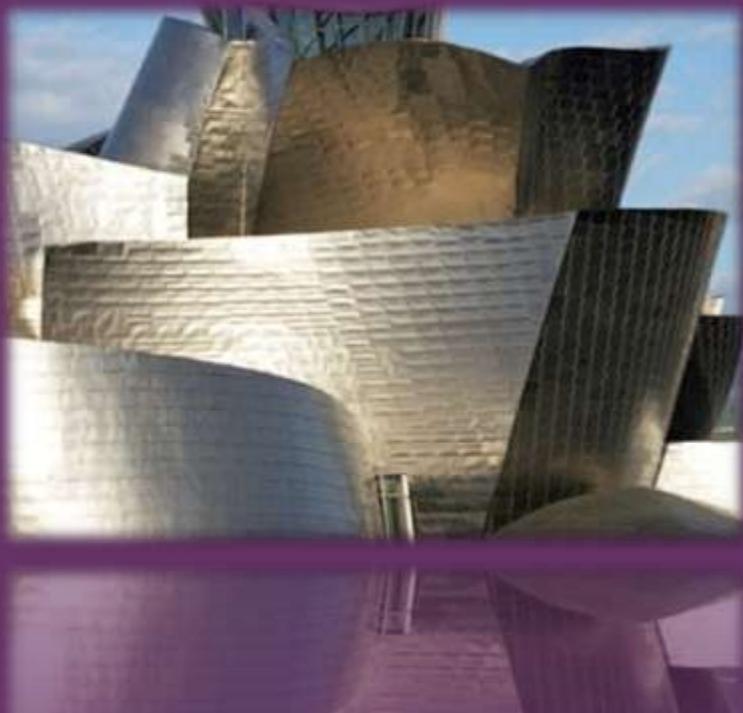


+



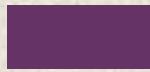
William Stallings
Computer Organization
and Architecture
10th Edition

Chapter 1

+

Basic Concepts and Computer Evolution

Computer Architecture



Computer Organization

- Attributes of a system visible to the programmer
- Have a direct impact on the logical execution of a program

- Instruction set, number of bits used to represent various data types, I/O mechanisms, techniques for addressing memory

Computer
Architecture

Architectural
attributes
include:

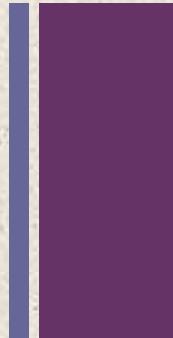
Organizational
attributes
include:

Computer
Organization

- Hardware details transparent to the programmer, control signals, interfaces between the computer and peripherals, memory technology used

- The operational units and their interconnections that realize the architectural specifications

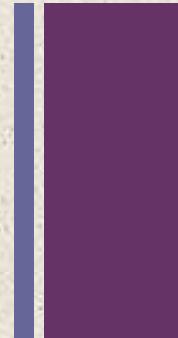
IBM System 370 Architecture



- IBM System/370 architecture
 - Was introduced in 1970
 - Included a number of models
 - Could upgrade to a more expensive, faster model without having to abandon original software
 - New models are introduced with improved technology, but retain the same architecture so that the customer's software investment is protected
 - Architecture has survived to this day as the architecture of IBM's mainframe product line



Structure and Function



- Hierarchical system
 - Set of interrelated subsystems
- Hierarchical nature of complex systems is essential to both their design and their description
- Designer need only deal with a particular level of the system at a time
 - Concerned with structure and function at each level
- Structure
 - The way in which components relate to each other
- Function
 - The operation of individual components as part of the structure





Function

- There are four basic functions that a computer can perform:
 - Data processing
 - Data may take a wide variety of forms and the range of processing requirements is broad
 - Data storage
 - Short-term
 - Long-term
 - Data movement
 - Input-output (I/O) - when data are received from or delivered to a device (peripheral) that is directly connected to the computer
 - Data communications – when data are moved over longer distances, to or from a remote device
 - Control
 - A control unit manages the computer's resources and orchestrates the performance of its functional parts in response to instructions

Structure

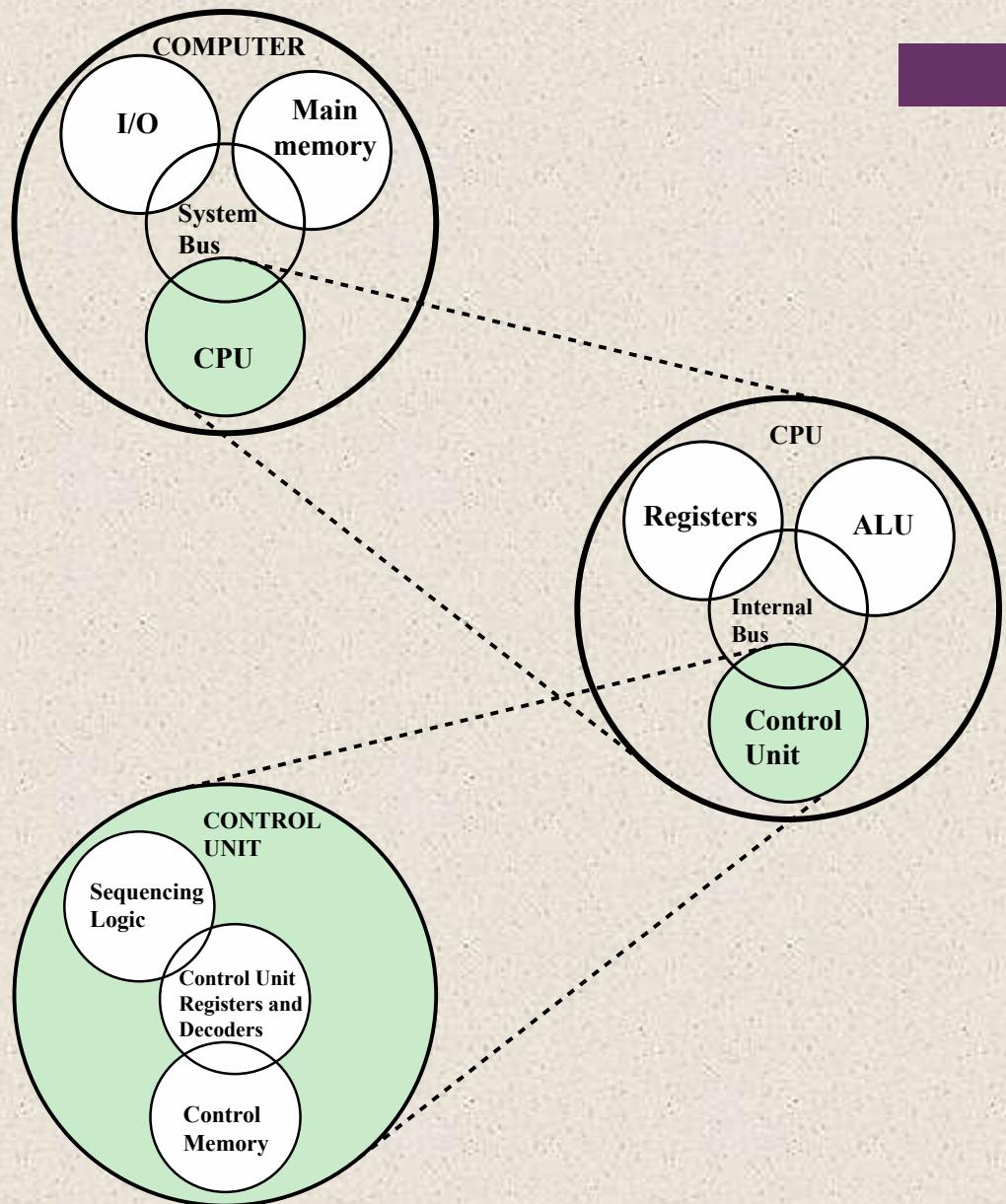
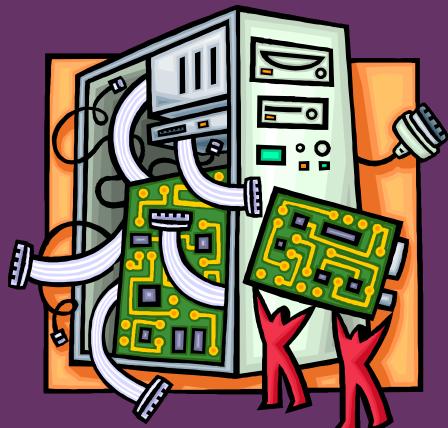


Figure 1.1 A Top-Down View of a Computer



There are four main structural components of the computer:

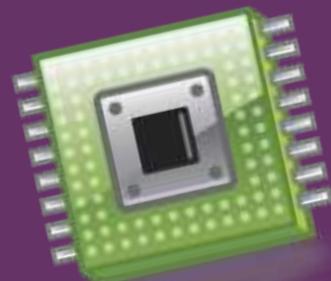
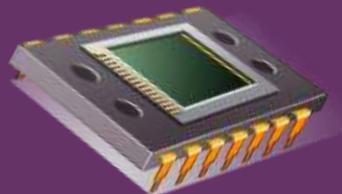


- ◆ CPU – controls the operation of the computer and performs its data processing functions
- ◆ Main Memory – stores data
- ◆ I/O – moves data between the computer and its external environment
- ◆ System Interconnection – some mechanism that provides for communication among CPU, main memory, and I/O



CPU

Major structural components:



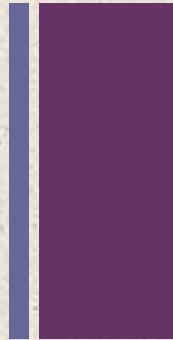
- Control Unit
 - Controls the operation of the CPU and hence the computer
- Arithmetic and Logic Unit (ALU)
 - Performs the computer's data processing function
- Registers
 - Provide storage internal to the CPU
- CPU Interconnection
 - Some mechanism that provides for communication among the control unit, ALU, and registers

Multicore Computer Structure

- Central processing unit (CPU)
 - Portion of the computer that fetches and executes instructions
 - Consists of an ALU, a control unit, and registers
 - Referred to as a processor in a system with a single processing unit
- Core
 - An individual processing unit on a processor chip
 - May be equivalent in functionality to a CPU on a single-CPU system
 - Specialized processing units are also referred to as cores
- Processor
 - A physical piece of silicon containing one or more cores
 - Is the computer component that interprets and executes instructions
 - Referred to as a *multicore processor* if it contains multiple cores



Cache Memory



- Multiple layers of memory between the processor and main memory
- Is smaller and faster than main memory
- Used to speed up memory access by placing in the cache data from main memory that is likely to be used in the near future
- A greater performance improvement may be obtained by using multiple levels of cache, with level 1 (L1) closest to the core and additional levels (L2, L3, etc.) progressively farther from the core

