driven I/O

Device interrupts CPU when it needs attention.

Direct Memory Access

Device interface has some intelligence for carrying out routine data transfer under the guidance of the CPU.

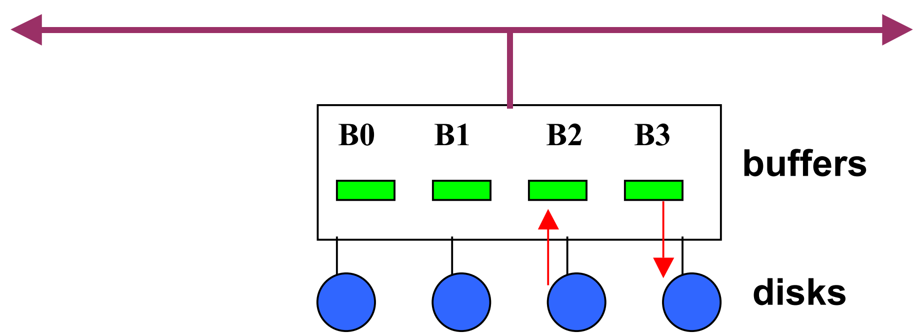
I/O Controllers

A separate special-purpose processor supervises I/O activities

Review these in case you do not remember …

Intelligent Controllers

Intelligent controllers use wider buses and adequate set of buffers to relax the timing constraints and improve efficiency. For example SCSI can support 16 devices on a single bus.



The smartness is due to its ability to handle

- Overlapped commands
- Re-order commands (command queuing)
- Scatter-gather, which provides multiple host addresses in one command.

Example of overlapped command processing

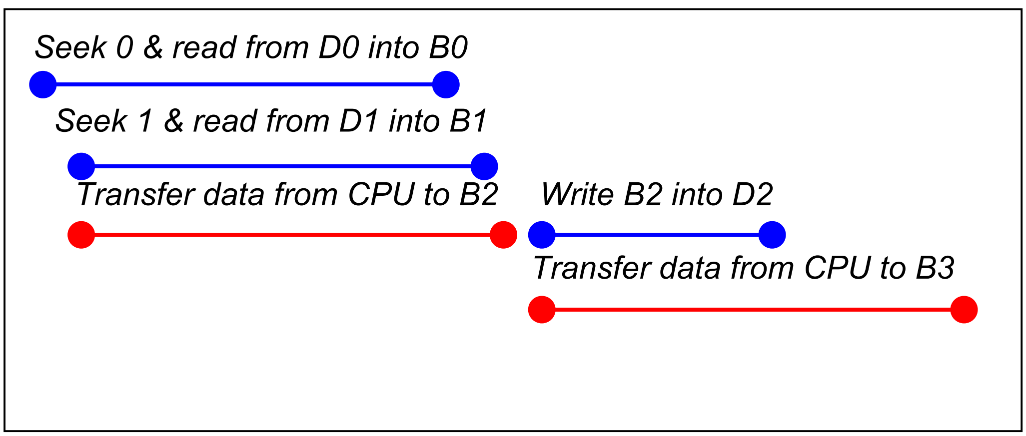
Consider the following queue of requests:

Read Disk 0

Read Disk 1

Write Disk 2

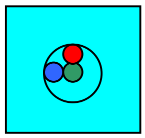
Write Disk 3



A Taxonomy of SCSI

Name	Bits	Transfer Rate
SCSI-1	8	5 MB/sec
Fast SCSI-1	8	10 MB/sec
Ultra SCSI-1	8	20 MB/sec
Wide Ultra SCSI-1	16	40 MB/sec
Ultra SCSI-2	8	40 MB/sec
Wide Ultra SCSI-2	16	80 MB/sec
Ultra SCSI-3	16	160 MB/sec

Graphics Terminals



Primitive BW: 1 bit per pixel

Gray scale: 8-bits per pixel

RGB true color = Red (8-bit) + Green (8 bit) + Blue (8-bit)

needs 24-bits. Often associated with an 8-bit A-component

for special effects.

CMYK true color (Cyan, Magenta, Yellow, Black): 32 bits.

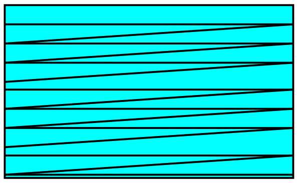
Memory Requirement

A frame buffer stores the image frame(s).

