MEMORY

and Basic CPU Cleanup
<table>
<thead>
<tr>
<th>Signal</th>
<th>Description</th>
<th>Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>alu</td>
<td>ALU function select</td>
<td>2-bits</td>
</tr>
<tr>
<td>rd</td>
<td>register bank destination</td>
<td>2-bits</td>
</tr>
<tr>
<td>rs</td>
<td>register bank source</td>
<td>2-bits</td>
</tr>
<tr>
<td>rw</td>
<td>register bank write</td>
<td>1-bit</td>
</tr>
<tr>
<td>rr</td>
<td>register bank read</td>
<td>1-bit</td>
</tr>
<tr>
<td>aw</td>
<td>register A write</td>
<td>1-bit</td>
</tr>
<tr>
<td>cw</td>
<td>register C write</td>
<td>1-bit</td>
</tr>
<tr>
<td>cr</td>
<td>register C read</td>
<td>1-bit</td>
</tr>
<tr>
<td>pcr</td>
<td>program counter read</td>
<td>1-bit</td>
</tr>
<tr>
<td>pcw</td>
<td>program counter write</td>
<td>1-bit</td>
</tr>
<tr>
<td>maw</td>
<td>memory address write</td>
<td>1-bit</td>
</tr>
<tr>
<td>mdr</td>
<td>memory data read</td>
<td>1-bit</td>
</tr>
<tr>
<td>mdw</td>
<td>memory data write</td>
<td>1-bit</td>
</tr>
<tr>
<td>irw</td>
<td>instruction register write</td>
<td>1-bit</td>
</tr>
</tbody>
</table>
This is the register transfer notation for a register to register add, where src and dest codes are used to specify the register numbers. These codes would come from the Instruction Register.

Note, time step 1 is unusual, because MD is being loaded directly from memory and does not use the signals controlling the data path.
This is the register transfer notation for a jump using a register to store the next address. The register to be used is stored in the Instruction Register.

Note, time step 1 is unusual, because MD is being loaded directly from memory and does not use the signals controlling the data path.
MEMORY ORGANIZATION

Processor package

Core 0
- Regs
- L1 d-cache
- L1 i-cache
- L2 unified cache
- L3 unified cache (shared by all cores)

Core 3
- Regs
- L1 d-cache
- L1 i-cache
- L2 unified cache

...
BASIC MEMORY TYPES

ROM - Read Only Memory
RAM - Random Access Memory
SRAM - Static Random Access Memory
DRAM - Dynamic Random Access Memory
SPEED DIFFERENCES

![Graph showing read throughput (MB/s) vs working set size (bytes) for different cache regions: Main memory region, L3 cache region, L2 cache region, and L1 cache region. The graph illustrates the speed differences and how they change with varying working set sizes.]
MEMORY MODULE

Diagram showing a memory controller connected to a DRAM chip. The DRAM chip contains a grid with rows and columns, and a Supercell is marked at position (2,1). The memory controller communicates with the chip using address and data signals.
MEMORY MODULE (FUNCTION)

Memory controller

CAS = 1
addr
data

Supercell (2,1)

Rows
0 1 2 3

Cols
0 1 2 3

Internal row buffer

Memory controller

CAS = 1
addr
data

Supercell (2,1)

Rows
0 1 2 3

Cols
0 1 2 3

Internal row buffer
Supercell \((i, j)\)!

31

8

15

16

23

24

32

63

39

40

47

48

55

56

64-bit word at main memory address \(A\)

64 MB memory module consisting of eight 8M x 8 DRAMs

Memory controller

64-bit word to CPU chip
The disk surface spins at a fixed rotational rate.

The read/write head is attached to the end of the arm and flies over the disk surface on a thin cushion of air.

By moving radially, the arm can position the read/write head over any track.
DISK ORGANIZATION

- Tracks
- Surface
- Spindle
- Track $k$
- Gaps
- Sectors
- Cylinder $k$
- Surface 0
- Surface 1
- Surface 2
- Surface 3
- Surface 4
- Surface 5
- Spindle
- Platter 0
- Platter 1
- Platter 2
Requests to read and write logical disk blocks
**BASIC ARCHITECTURE**

Diagram showing the basic architecture of a computer system, including:
- CPU with ALU and Register file
- Bus interface
- System bus
- Memory bus
- Main memory
- I/O bridge
- I/O bus
- USB controller
- Graphics adapter
- Host bus adapter (SCSI/SATA)
- Disk controller
- Disk drive
- Expansion slots for other devices such as network adapters.

Device connections include:
- Mouse
- Solid state disk
- Keyboard
- Monitor
- Monitor
- Monitor
- Monitor
- Disk drive
MEMORY PYRAMID

- **L0:** CPU registers hold words retrieved from cache memory.
- **L1:** L1 cache holds cache lines retrieved from the L2 cache.
- **L2:** L2 cache holds cache lines retrieved from the L3 cache.
- **L3:** L3 cache holds cache lines retrieved from memory.
- **L4:** Main memory holds disk blocks retrieved from local disks.
- **L5:** Local disks hold files retrieved from disks on remote network servers.
- **L6:** Remote secondary storage (distributed file systems, Web servers).

**Storage Devices:**

- **Larger, slower, and cheaper (per byte):** Local secondary storage (local disks)
- **Smaller, faster, and costlier (per byte):** Remote secondary storage (distributed file systems, Web servers)