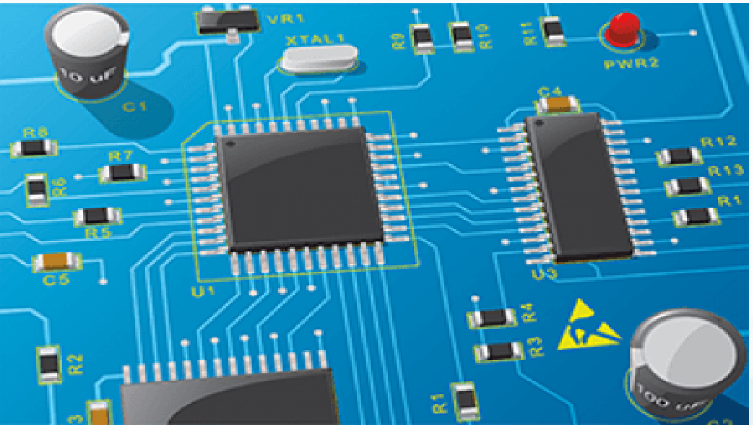
Bus Interconnection & Internal Memory

National College of Ireland

Dublin, Ireland.



Edited by Dr Muhammad Iqbal

Peripherals

- **A peripheral is a "device that is used to deliver/ receive information to/ from the computer" and the process is known as input–output (I/O).**

- **Input Devices: which interact with or send data to the computer (mouse, keyboards, etc.)**

- **Output Devices: which provide output to the user from the computer (monitors, printers, etc.).**

- **Touch Screen Devices: combine different devices into a single hardware component that can be used both as an input and output device.**

Introduction

- The data transfer rate of peripherals is much slower than that of the memory or processor.
- This means that the use the high-speed system bus is not suitable to communicate directly with a peripheral.
- On the other hand, the data transfer rate of some peripherals, such as disk drives is faster than that of the memory.
- Mismatch would lead to inefficiencies if not managed properly.
- Peripherals use different data formats and word lengths than the computer to which they are attached. Thus, an I/O module is required.

Generic Model of an I/O Module

This module has two major functions as shown in Figure 7.1.

- Interface to the processor and memory via the system bus or central switch.

- Interface to one or more peripheral devices by tailored data links.

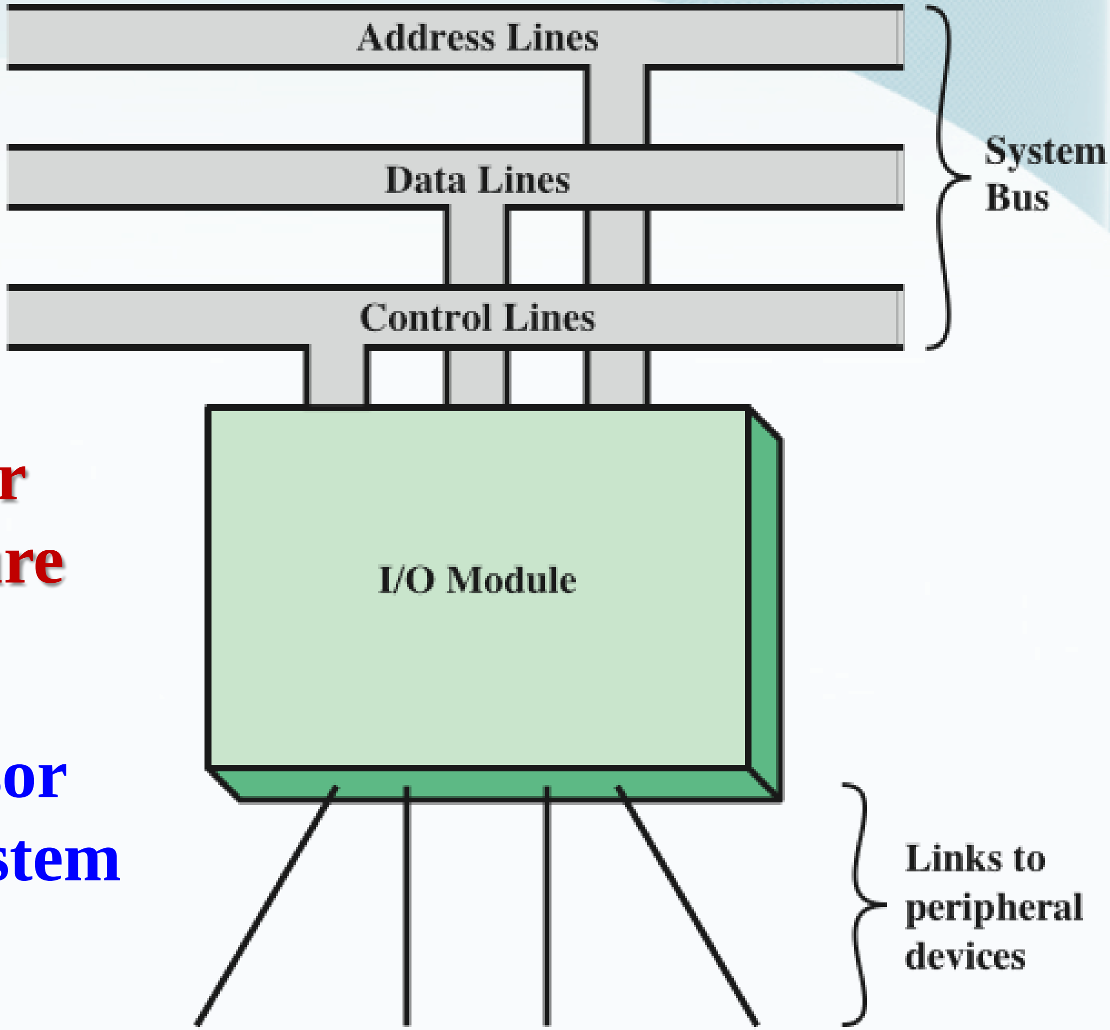


Figure 7.1 Generic Model of an I/O Module

I/O Function

I/O module (disk controller) can exchange data directly with the processor.