Small BW display 512 x 340 x 1 bit □ 20 KB

Large color terminal 1280 x 1024 x 24 bits □ 4 MB

Displaying a Picture



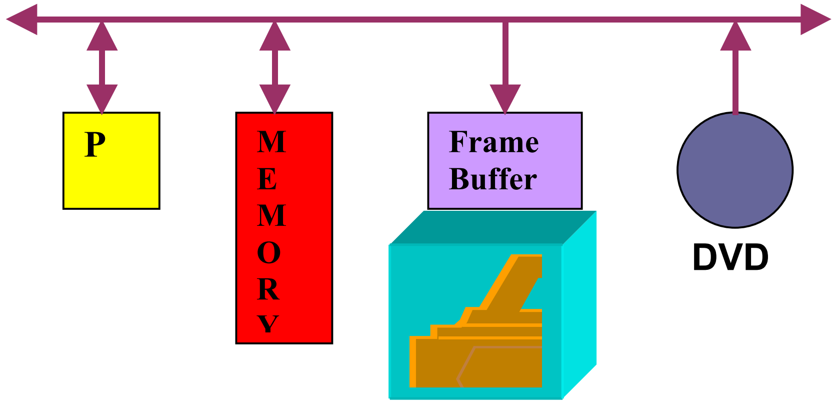
Refresh Rate = 60 frames /second

Memory bandwidth for still pictures

= 60 x 4 = 240 MB/sec

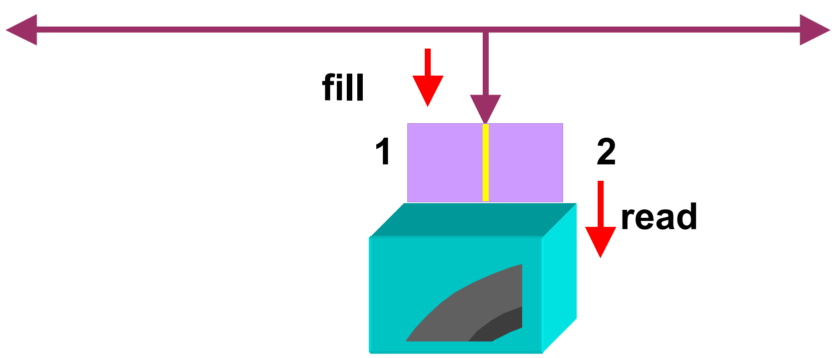
Good quality animation needs at least 30 frames/sec. So the memory has to be written @ 30 x 4 = 120 MB sec

Animation



Two frame buffers improve the quality of animation.

When one buffer fills, data is read from the other buffer and vice versa.



This doubles the memory requirement. Memory requirement can be reduced using color map table.

<u>Color Map table</u>

Index	RED	BLUE	GREEN
0	**0000001**	**01011010**	**11001100**
1	**00001111**	**10011001**	**11111111**
2	**00011111**	**10101011**	**11010101**
255	**11111111**	**11001110**	**11011101**

Each shade is a blend of three basic colors. Let us define 256 distinct shades. You can now specify a shade by the 8-bit index to the color map table , instead of the 24-bit pattern! This is called indexed color, used in GIF.

You can define < 255 shades and save more memory, but the quality of the image will suffer.

Calculate the savings in buffer size and bandwidth.

<u>BUS</u>

Connects processors with memory and peripherals.

Synchronous Bus

Transactions synchronized with clocks.

Faster, but works on buses of short length.

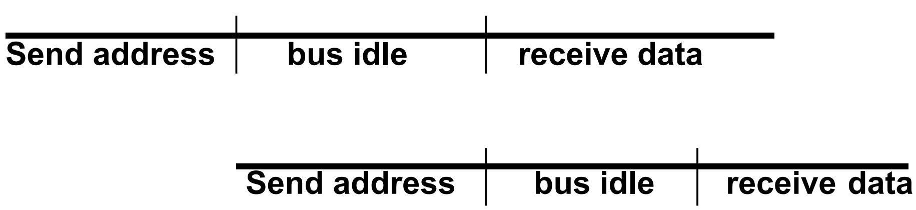
Asynchronous Bus

Slower than synchronous bus. Clocks are not used.

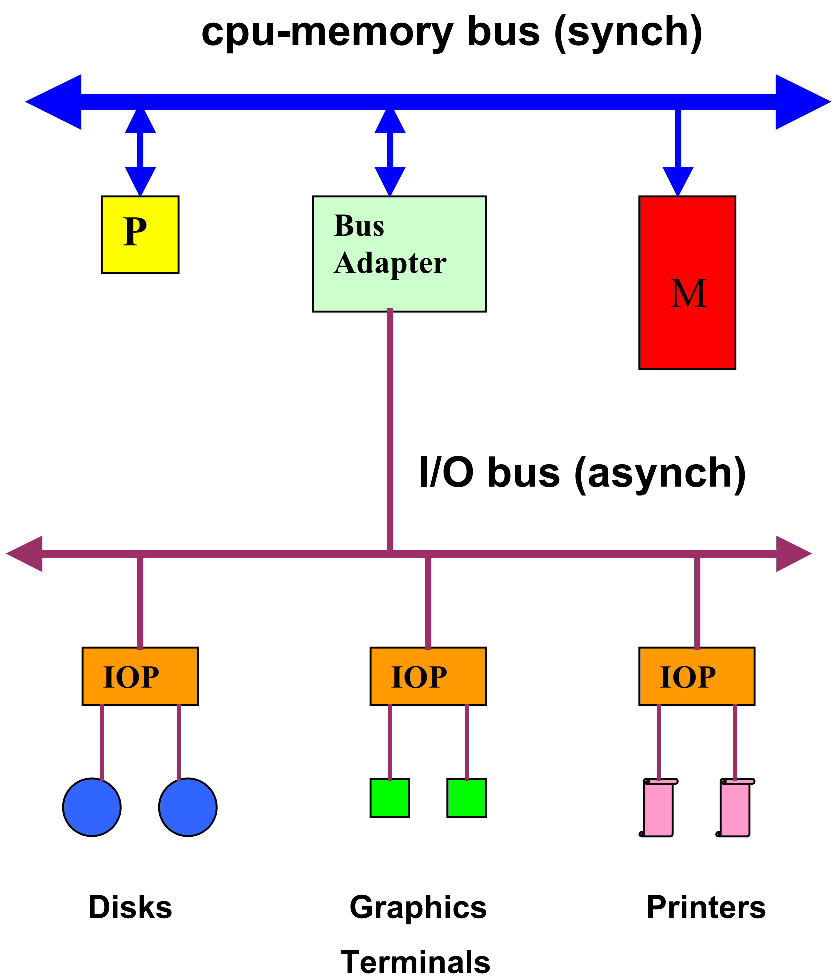
Transactions are self-timed and use handshaking protocols.

Split-transaction Bus

Also called *packet-switched* or *pipelined* bus



Maximizes bus utilization.


cpu-memory bus (synch)
P
Bus
Adapter
M
I/O bus (asynch)
IOP
IOP
IOP
Disks
Graphics
Terminals
Printers