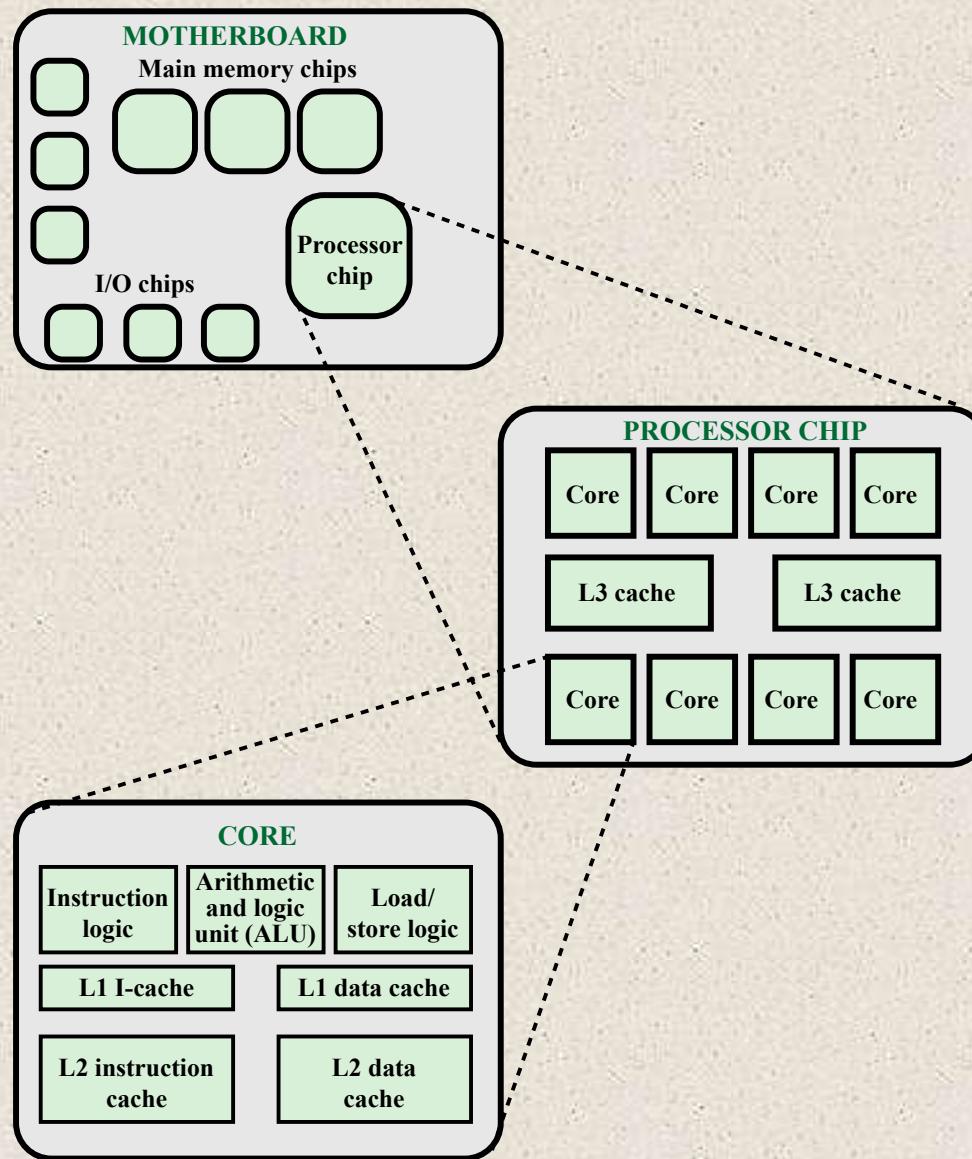


Figure 1.2 Simplified View of Major Elements of a Multicore Computer

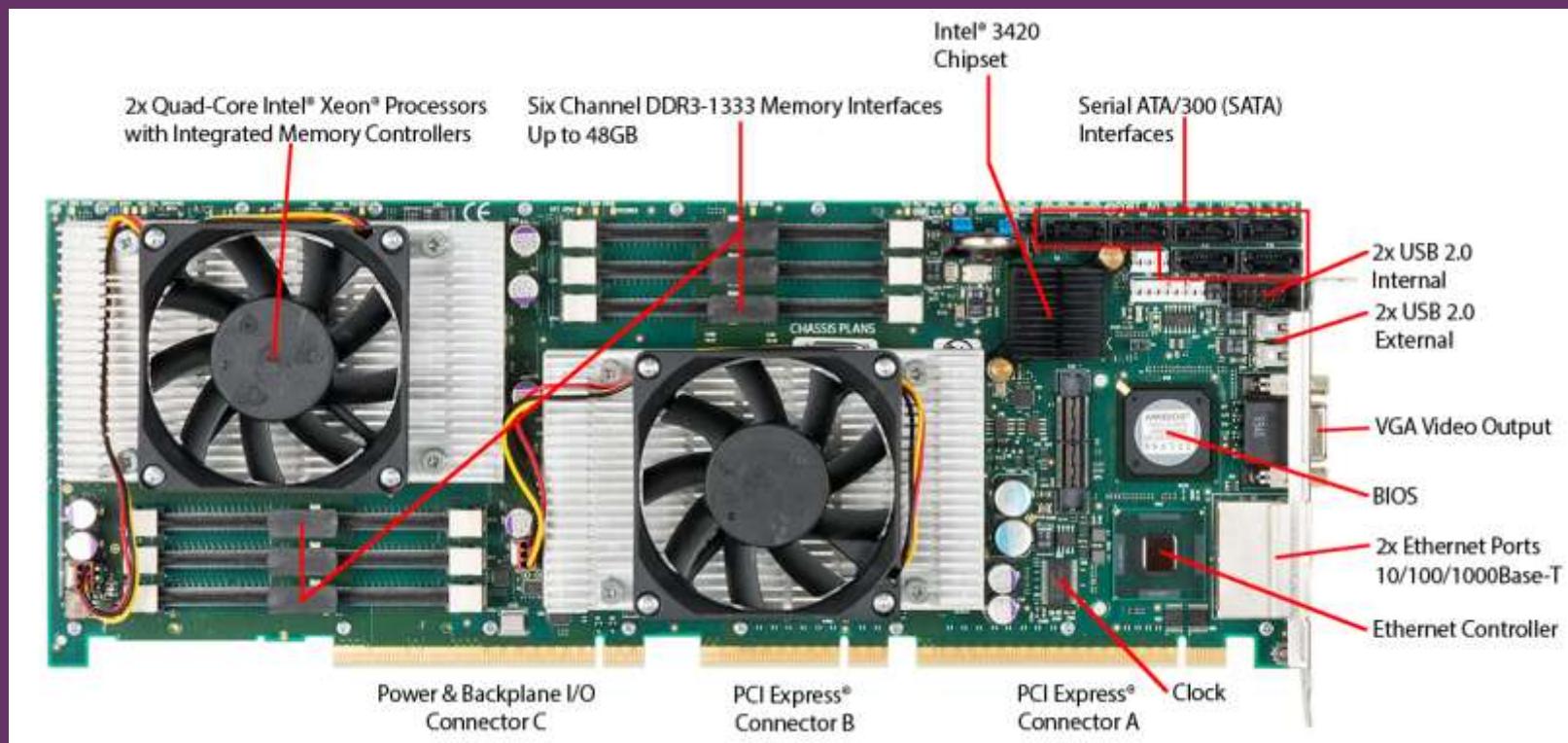


Figure 1.3
Motherboard with Two Intel Quad-Core Xeon Processors

Figure 1.4

zEnterprise
EC12 Processor
Unit (PU)
Chip Diagram

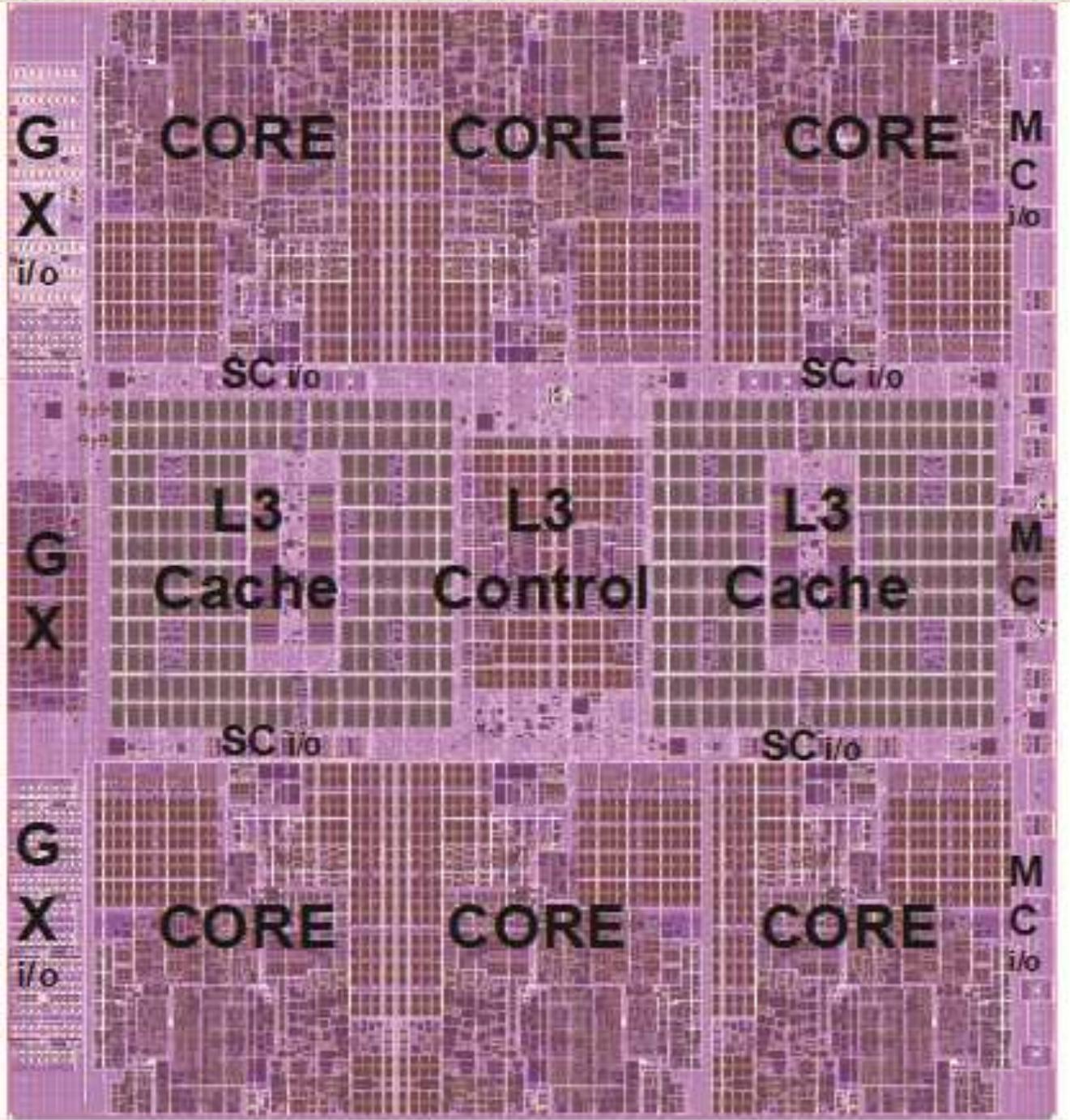
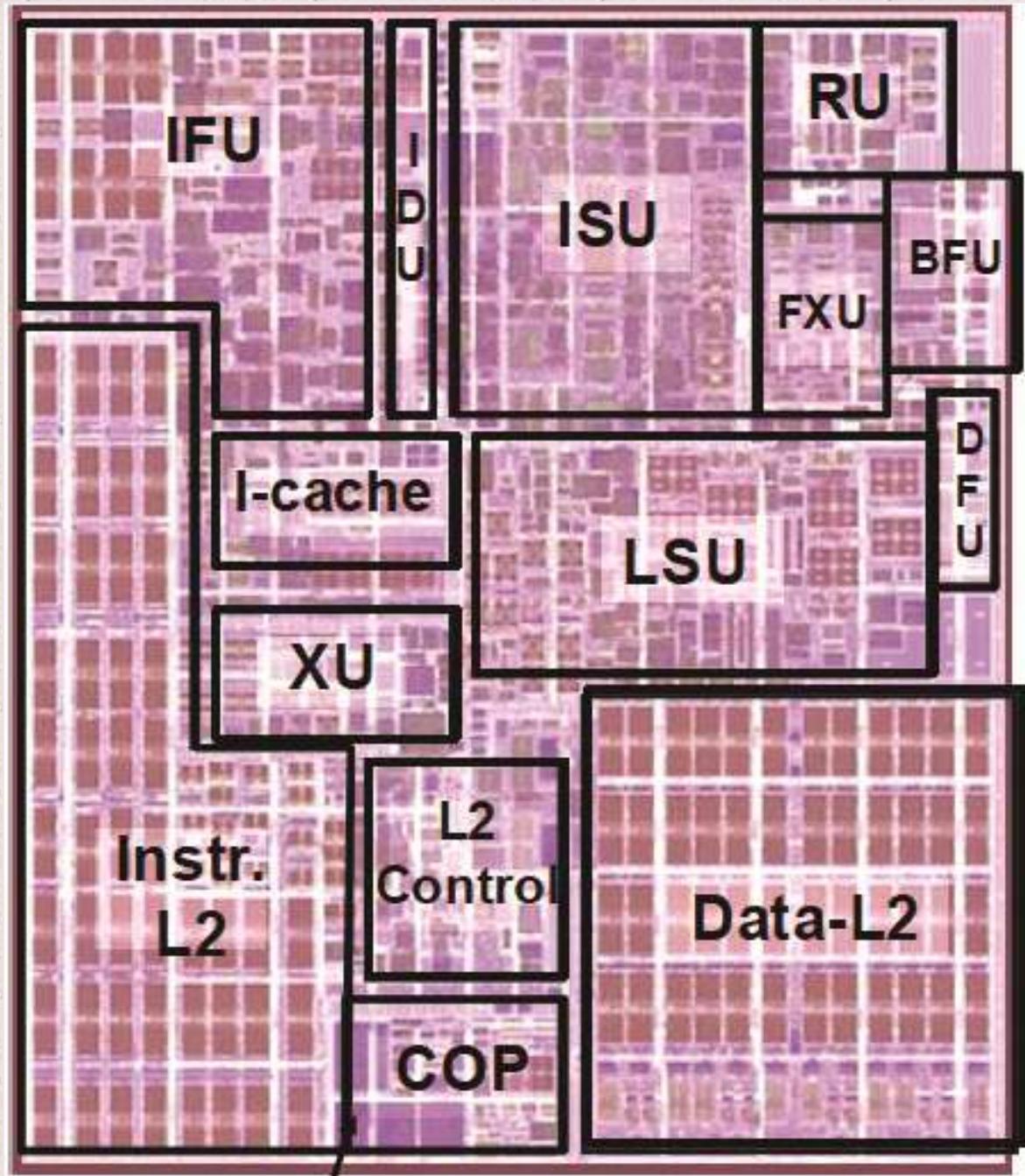