Processor can read data from or write data to an I/O module

- **Processor identifies a specific device that is controlled by a particular I/O module**
- **I/O instructions rather than memory referencing instructions**

I/O Function

In some cases it is desirable to allow I/O exchanges to occur directly with memory

- **The processor grants to an I/O module the authority to read from or write to memory so that the I/O memory transfer can occur without tying up the processor**
- **The I/O module issues read or write commands to memory relieving the processor of responsibility for the exchange**
- **This operation is known as direct memory access (DMA)**

Major functions of I/O Module

i. **Control and timing**

ii. **Processor communication**

iii. **Device communication**

iv. **Data buffering**

v. **Error detection.**

Computer Modules

A computer consists of a set of components or modules of three basic types (processor, memory, I/O) that communicate with each other.

- Actually, a computer is a network of basic modules.
- The collection of paths connecting the various modules is called the interconnection structure.

Memory: A memory module will consist of N words of equal length. Each word is assigned a unique numerical address (0, 1, …, N - 1).

- A word of data can be read from or written into the memory. The nature of the operation is indicated by read and write control signals.
- The location for the operation is specified by an address.

Computer Modules

I/O module: I/O is functionally similar to memory. There are two operations, *read and write*.

- I/O module may control more than one external device.
- We can refer to each of the interfaces to an external device as a port and give each a unique address (e.g., 0, 1, …, M - 1).
- There are external data paths for the input and output of data with an external device.
- Finally, an I/O module may be able to send interrupt signals to the processor.

Processor:

The processor reads in instructions and data, writes out data after processing, and uses control signals to control the overall operation of the system.

- It also receives interrupt signals.

The interconnection structure must support the following types of transfers:

Memory to Processor

Processor reads an instruction or a unit of data from memory

Processor to Memory

Processor writes a unit of data to memory

I/O to Processor

Processor reads data from an I/O device via an I/O module

Processor to I/O

Processor sends data to the I/O device

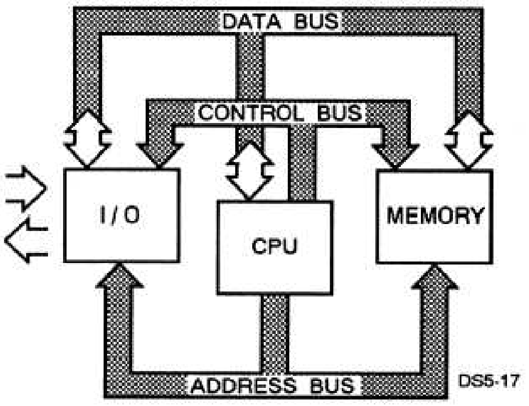
I/O to or from Memory

An I/O module is allowed to exchange data directly with memory without going through the processor using direct memory access

Bus Interconnection

- **A communication pathway connecting two or more devices.**
- **Signals transmitted by any one device are available for reception by all other devices attached to the bus.**
- **Two devices transmit during the same time period their signals will overlap and become garbled.**
- **Each line is capable of transmitting signals representing binary 1 and binary 0.**
- **System bus: A bus that connects major computer components (processor, memory, I/O).**

Data Bus



- **Data lines that provide a path for moving data among system modules.**

- **May consist of 32, 64, 128, or more separate lines.**

- **The number of lines is referred to as the *width* of the data bus.**

- **The width of the data bus is a key factor in determining overall system performance.**

- **For example, if the data bus is 32 bits wide and each instruction is 64 bits long, then the processor must access the memory module twice during each instruction cycle.**

Address Bus

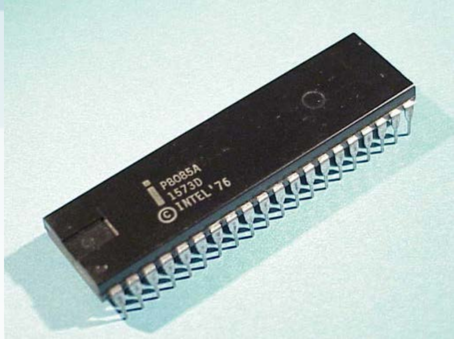
- The address lines are used to designate the source or destination of the data on the data bus
  - If the processor wishes to read a word (8, 16, 32 bits) of data from memory, it puts the address of the desired word on the address lines.
- Width determines the maximum possible memory capacity of the system.
- Also used to address I/O ports
  - The higher order bits are used to select a particular module on the bus and the lower order bits select a memory location or I/O port within the module.

Control Bus

- This is a dedicated bus, because all timing signals are generated according to control signal.
- The control lines used to control the access and the use of the data and address lines.
- Because the data and address lines are shared by all components there must be a means of controlling their use.
- Control signals transmit both command and timing information among system modules.
- Command signals specify operations to be performed.

Data Bus

Address Bus



Intel 8085 microprocessor

Data bus is bidirectional, while address bus is unidirectional. That means data travels in both directions but the addresses will travel in only one direction. The reason for this is that unlike the data, the address is always specified by the processor.

Example: Address bus is uni-directional. In Intel 8085 microprocessor, Address bus was of 16 bits. This means that Microprocessor 8085 can transfer maximum 16 bit address which means it can address $2^{16} = 65{,}536$ different memory locations.

Bus Interconnection Scheme

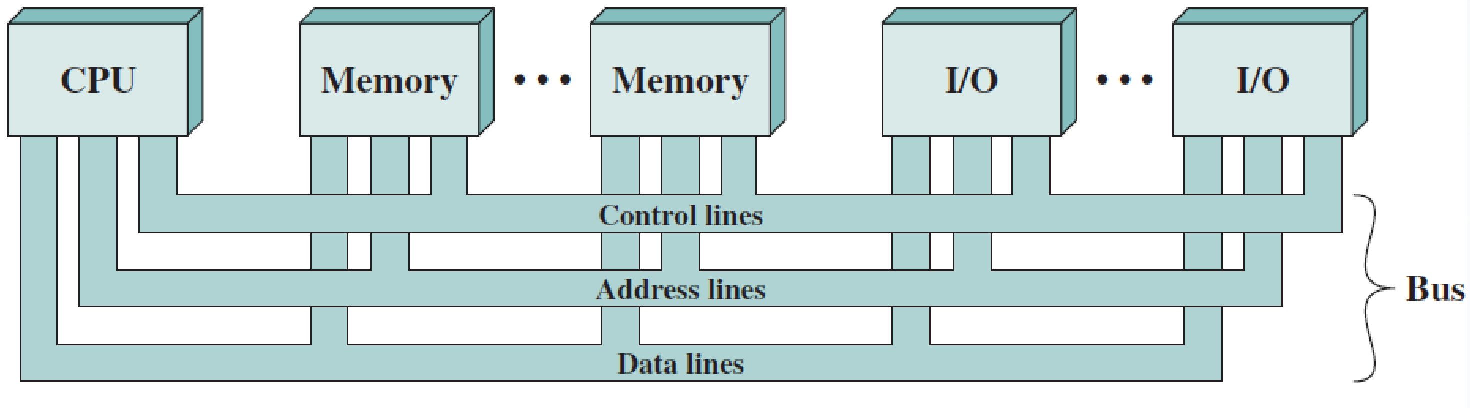


Figure 3.16 Bus Interconnection Scheme

The operation of the bus is as follows. If one module wishes to send data to another, it must do two things:

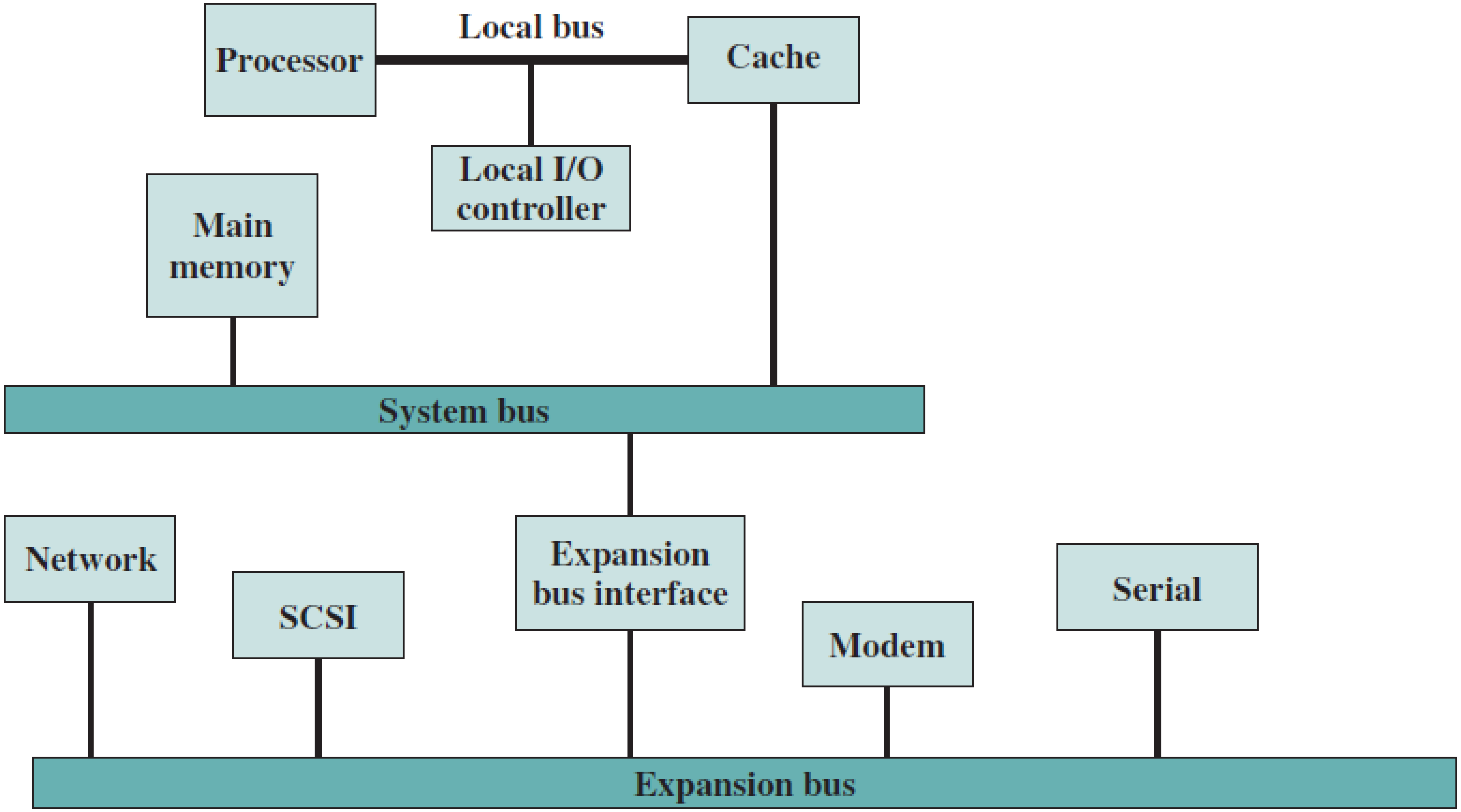
1) **obtain the use of the bus**
2) **transfer data via the bus**

- **If one module wishes to request data from another module, it must**

1) **obtain the use of the bus**
2) **transfer a request to the other module over the appropriate control and address lines.**

- **It must then wait for that second module to send the data.**

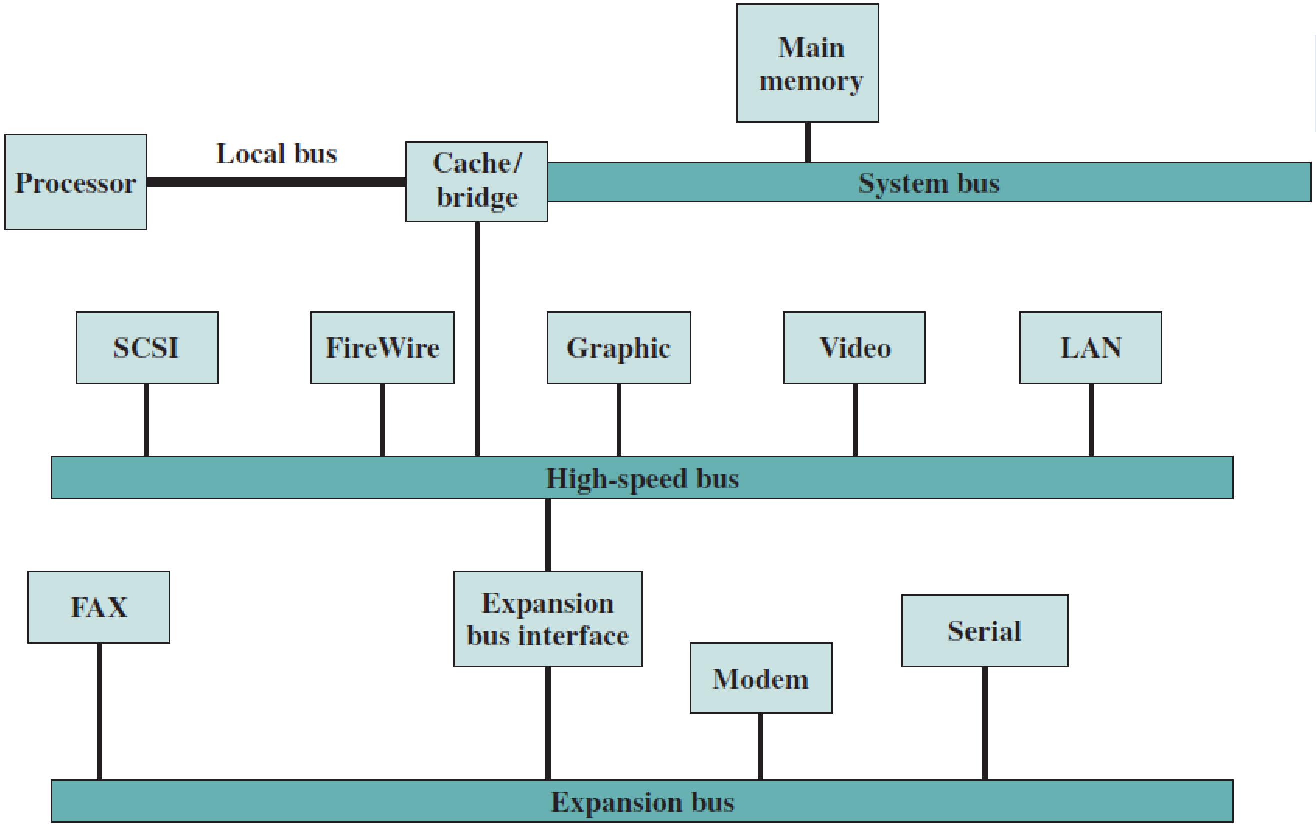
Bus Configurations



(a) Traditional Bus Architecture

- **Most bus-based computer systems use multiple buses, generally laid out in a hierarchy.**
- **There is a local bus that connects the processor to a cache memory and that may support one or more local devices.**
- **The cache memory controller connects the cache not only to this local bus, but to a system bus to which are attached all of the main memory modules.**

Bus Configuration



(b) High Performance Architecture

The Small Computer System Interface (SCSI) is a set of parallel interface standards developed by the American National Standards Institute (ANSI) for attaching printers, disk drives, scanners and other peripherals to computers.

- Local bus that connects the processor to a cache controller, which is in turn connected to a system bus that supports main memory.
- The cache controller is integrated into a bridge, or buffering device, that connects to the high-speed bus.
- This bus supports connections to high-speed LANs, such as Fast Ethernet at 100 Mbps, video and graphics workstation controllers, as well as interface controllers to local peripheral buses, including SCSI and FireWire.

Advantages of Multiple Bus Architecture

With multiple buses, there are fewer devices per bus.

1. **Improve the speed and enhance the performance of processor in execution of different instructions.**
2. **Using multiple-bus architecture will make each device to connect to own bus, which means that each device will have its own bus.**
3. **This way, it will be faster to transfer data of each devices, so the data transfer doesn't have to stuck like in the single-bus architecture where many devices are connected to a single-bus, that will eventually reach the capacity of the bus and thus will make the data "queue".**
4. **The cost of multiple bus architecture is higher, but the cost will not match the need of faster speed, compared to the one of that single-bus architecture.**
5. **This reduces propagation delay, because each bus can be shorter.**
6. **Thus the bottleneck effects are reduced with multiple bus architecture.**

Point-to-Point Interconnect

- **The *shared bus architecture* was the standard approach to interconnection between the processor and other components (memory, I/O, and so on).**

- **But contemporary systems increasingly rely on point-to-point interconnection rather than shared buses.**

Semiconductor Main Memory

- **The basic element of a semiconductor memory is the memory cell.**
- **The memory cell is the fundamental building block of computer memory.**
- **The memory cell is an electronic circuit that stores one bit of binary information and it must be set to store a logic 1 (high voltage level) and reset to store a logic 0 (low voltage level).**
- **Its value is maintained/stored until it is changed by the set/reset process.**
- **The value in the memory cell can be accessed by reading it.**

Memory Cell Operation

- **The figure shows the operation of a memory cell.**
- **Commonly, the cell has three functional terminals capable of carrying an electrical signal.**

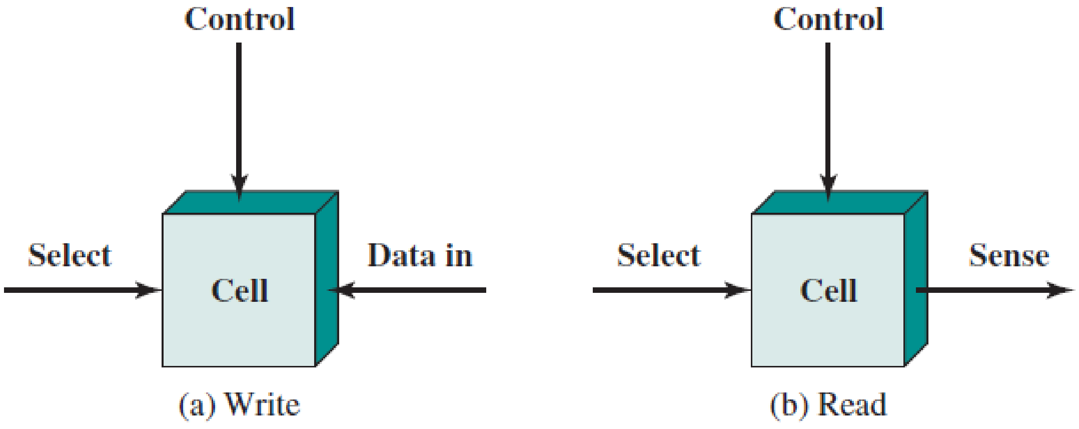


Figure 5.1 Memory Cell Operation

- **The select terminal, as the name suggests, selects a memory cell for a read or write operation.**
- **The control terminal indicates read or write.**
- **For writing, the other terminal provides an electrical signal that sets the state of the cell to 1 or 0.**
- **For reading, that terminal is used for output of the cell's state.**