Figure 1.5
zEnterprise
EC12
Core Layout



History of Computers

First Generation: Vacuum Tubes

- Vacuum tubes were used for digital logic and memory
- IAS computer
 - Fundamental design approach was the stored program concept
 - Attributed to the mathematician John von Neumann
 - First publication of the idea was in 1945 for the EDVAC
 - Design began at the Princeton Institute for Advanced Studies
 - Completed in 1952
 - Prototype of all subsequent general-purpose computers



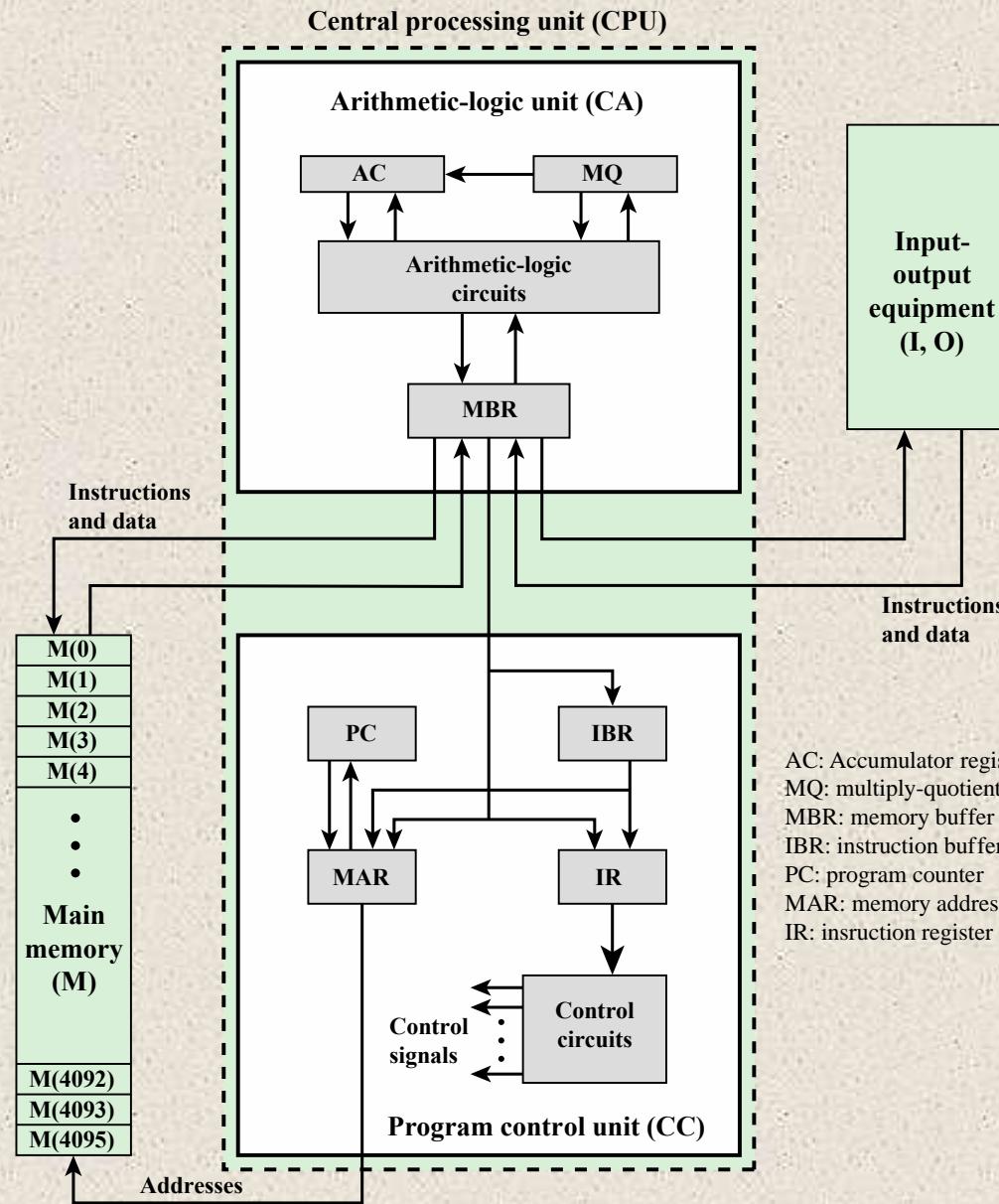


Figure 1.6 IAS Structure

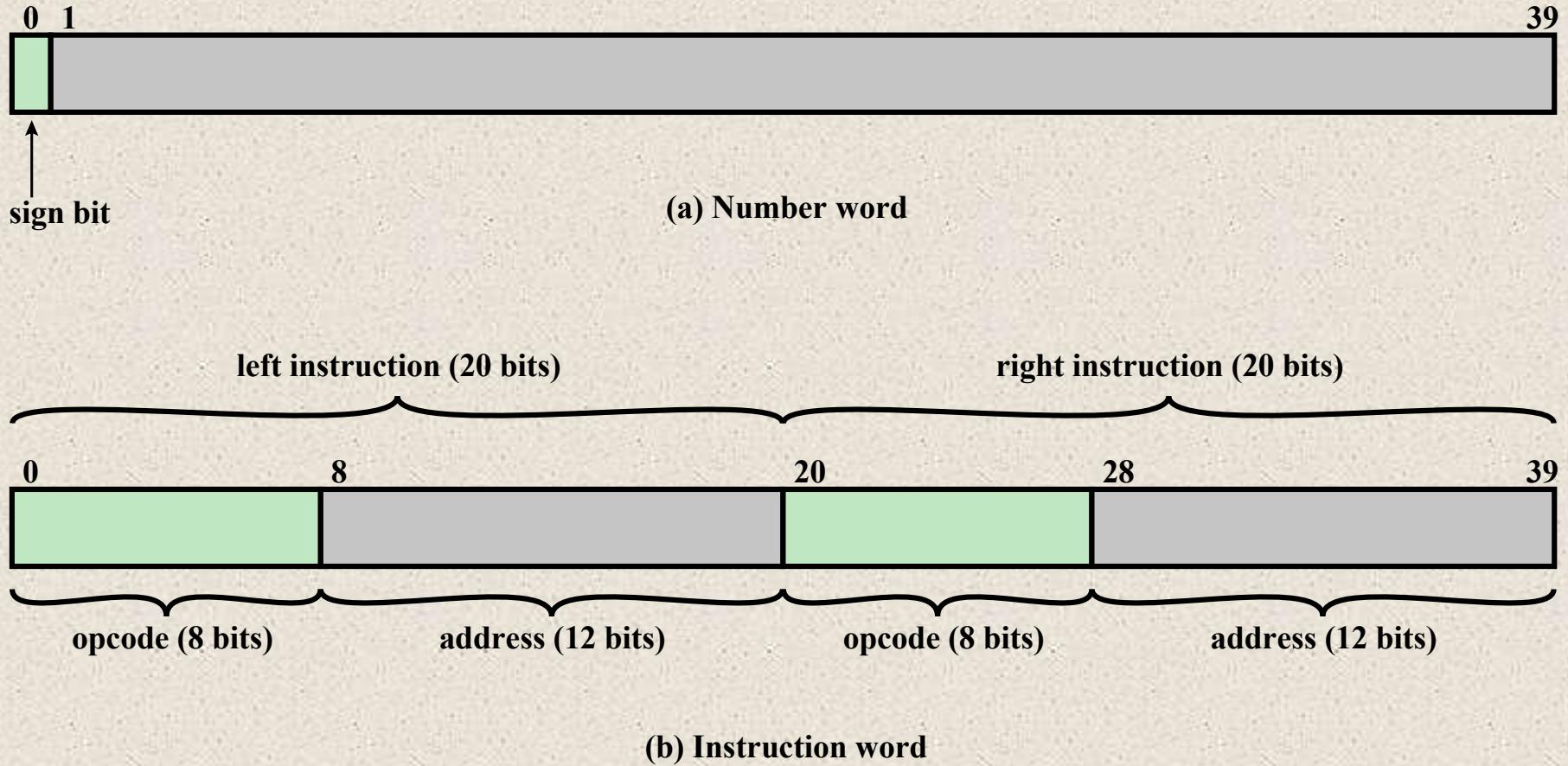


Figure 1.7 IAS Memory Formats

Registers

Memory buffer register (MBR)

- Contains a word to be stored in memory or sent to the I/O unit
- Or is used to receive a word from memory or from the I/O unit

Memory address register (MAR)

- Specifies the address in memory of the word to be written from or read into the MBR

Instruction register (IR)

- Contains the 8-bit opcode instruction being executed

Instruction buffer register (IBR)

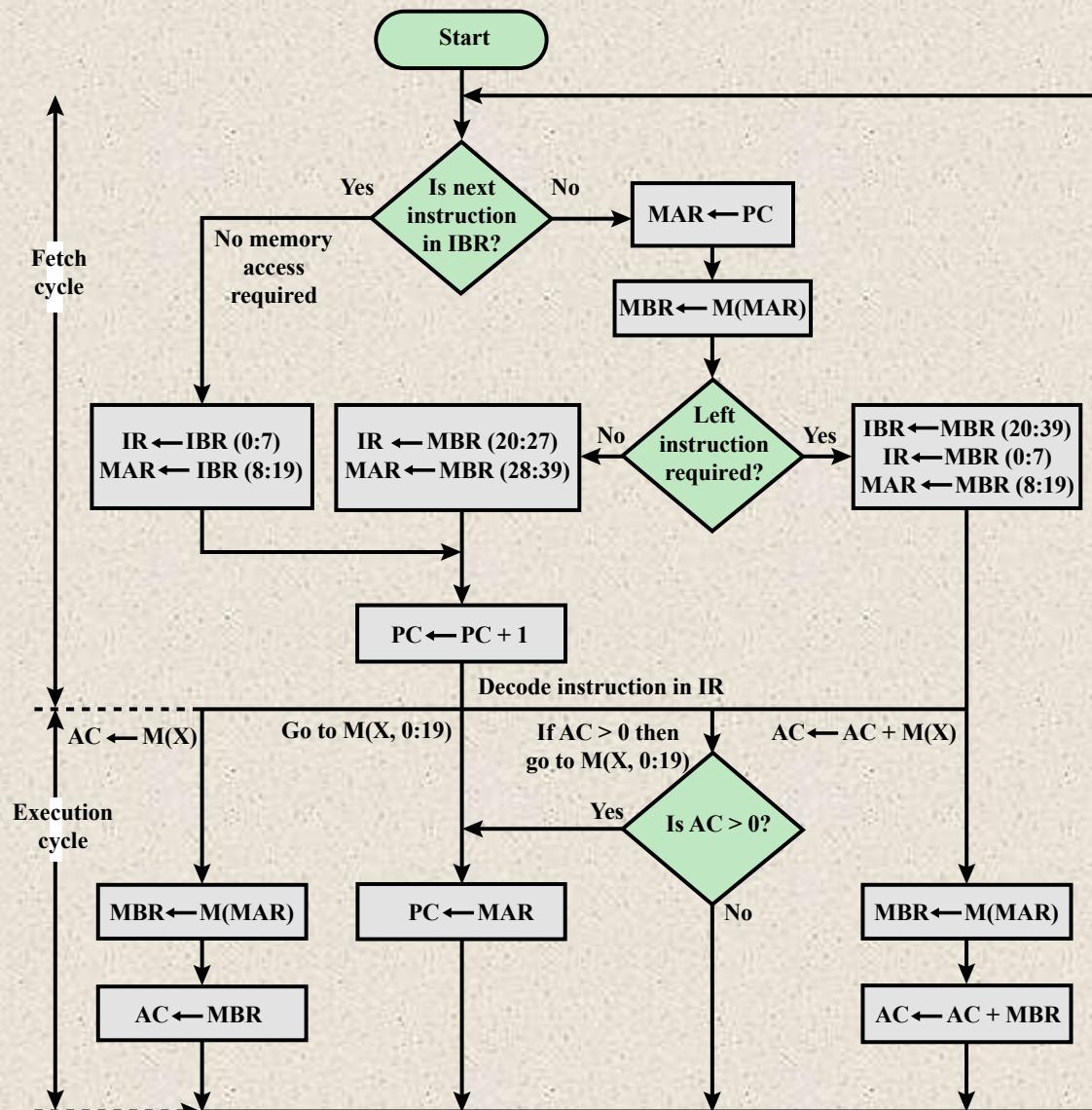
- Employed to temporarily hold the right-hand instruction from a word in memory

Program counter (PC)

- Contains the address of the next instruction pair to be fetched from memory

Accumulator (AC) and multiplier quotient (MQ)

- Employed to temporarily hold operands and results of ALU operations



$M(X)$ = contents of memory location whose address is X
 $(i:j)$ = bits i through j

Figure 1.8 Partial Flowchart of IAS Operation

Instruction Type	Opcode	Symbolic Representation	Description
Data transfer	00001010	LOAD MQ	Transfer contents of register MQ to the accumulator AC
	00001001	LOAD MQ,M(X)	Transfer contents of memory location X to MQ
	00100001	STOR M(X)	Transfer contents of accumulator to memory location X
	00000001	LOAD M(X)	Transfer M(X) to the accumulator
	00000010	LOAD -M(X)	Transfer -M(X) to the accumulator
	00000011	LOAD M(X)	Transfer absolute value of M(X) to the accumulator
Unconditional branch	00000100	LOAD - M(X)	Transfer - M(X) to the accumulator
	00001101	JUMP M(X,0:19)	Take next instruction from left half of M(X)
Conditional branch	00001110	JUMP M(X,20:39)	Take next instruction from right half of M(X)
	00001111	JUMP+ M(X,0:19)	If number in the accumulator is nonnegative, take next instruction from left half of M(X)
Arithmetic	JU MP $+$ $M(X)$ $,20:$ $39)$		<i>If number in the accumulator is nonnegative, take next instruction from right half of M(X)</i>
	00000101	ADD M(X)	Add M(X) to AC; put the result in AC
	00000111	ADD M(X)	Add M(X) to AC; put the result in AC
	00000110	SUB M(X)	Subtract M(X) from AC; put the result in AC
	00001000	SUB M(X)	Subtract M(X) from AC; put the remainder in AC
	00001011	MUL M(X)	Multiply M(X) by MQ; put most significant bits of result in AC, put least significant bits in MQ
	00001100	DIV M(X)	Divide AC by M(X); put the quotient in MQ and the remainder in AC
	00010100	LSH	Multiply accumulator by 2; i.e., shift left one bit position
	00010101	RSH	Divide accumulator by 2; i.e., shift right one position
Address modify	00010010	STOR M(X,8:19)	Replace left address field at M(X) by 12 rightmost bits of AC
	00010011	STOR M(X,28:39)	Replace right address field at M(X) by 12 rightmost bits of AC

Table 1.1

The IAS Instruction Set

(Table can be found on page 17 in the textbook.)

History of Computers

Second Generation: Transistors

- Smaller
- Cheaper
- Dissipates less heat than a vacuum tube
- Is a *solid state device* made from silicon
- Was invented at Bell Labs in 1947
- It was not until the late 1950's that fully transistorized computers were commercially available

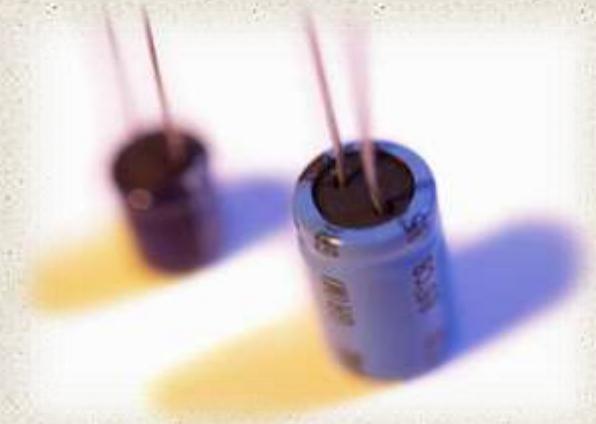


Table 1.2

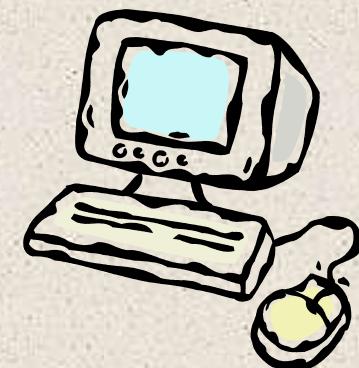
Computer Generations

Generation	Approximate Dates	Technology	Typical Speed (operations per second)
1	1946–1957	Vacuum tube	40,000
2	1957–1964	Transistor	200,000
3	1965–1971	Small and medium scale integration	1,000,000
4	1972–1977	Large scale integration	10,000,000
5	1978–1991	Very large scale integration	100,000,000
6	1991–	Ultra large scale integration	>1,000,000,000

Second Generation Computers

■ Introduced:

- More complex arithmetic and logic units and control units
- The use of high-level programming languages
- Provision of *system software* which provided the ability to:
 - Load programs
 - Move data to peripherals
 - Libraries perform common computations



IBM 7094 computer

Peripheral devices

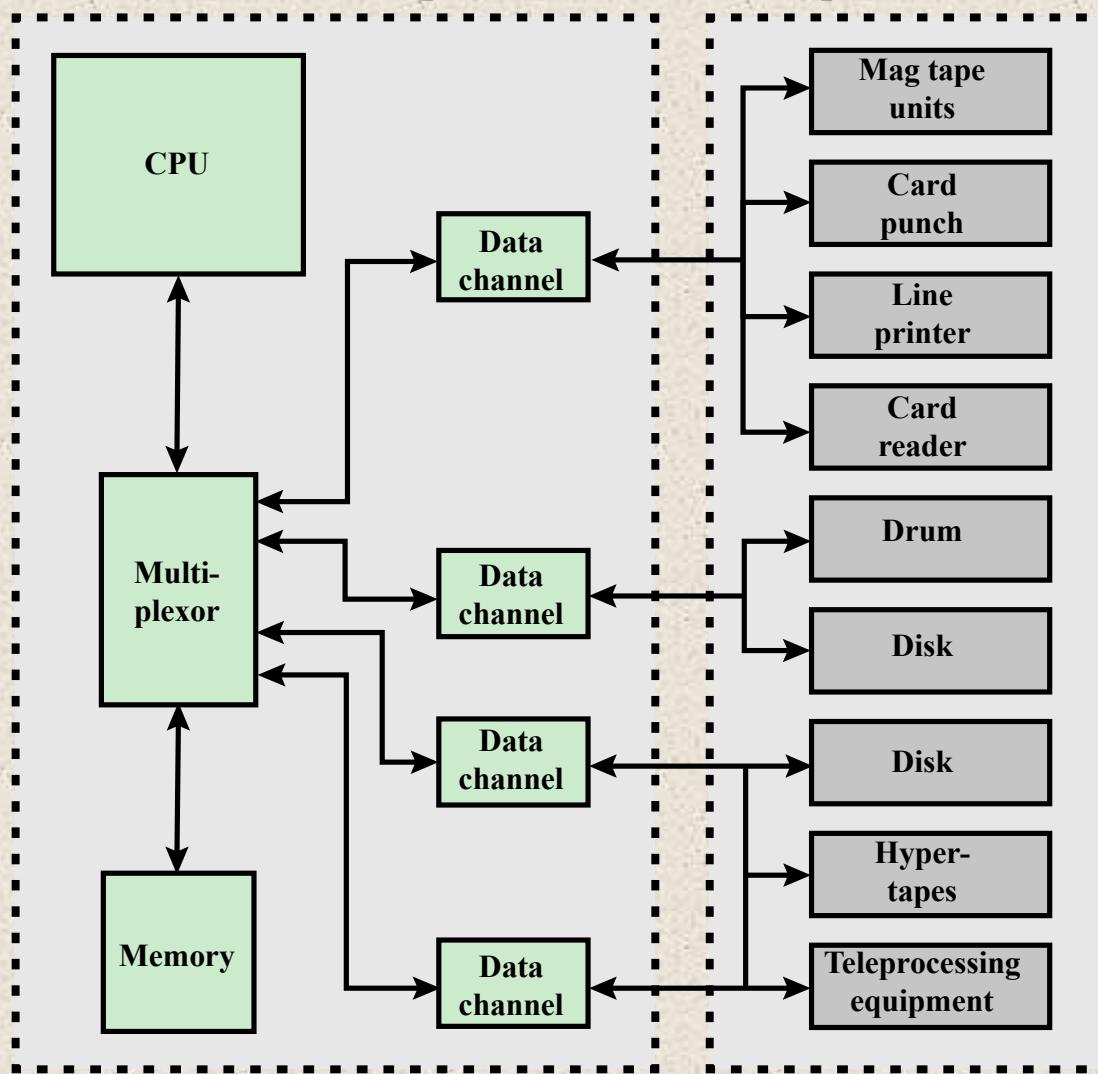


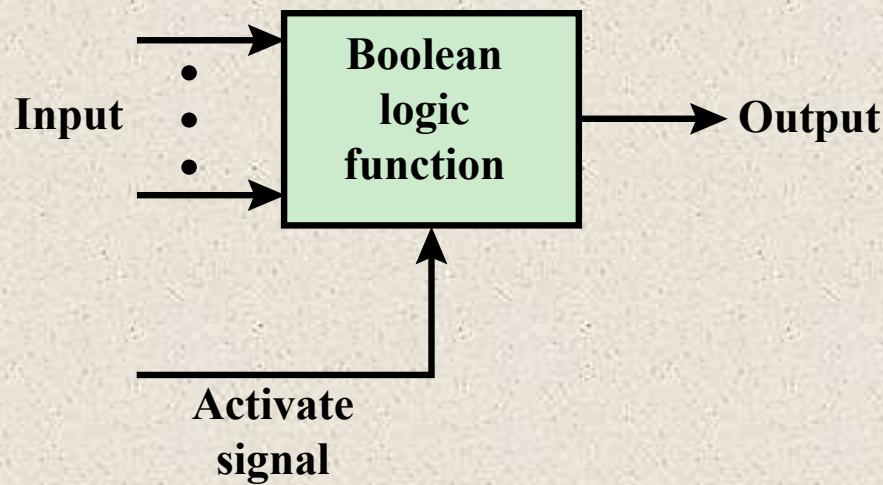
Figure 1.9 An IBM 7094 Configuration

History of Computers

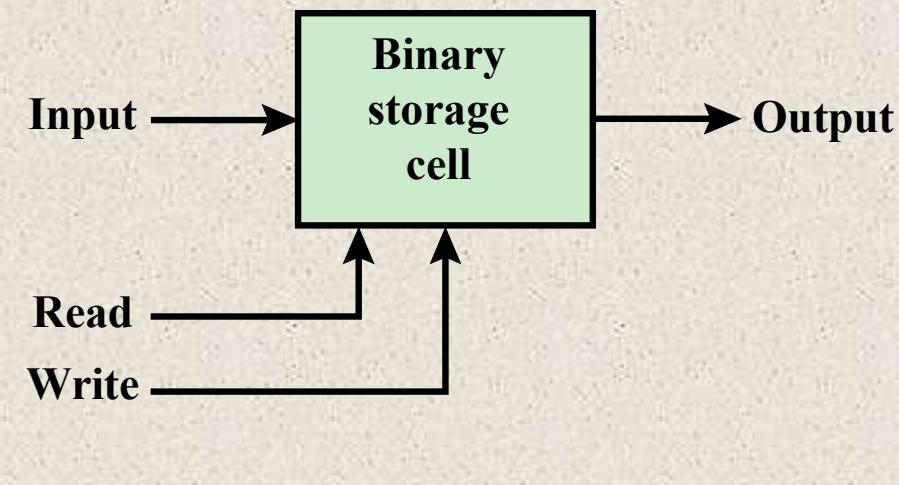
Third Generation: Integrated Circuits

- 1958 – the invention of the integrated circuit
- *Discrete component*
 - Single, self-contained transistor
 - Manufactured separately, packaged in their own containers, and soldered or wired together onto masonite-like circuit boards
 - Manufacturing process was expensive and cumbersome
- The two most important members of the third generation were the IBM System/360 and the DEC PDP-8