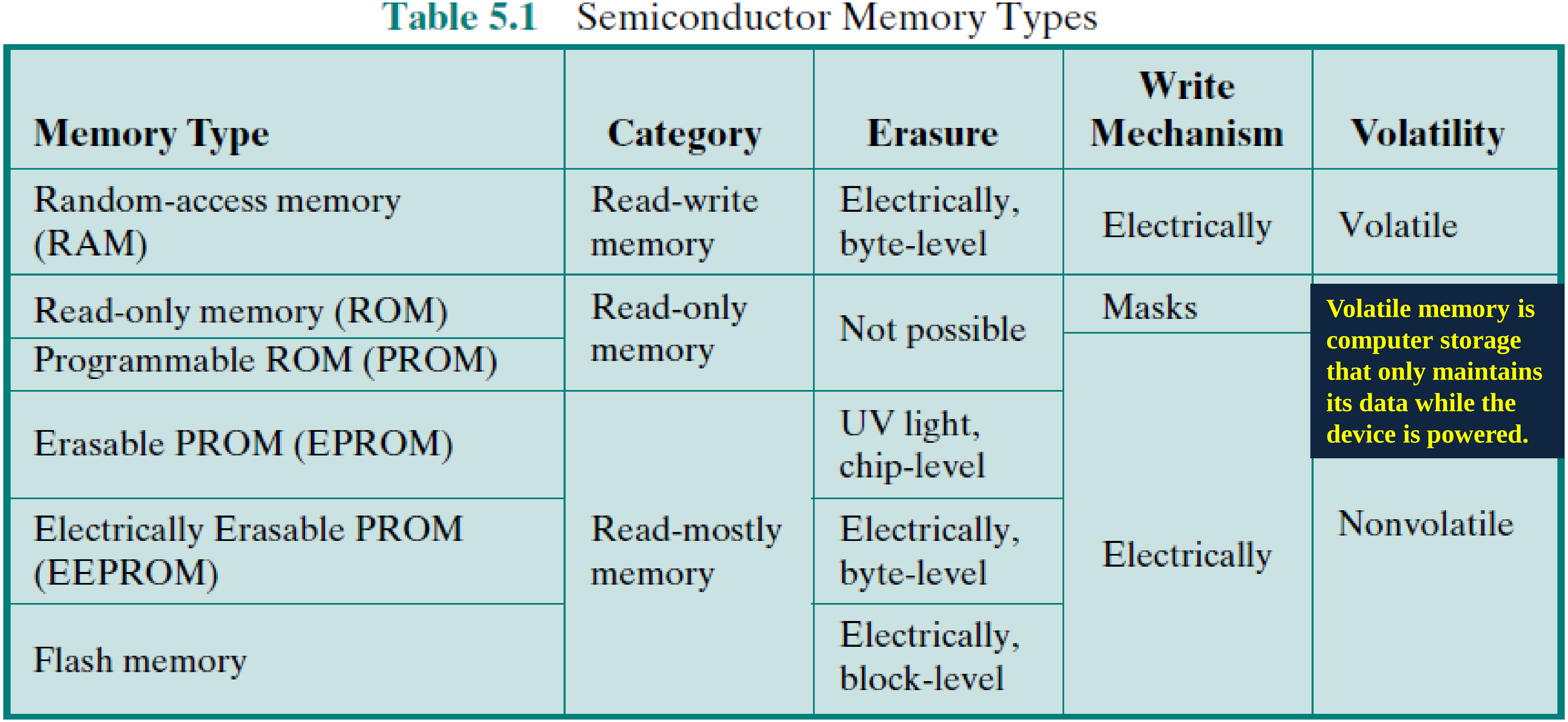


Table 5.1 Semiconductor Memory Types

Memory Type	Category	Erasure	Write Mechanism	Volatility
Random-access memory (RAM)	Read-write memory	Electrically, byte-level	Electrically	Volatile
Read-only memory (ROM)	Read-only memory	Not possible	Masks	Nonvolatile
Programmable ROM (PROM)			Electrically	
Erasable PROM (EPROM)	Read-mostly memory	UV light, chip-level		
Electrically Erasable PROM (EEPROM)		Electrically, byte-level		
Flash memory		Electrically, block-level		

Volatile memory is computer storage that only maintains its data while the device is powered.

- One distinguishing characteristic of memory that is designated as ***RAM*** is that it is possible both ***to read data*** from the memory and ***to write new data*** into the memory easily and rapidly.

- Both the reading and writing are accomplished through the use of electrical signals.

Dynamic RAM (DRAM)

- **RAM technology is divided into two technologies:**
  - **Dynamic RAM (DRAM)**
  - **Static RAM (SRAM)**

– **DRAM**

- **Made with cells that stores each bit of data as a charge in a separate capacitor within an integrated circuit.**
- **Presence or absence of charge in a capacitor is interpreted as a binary 1 or 0.**
- **Requires periodic charge refreshing to maintain data storage.**
- **The term *dynamic* refers to tendency of the stored charge to leak away, even with power continuously applied.**

Dynamic RAM Structure

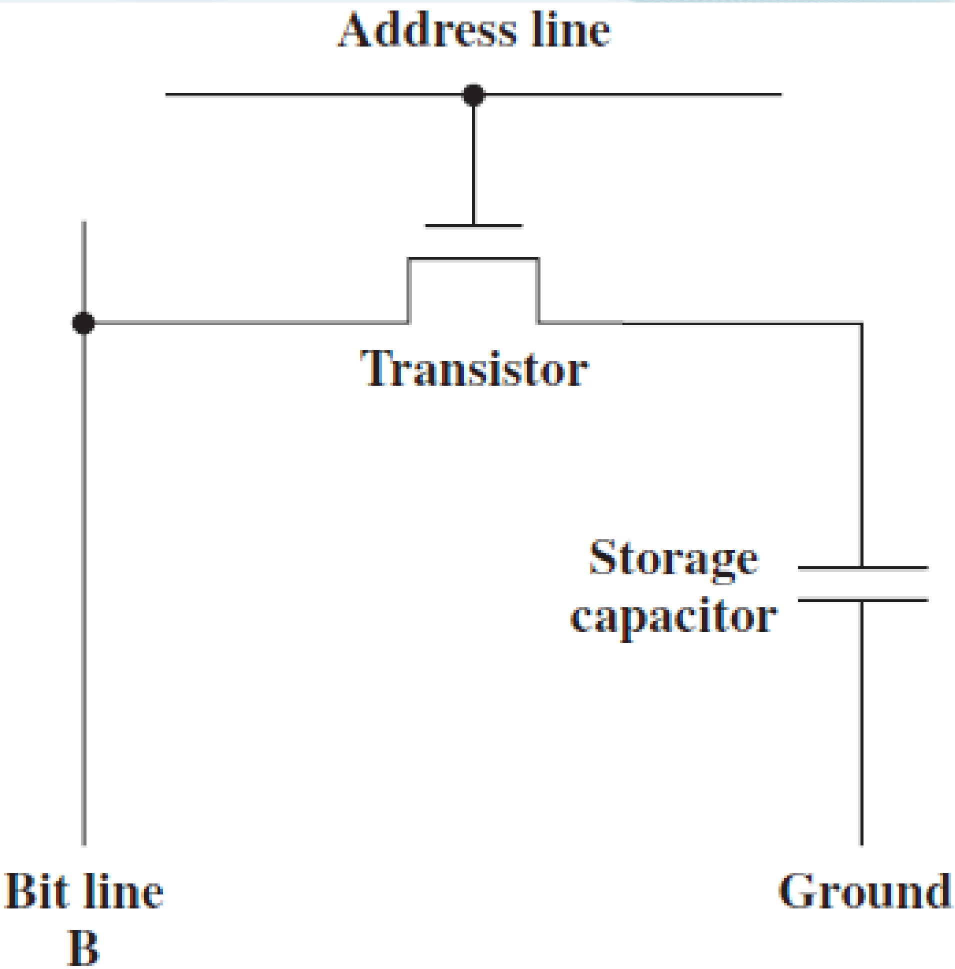
- **This figure** is a typical **DRAM** structure for an individual cell that stores **1 bit**.