(a) Gate



(b) Memory cell

Figure 1.10 Fundamental Computer Elements

Integrated Circuits

- Data storage – provided by memory cells
- Data processing – provided by gates
- Data movement – the paths among components are used to move data from memory to memory and from memory through gates to memory
- Control – the paths among components can carry control signals
- A computer consists of gates, memory cells, and interconnections among these elements
- The gates and memory cells are constructed of simple digital electronic components
- Exploits the fact that such components as transistors, resistors, and conductors can be fabricated from a semiconductor such as silicon
- Many transistors can be produced at the same time on a single wafer of silicon
- Transistors can be connected with a processor metallization to form circuits

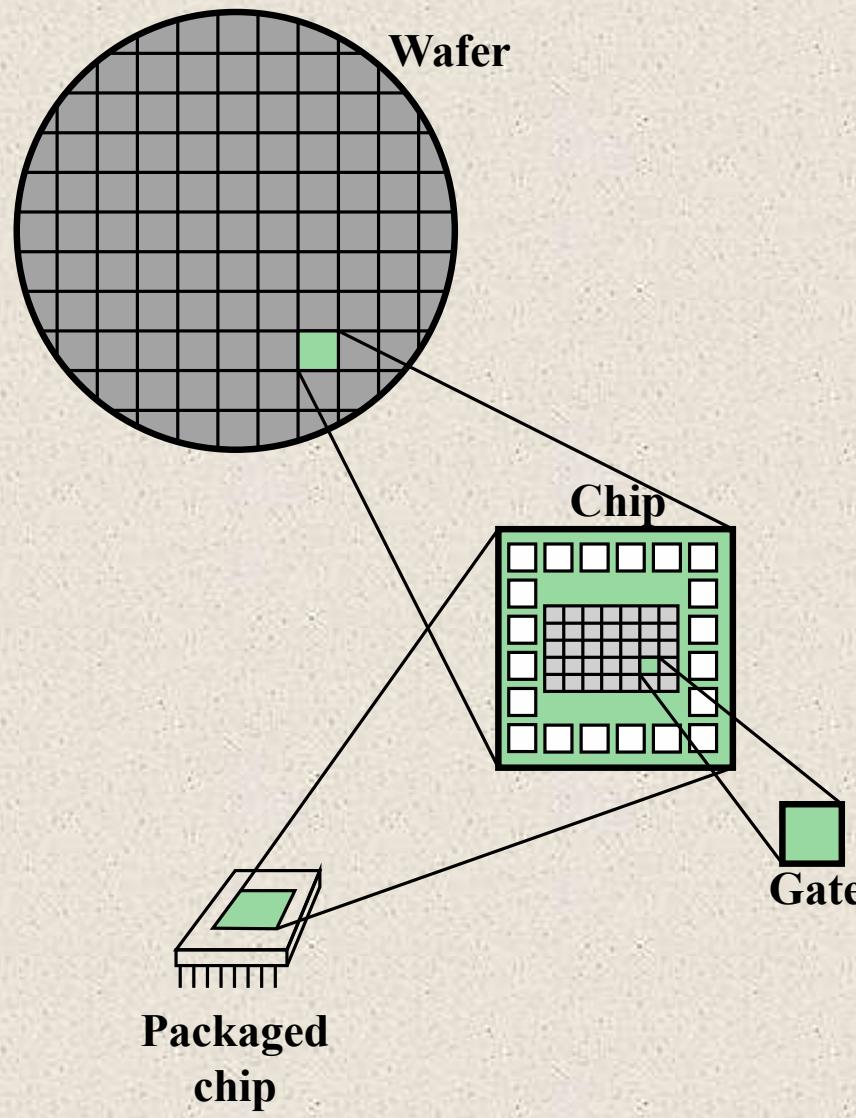


Figure 1.11 Relationship Among Wafer, Chip, and Gate

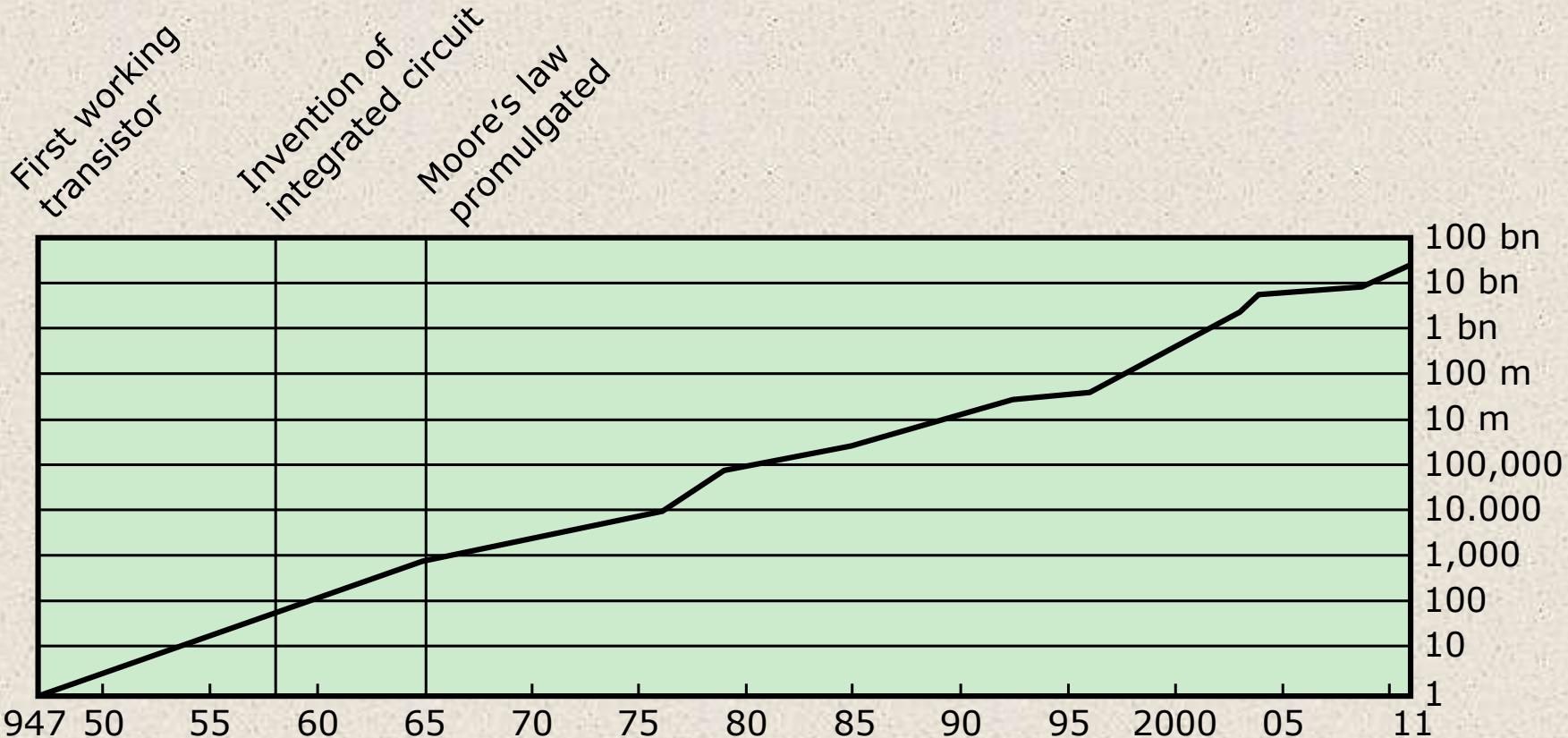


Figure 1.12 Growth in Transistor Count on Integrated Circuits (DRAM memory)

Moore's Law

1965; Gordon Moore – co-founder of Intel

Observed number of transistors that could be put on a single chip was doubling every year

The pace slowed to a doubling every 18 months in the 1970's but has sustained that rate ever since

Consequences of Moore's law:

The cost of computer logic and memory circuitry has fallen at a dramatic rate

The electrical path length is shortened, increasing operating speed

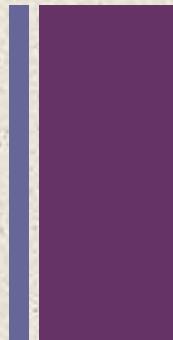
Computer becomes smaller and is more convenient to use in a variety of environments

Reduction in power and cooling requirements

Fewer interchip connections



IBM System/360



- Announced in 1964
- Product line was incompatible with older IBM machines
- Was the success of the decade and cemented IBM as the overwhelmingly dominant computer vendor
- The architecture remains to this day the architecture of IBM's mainframe computers
- Was the industry's first planned family of computers
 - Models were compatible in the sense that a program written for one model should be capable of being executed by another model in the series

+ Family Characteristics

Similar or identical instruction set

Similar or identical operating system

Increasing speed

Increasing number of I/O ports

Increasing memory size

Increasing cost

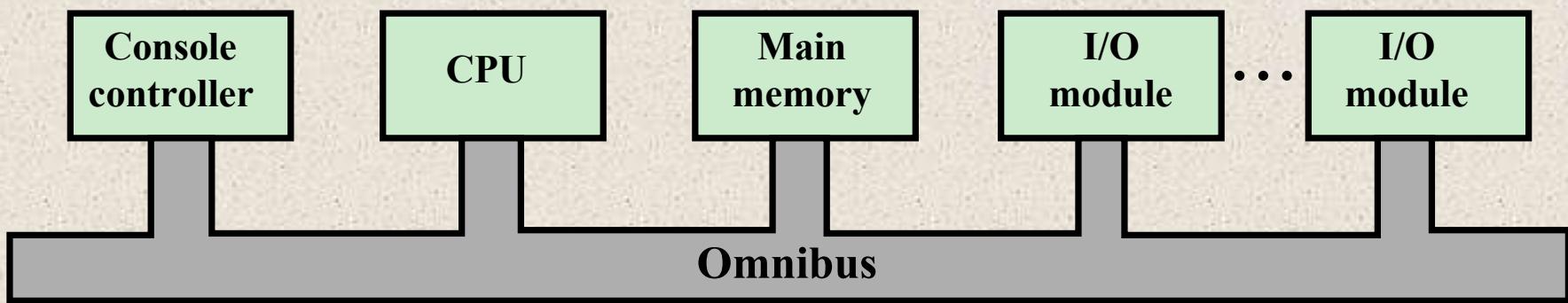
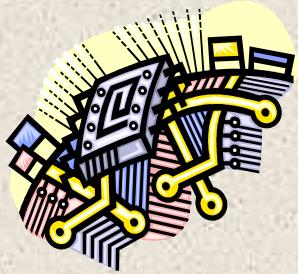


Figure 1.13 PDP-8 Bus Structure



Later Generations



Semiconductor Memory
Microprocessors

LSI
Large
Scale
Integration

VLSI
Very Large
Scale
Integration

ULSI
Ultra Large
Scale
Integration

Semiconductor Memory

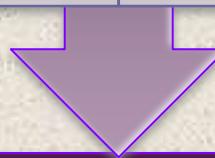
In 1970 Fairchild produced the first relatively capacious semiconductor memory

Chip was about the size of a single core

Could hold 256 bits of memory

Non-destructive

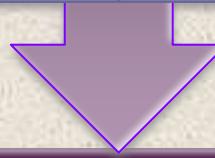
Much faster than core



In 1974 the price per bit of semiconductor memory dropped below the price per bit of core memory

There has been a continuing and rapid decline in memory cost accompanied by a corresponding increase in physical memory density

Developments in memory and processor technologies changed the nature of computers in less than a decade



Since 1970 semiconductor memory has been through 13 generations

Each generation has provided four times the storage density of the previous generation, accompanied by declining cost per bit and declining access time

Microprocessors

- The density of elements on processor chips continued to rise
 - More and more elements were placed on each chip so that fewer and fewer chips were needed to construct a single computer processor
- 1971 Intel developed 4004
 - First chip to contain all of the components of a CPU on a single chip
 - Birth of microprocessor
- 1972 Intel developed 8008
 - First 8-bit microprocessor
- 1974 Intel developed 8080
 - First general purpose microprocessor
 - Faster, has a richer instruction set, has a large addressing capability



Evolution of Intel Microprocessors

	4004	8008	8080	8086	8088
Introduced	1971	1972	1974	1978	1979
Clock speeds	108 kHz	108 kHz	2 MHz	5 MHz, 8 MHz, 10 MHz	5 MHz, 8 MHz
Bus width	4 bits	8 bits	8 bits	16 bits	8 bits
Number of transistors	2,300	3,500	6,000	29,000	29,000
Feature size (μm)	10	8	6	3	6
Addressable memory	640 Bytes	16 KB	64 KB	1 MB	1 MB

(a) 1970s Processors

Evolution of Intel Microprocessors

	80286	386TM DX	386TM SX	486TM DX CPU
Introduced	1982	1985	1988	1989
Clock speeds	6 MHz - 12.5 MHz	16 MHz - 33 MHz	16 MHz - 33 MHz	25 MHz - 50 MHz
Bus width	16 bits	32 bits	16 bits	32 bits
Number of transistors	134,000	275,000	275,000	1.2 million
Feature size (μm)	1.5	1	1	0.8 - 1
Addressable memory	16 MB	4 GB	16 MB	4 GB
Virtual memory	1 GB	64 TB	64 TB	64 TB
Cache	—	—	—	8 kB

(b) 1980s Processors

Evolution of Intel Microprocessors

	486TM SX	Pentium	Pentium Pro	Pentium II
Introduced	1991	1993	1995	1997
Clock speeds	16 MHz - 33 MHz	60 MHz - 166 MHz,	150 MHz - 200 MHz	200 MHz - 300 MHz
Bus width	32 bits	32 bits	64 bits	64 bits
Number of transistors	1.185 million	3.1 million	5.5 million	7.5 million
Feature size (μm)	1	0.8	0.6	0.35
Addressable memory	4 GB	4 GB	64 GB	64 GB
Virtual memory	64 TB	64 TB	64 TB	64 TB
Cache	8 kB	8 kB	512 kB L1 and 1 MB L2	512 kB L2

(c) 1990s Processors

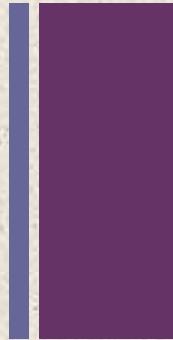
Evolution of Intel Microprocessors

	Pentium III	Pentium 4	Core 2 Duo	Core i7 EE 4960X
Introduced	1999	2000	2006	2013
Clock speeds	450 - 660 MHz	1.3 - 1.8 GHz	1.06 - 1.2 GHz	4 GHz
Bus width	64 bits	64 bits	64 bits	64 bits
Number of transistors	9.5 million	42 million	167 million	1.86 billion
Feature size (nm)	250	180	65	22
Addressable memory	64 GB	64 GB	64 GB	64 GB
Virtual memory	64 TB	64 TB	64 TB	64 TB
Cache	512 kB L2	256 kB L2	2 MB L2	1.5 MB L2/15 MB L3
Number of cores	1	1	2	6

(d) Recent Processors



The Evolution of the Intel x86 Architecture



- Two processor families are the Intel x86 and the ARM architectures
- Current x86 offerings represent the results of decades of design effort on complex instruction set computers (CISCs)
- An alternative approach to processor design is the reduced instruction set computer (RISC)
- ARM architecture is used in a wide variety of embedded systems and is one of the most powerful and best-designed RISC-based systems on the market

Highlights of the Evolution of the Intel Product Line:

8080

- World's first general-purpose microprocessor
- 8-bit machine, 8-bit data path to memory
- Was used in the first personal computer (Altair)

8086

- A more powerful 16-bit machine
- Has an instruction cache, or queue, that prefetches a few instructions before they are executed
- The first appearance of the x86 architecture
- The 8088 was a variant of this processor and used in IBM's first personal computer (securing the success of Intel)

80286

- Extension of the 8086 enabling addressing a 16-MB memory instead of just 1MB

80386

- Intel's first 32-bit machine
- First Intel processor to support multitasking

80486

- Introduced the use of much more sophisticated and powerful cache technology and sophisticated instruction pipelining
- Also offered a built-in math coprocessor

Highlights of the Evolution of the Intel Product Line:

Pentium

- Intel introduced the use of superscalar techniques, which allow multiple instructions to execute in parallel

Pentium Pro

- Continued the move into superscalar organization with aggressive use of register renaming, branch prediction, data flow analysis, and speculative execution

Pentium II

- Incorporated Intel MMX technology, which is designed specifically to process video, audio, and graphics data efficiently

Pentium III

- Incorporated additional floating-point instructions
- Streaming SIMD Extensions (SSE)

Pentium 4

- Includes additional floating-point and other enhancements for multimedia

Core

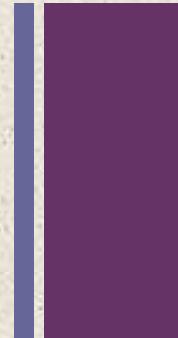
- First Intel x86 micro-core

Core 2

- Extends the Core architecture to 64 bits
- Core 2 Quad provides four cores on a single chip
- More recent Core offerings have up to 10 cores per chip
- An important addition to the architecture was the Advanced Vector Extensions instruction set



Embedded Systems



- The use of electronics and software within a product
- Billions of computer systems are produced each year that are embedded within larger devices
- Today many devices that use electric power have an embedded computing system
- Often embedded systems are tightly coupled to their environment
 - This can give rise to real-time constraints imposed by the need to interact with the environment
 - Constraints such as required speeds of motion, required precision of measurement, and required time durations, dictate the timing of software operations
 - If multiple activities must be managed simultaneously this imposes more complex real-time constraints



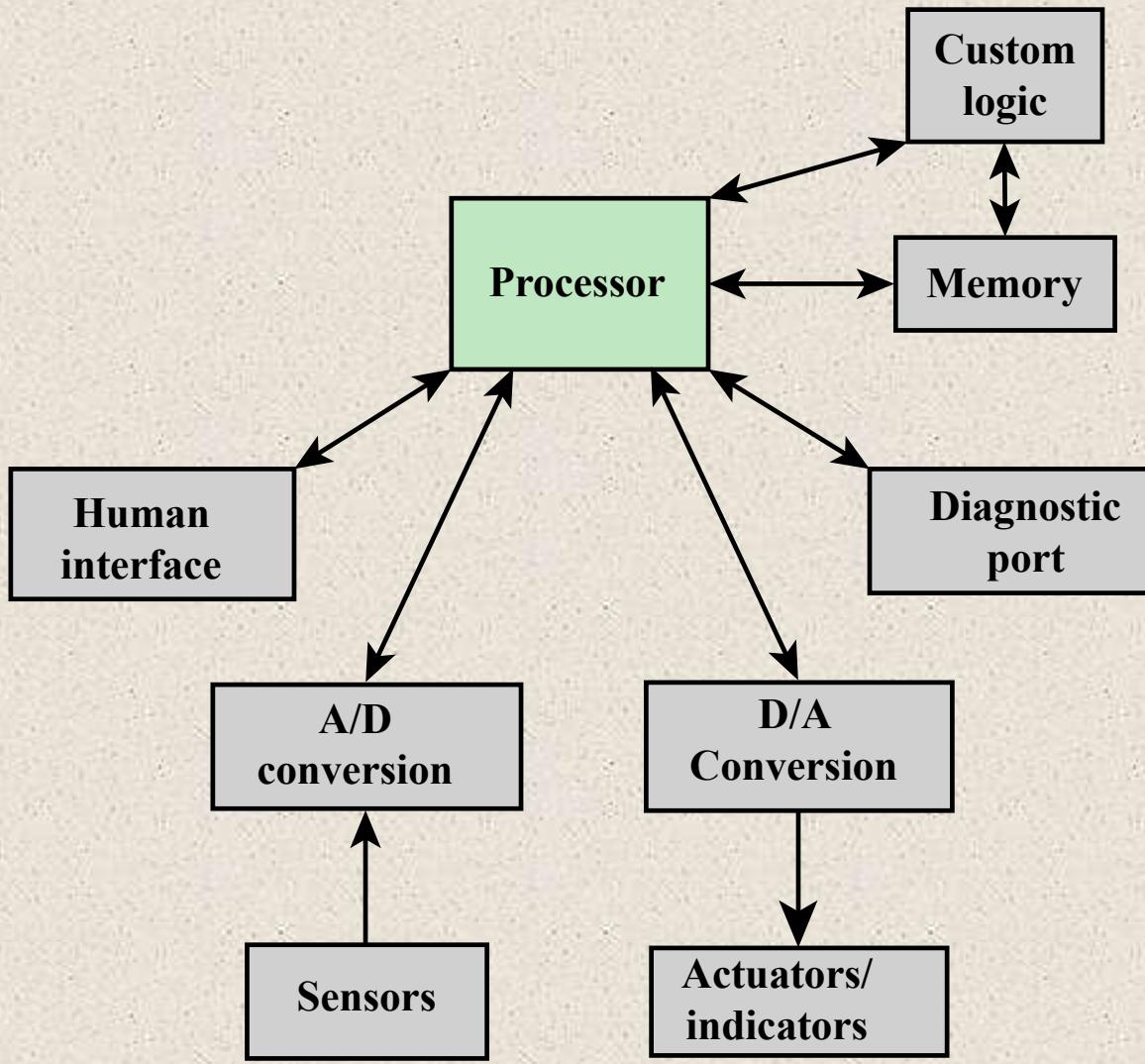


Figure 1.14 Possible Organization of an Embedded System



The Internet of Things (IoT)

- Term that refers to the expanding interconnection of smart devices, ranging from appliances to tiny sensors
- Is primarily driven by deeply embedded devices
- Generations of deployment culminating in the IoT:
 - Information technology (IT)
 - PCs, servers, routers, firewalls, and so on, bought as IT devices by enterprise IT people and primarily using wired connectivity
 - Operational technology (OT)
 - Machines/appliances with embedded IT built by non-IT companies, such as medical machinery, SCADA, process control, and kiosks, bought as appliances by enterprise OT people and primarily using wired connectivity
 - Personal technology
 - Smartphones, tablets, and eBook readers bought as IT devices by consumers exclusively using wireless connectivity and often multiple forms of wireless connectivity
 - Sensor/actuator technology
 - Single-purpose devices bought by consumers, IT, and OT people exclusively using wireless connectivity, generally of a single form, as part of larger systems
- It is the fourth generation that is usually thought of as the IoT and it is marked by the use of billions of embedded devices

Embedded Operating Systems

- There are two general approaches to developing an embedded operating system (OS):
 - Take an existing OS and adapt it for the embedded application
 - Design and implement an OS intended solely for embedded use