- The address line is activated when the bit value from this cell is to be read or written.

- The transistor acts as a switch that is closed (allowing current to flow) if a voltage is applied to the address line and open (no current flows) if no voltage is present on the address line.

Figure 5.2a

Typical Memory Cell Structures



(a) Dynamic RAM (DRAM) cell

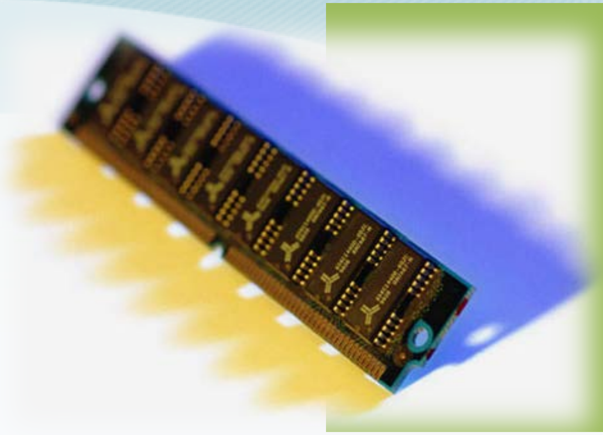
In electrical engineering, a switch is an electrical component that can break an electrical circuit, interrupting the current or diverting it from one conductor to another.

Dynamic RAM (DRAM)

- For the write operation, a voltage signal is applied to the bit line; a high voltage represents 1, and a low voltage represents 0.
- A signal is then applied to the address line, allowing a charge to be transferred to the capacitor.
- For the read operation, when the address line is selected, the transistor turns on and the charge stored on the capacitor is fed out onto a bit line and to a sense amplifier.

A capacitor (originally known as a condenser) is a passive two-terminal electrical component used to store electrical energy temporarily in an electric field.

Static RAM (SRAM)



- Digital device that uses the same logic elements used in the processor
- Binary values are stored using traditional flip-flop logic gate configurations
- Will hold its data as long as power is supplied to it

- SRAM does not have to be periodically refreshed.
- Static RAM provides faster access to data and is more expensive than DRAM.
- SRAM is used for a computer's cache memory and as part of the random access memory digital-to-analog converter on a video card.

SRAM versus DRAM

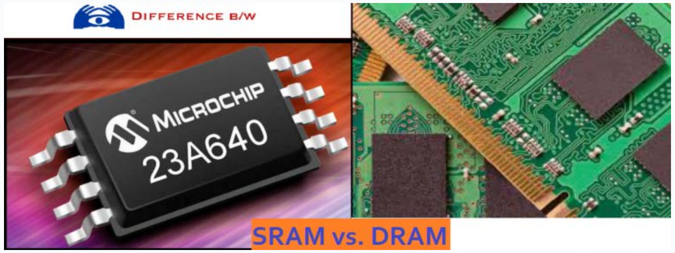
SRAM

- **Both Volatile**
  - **Power must be continuously supplied to the memory to preserve the bit values**

- **Dynamic Cell**

DRAM

  - **Simpler to build, smaller**
  - **More dense (smaller cells = more cells per unit area)**
  - **Less expensive**
  - **Requires the supporting refresh circuitry**
  - **Tend to be favored for large memory requirements**
  - **Used for main memory**



- **Static**
  - **Faster**
  - **Used for cache memory (both on and off chip)**

Memory Hierarchy

A typical hierarchy is illustrated in Figure 4.1.

As one goes down the hierarchy, the following occur:

a) Decreasing cost per bit

b) Increasing capacity

c) Increasing access time

d) Decreasing frequency of access of the memory by the processor

Thus, smaller, more expensive and faster memories are supplemented by larger, cheaper and slower memories.

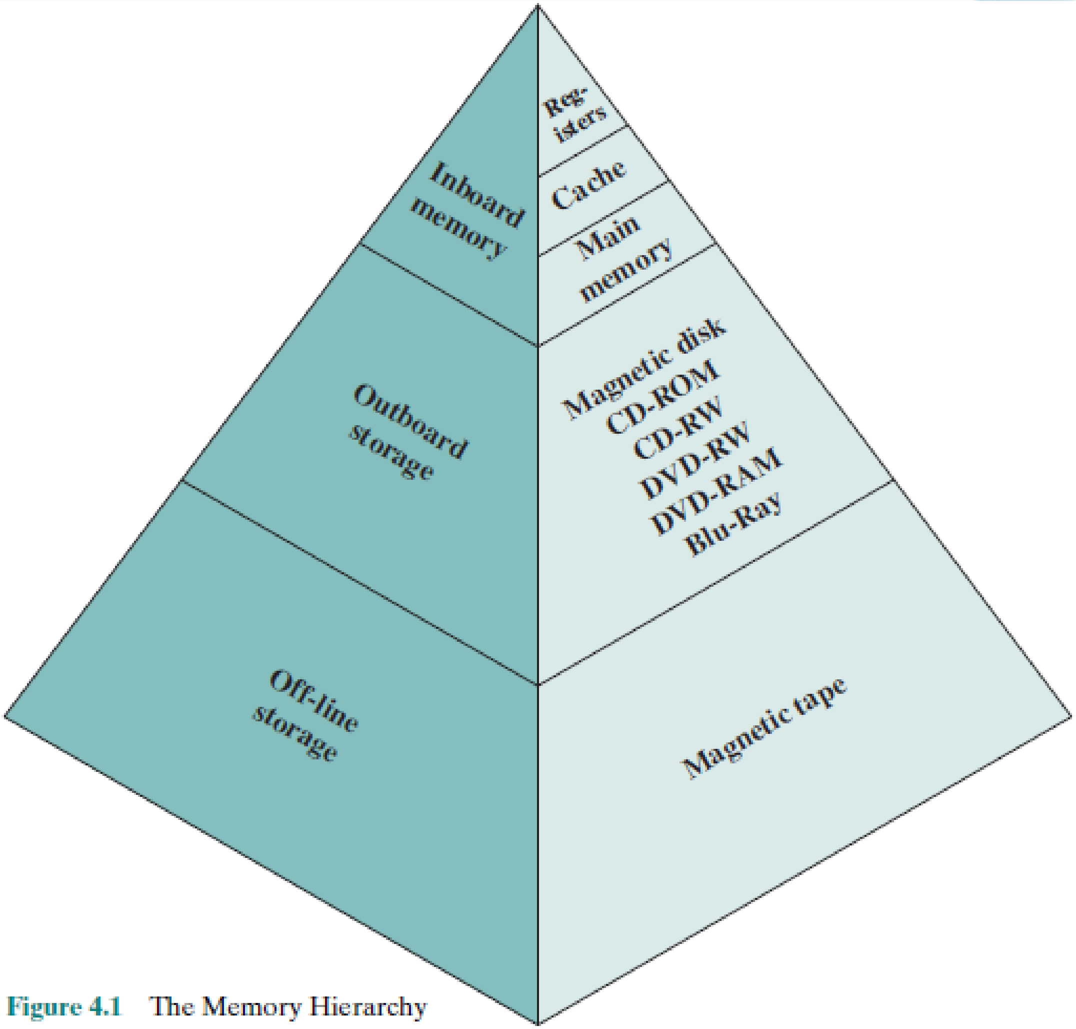


Figure 4.1 The Memory Hierarchy

The fastest, smallest, and most expensive type of memory consists of the registers internal to the processor.

Read Only Memory (ROM)

- **Contains a permanent pattern of data that cannot be changed.**
- **No power source is required to maintain the bit values in the memory.**
- **Data or program is permanently in main memory and never needs to be loaded from a secondary storage device.**
- **Data is actually wired into the chip as a part of the fabrication process.**
  - **Disadvantages of this:**
    - **No room for error, if one bit is wrong the whole batch of ROMs must be thrown out**
    - **Data insertion step includes a relatively large fixed cost**

Programmable ROM (PROM)

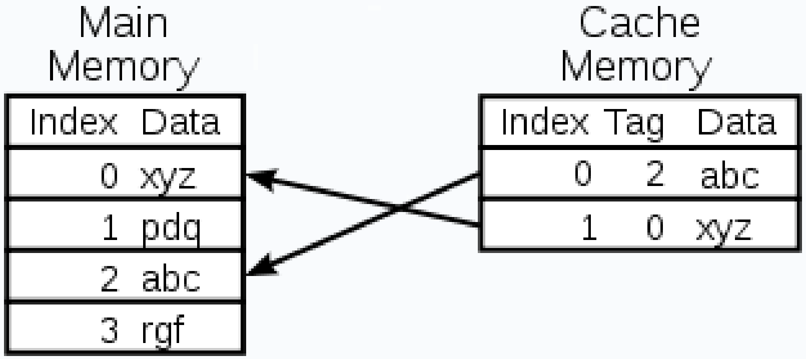
- **Less expensive alternative**
- ***Nonvolatile*** **and may be written into only once**
- **Writing process is performed electrically and may be performed by supplier or customer at a time later than the original chip fabrication**
- **Special equipment is required for the writing process**
- **Provides flexibility and convenience**
- **Attractive for high volume production runs**

Cache Memory

- ***Cache*** is a component that stores data so future requests for that data can be served faster.

- A ***cache hit*** occurs when the requested data can be found in a cache, while a ***cache miss*** occurs when it cannot. Cache has three levels

  - **L1 – Fatest**
  - **L2 – Less Fast**
  - **L3 – Slowest**



Main Memory

Index	Data
0	xyz
1	pdq
2	abc
3	rgf

Cache Memory

Index	Tag	Data
0	2	abc
1	0	xyz

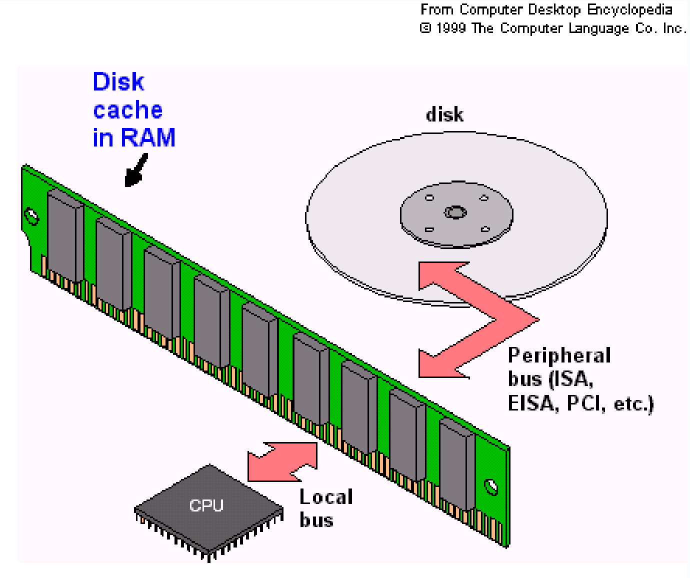
Imagine like eating a food, bag of clips like L1, L2 like a kitchen and kitchen is much larger and takes more time to find. L3 is the backward storage and needs more time to find.

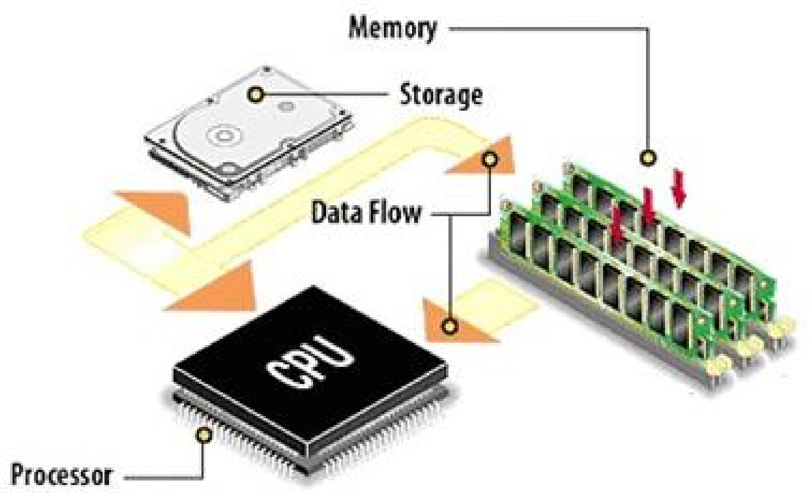
Cache Memory

- **Caches are generally the top level or levels of the memory hierarchy.**

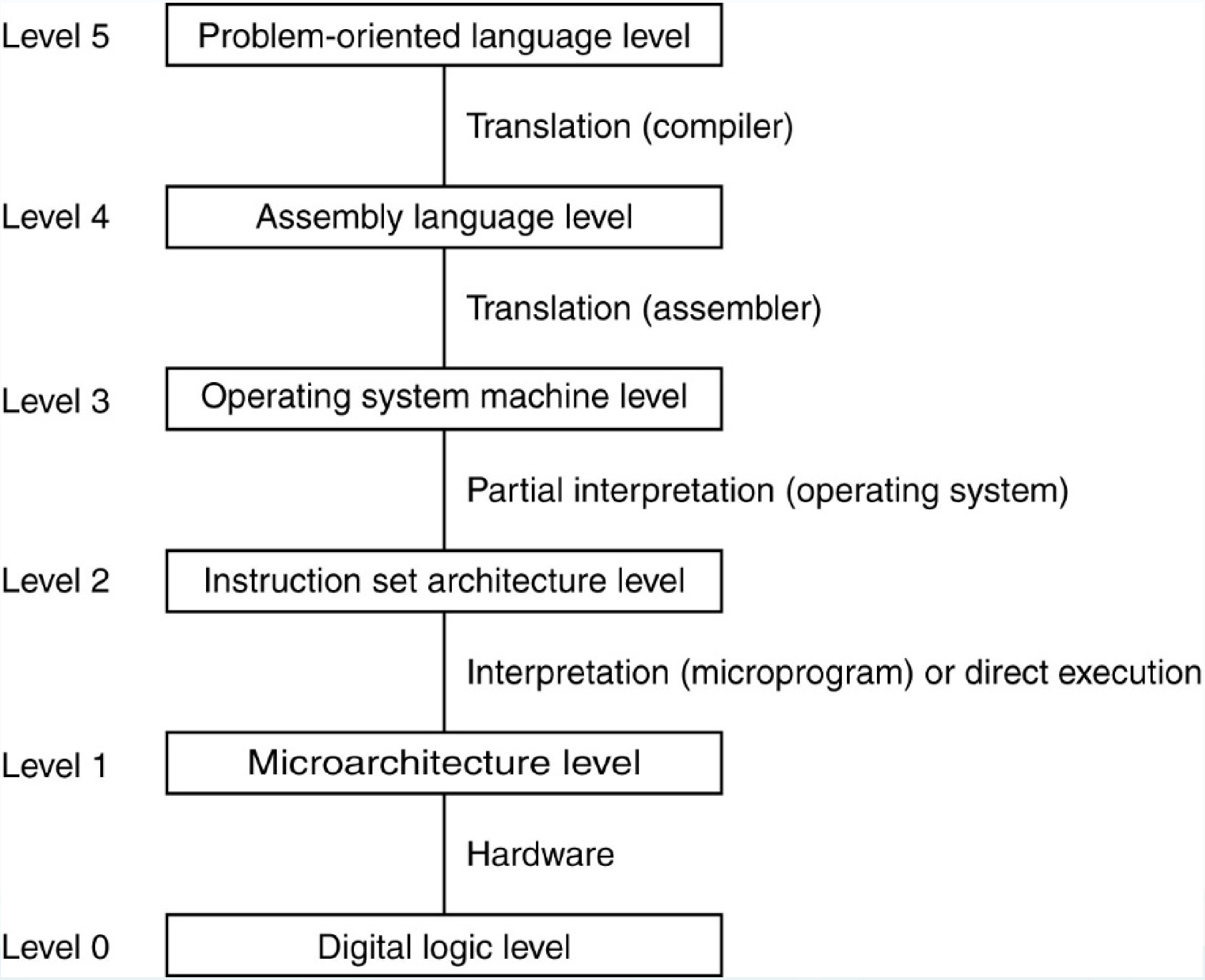
- **The main structural difference between a cache and other levels in the memory hierarchy is that caches contain hardware to track the memory addresses that are contained in the cache and to move data into and out of cache as necessary.**

- **Lower levels of memory requires a software or a combination of hardware and software to perform this function.**


From Computer Desktop Encyclopedia
© 1999 The Computer Language Co. Inc.



Contemporary Multi-level Computer Architecture



Level 5 Problem-oriented language level

Translation (compiler)

Level 4 Assembly language level

Translation (assembler)

Level 3 Operating system machine level

Partial interpretation (operating system)

Level 2 Instruction set architecture level

Interpretation (microprogram) or direct execution

Level 1 Microarchitecture level

Hardware

Level 0 Digital logic level

Module Resources/ References

Dr. Muhammad M Iqbal* (NCIRL)

Recommended Book Resources

- **William Stallings, COMPUTER ORGANIZATION AND ARCHITECTURE DESIGNING FOR PERFORMANCE, NINTH EDITION, Boston, ISBN 13: 978-0-13-293633-0, PEARSON**

- **Patterson, D and Hennessy, J 2012, omputer Organization and Design: The Hardware/Software Interface, Revised 4th Edition Ed., Waltham, MA : Morgan Kaufmann**

Supplementary Book Resources

- **Morris, M. and Kime C 2008, Logic and Computer Design Fundamentals, Pearson International Edition**