Application Processors versus Dedicated Processors

- Application processors
 - Defined by the processor's ability to execute complex operating systems
 - General-purpose in nature
 - An example is the smartphone – the embedded system is designed to support numerous apps and perform a wide variety of functions
- Dedicated processor
 - Is dedicated to one or a small number of specific tasks required by the host device
 - Because such an embedded system is dedicated to a specific task or tasks, the processor and associated components can be engineered to reduce size and cost

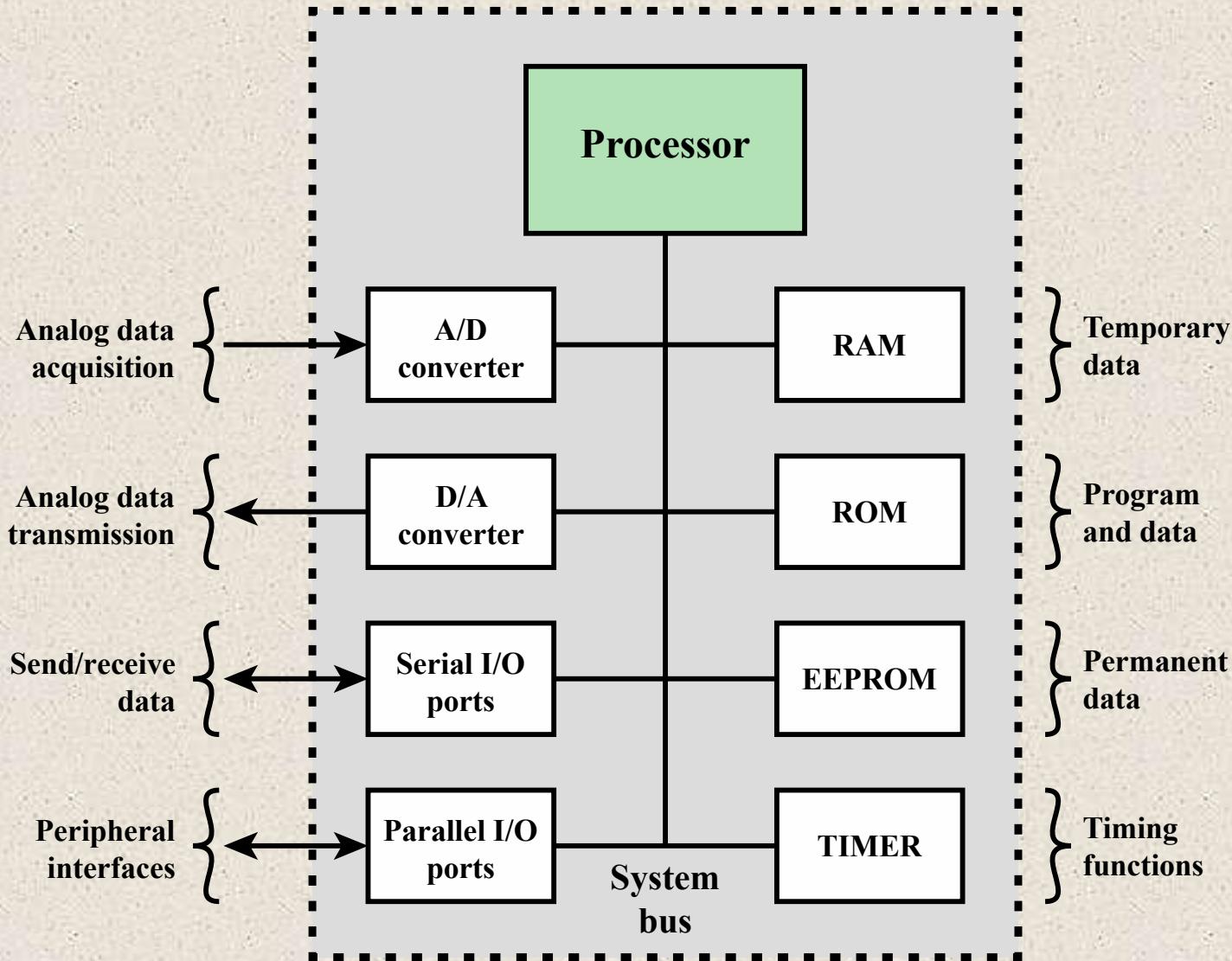


Figure 1.15 Typical Microcontroller Chip Elements



Deeply Embedded Systems

- Subset of embedded systems
- Has a processor whose behavior is difficult to observe both by the programmer and the user
- Uses a microcontroller rather than a microprocessor
- Is not programmable once the program logic for the device has been burned into ROM
- Has no interaction with a user
- Dedicated, single-purpose devices that detect something in the environment, perform a basic level of processing, and then do something with the results
- Often have wireless capability and appear in networked configurations, such as networks of sensors deployed over a large area
- Typically have extreme resource constraints in terms of memory, processor size, time, and power consumption

Refers to a processor architecture that has evolved from RISC design principles and is used in embedded systems

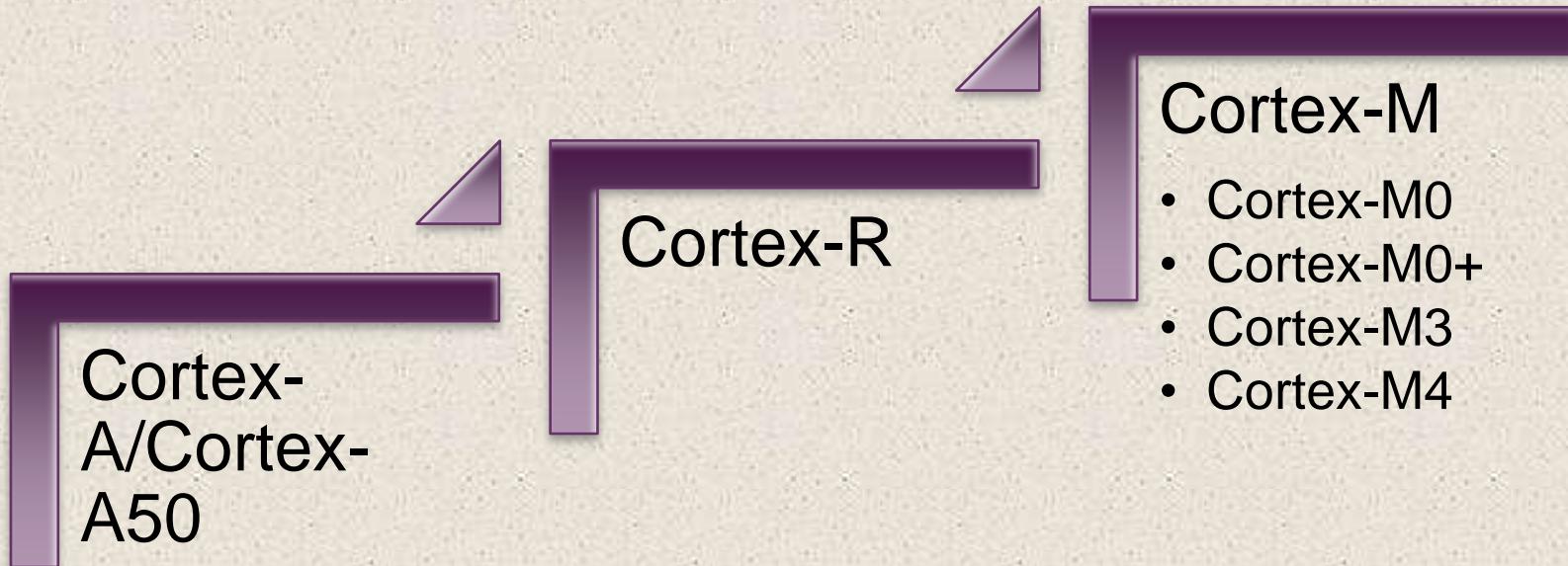
Family of RISC-based microprocessors and microcontrollers designed by ARM Holdings, Cambridge, England

Chips are high-speed processors that are known for their small die size and low power requirements

Probably the most widely used embedded processor architecture and indeed the most widely used processor architecture of any kind in the world

Acorn RISC Machine/Advanced RISC Machine

ARM Products



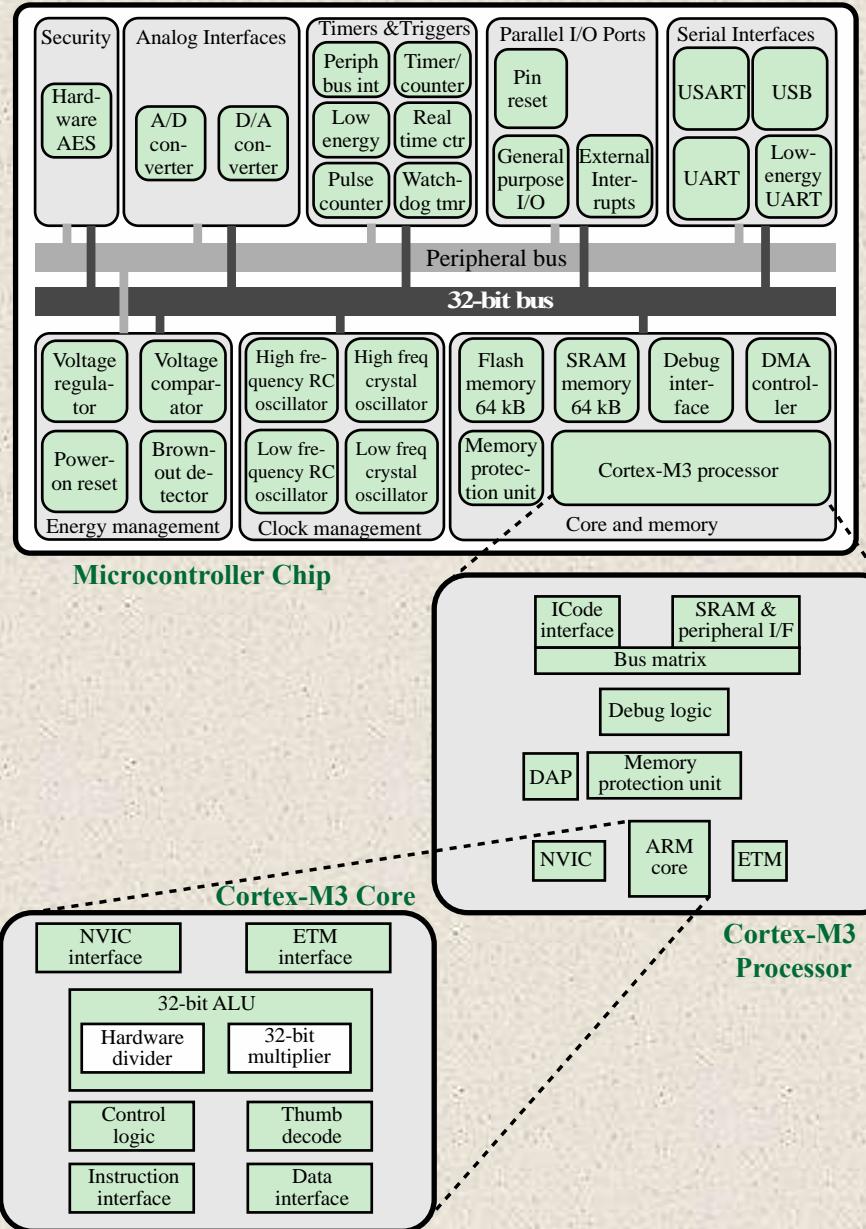


Figure 1.16 Typical Microcontroller Chip Based on Cortex-M3

Cloud Computing



- NIST defines cloud computing as:

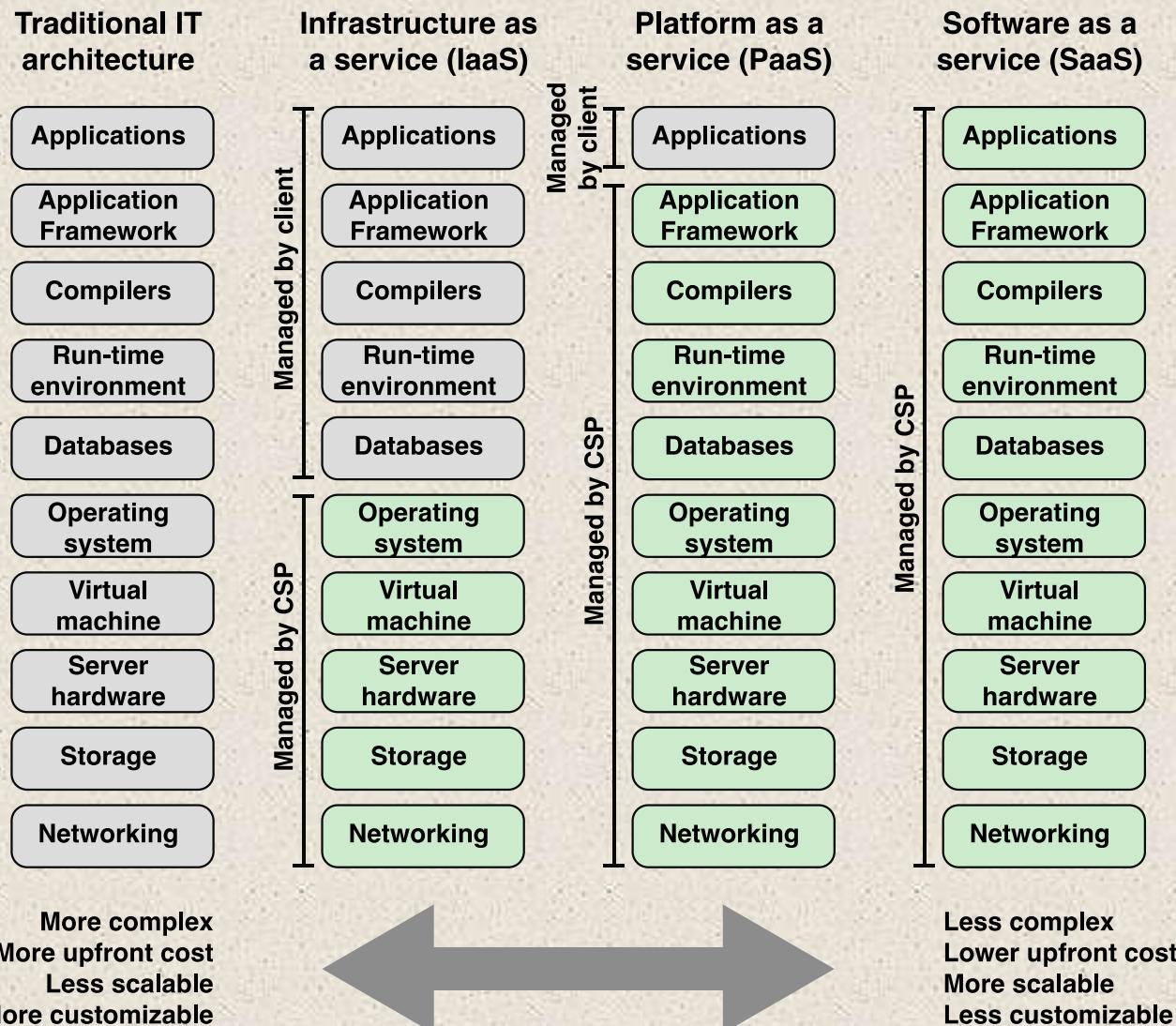
“A model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction.”
- You get economies of scale, professional network management, and professional security management
- The individual or company only needs to pay for the storage capacity and services they need
- Cloud provider takes care of security

Cloud Networking

- Refers to the networks and network management functionality that must be in place to enable cloud computing
- One example is the provisioning of high-performance and/or high-reliability networking between the provider and subscriber
- The collection of network capabilities required to access a cloud, including making use of specialized services over the Internet, linking enterprise data center to a cloud, and using firewalls and other network security devices at critical points to enforce access security policies

Cloud Storage

- Subset of cloud computing
- Consists of database storage and database applications hosted remotely on cloud servers
- Enables small businesses and individual users to take advantage of data storage that scales with their needs and to take advantage of a variety of database applications without having to buy, maintain, and manage the storage assets



IT = information technology
CSP = cloud service provider

Figure 1.17 Alternative Information Technology Architectures

+ Summary

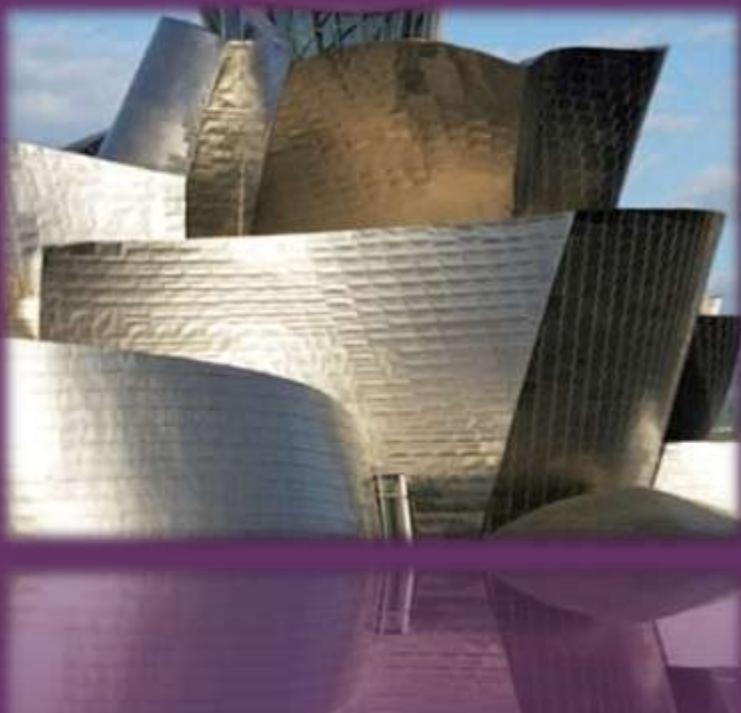
Chapter 1

- Organization and architecture
- Structure and function
- Brief history of computers
 - The First Generation: Vacuum tubes
 - The Second Generation: Transistors
 - The Third Generation: Integrated Circuits
 - Later generations
- The evolution of the Intel x86 architecture
- Cloud computing
 - Basic concepts
 - Cloud services

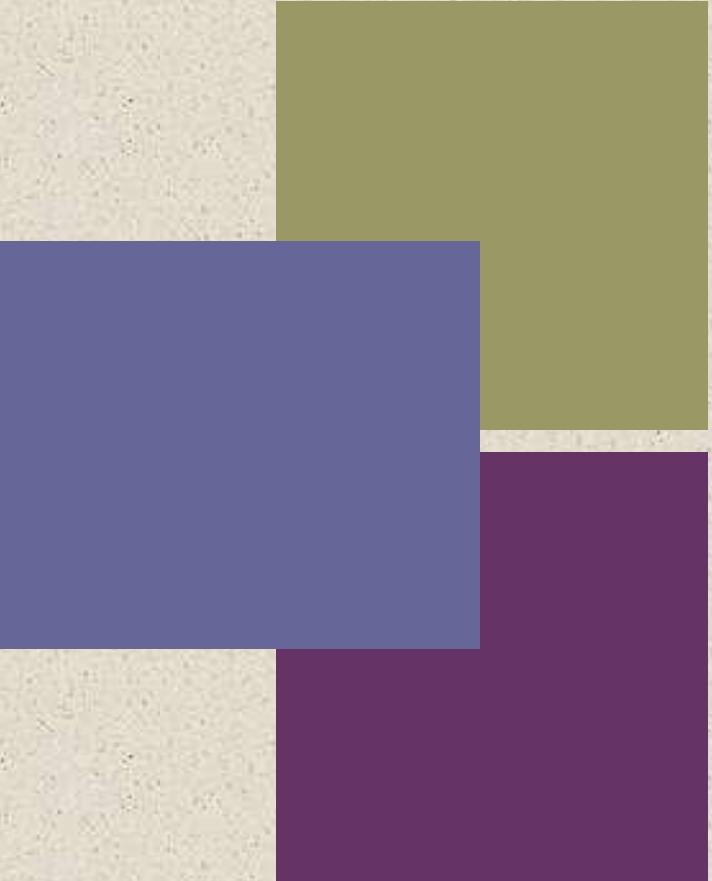
Basic Concepts and Computer Evolution

- Embedded systems
 - The Internet of things
 - Embedded operating systems
 - Application processors versus dedicated processors
 - Microprocessors versus microcontrollers
 - Embedded versus deeply embedded systems
- ARM architecture
 - ARM evolution
 - Instruction set architecture
 - ARM products

+



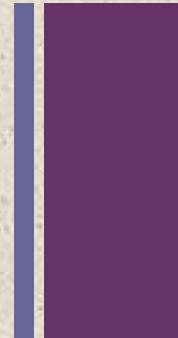
William Stallings
Computer Organization
and Architecture
10th Edition



+ Chapter 3

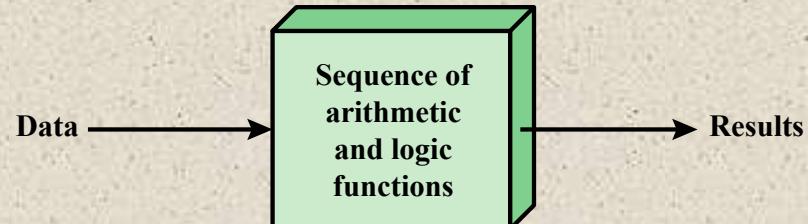
A Top-Level View of Computer Function and Interconnection

Computer Components

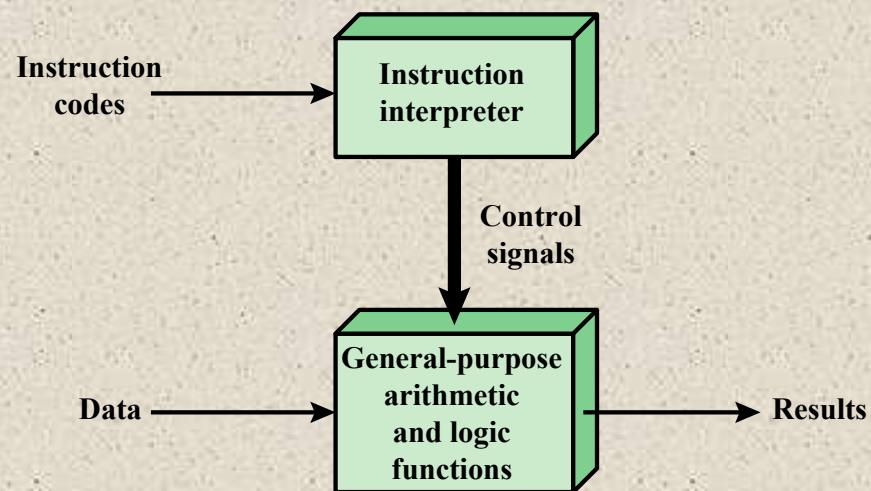


- Contemporary computer designs are based on concepts developed by John von Neumann at the Institute for Advanced Studies, Princeton
- Referred to as the *von Neumann architecture* and is based on three key concepts:
 - Data and instructions are stored in a single read-write memory
 - The contents of this memory are addressable by location, without regard to the type of data contained there
 - Execution occurs in a sequential fashion (unless explicitly modified) from one instruction to the next
- *Hardwired program*
 - The result of the process of connecting the various components in the desired configuration

Hardware and Software Approaches



(a) Programming in hardware



(b) Programming in software

Figure 3.1 Hardware and Software Approaches

Software

- A sequence of codes or instructions
- Part of the hardware interprets each instruction and generates control signals
- Provide a new sequence of codes for each new program instead of rewiring the hardware

Major components:

- CPU
 - Instruction interpreter
 - Module of general-purpose arithmetic and logic functions
- I/O Components
 - Input module
 - Contains basic components for accepting data and instructions and converting them into an internal form of signals usable by the system
 - Output module
 - Means of reporting results

Software

I/O
Components



MEMORY

Memory address register (MAR)

- Specifies the address in memory for the next read or write

Memory buffer register (MBR)

- Contains the data to be written into memory or receives the data read from memory

I/O address register (I/OAR)

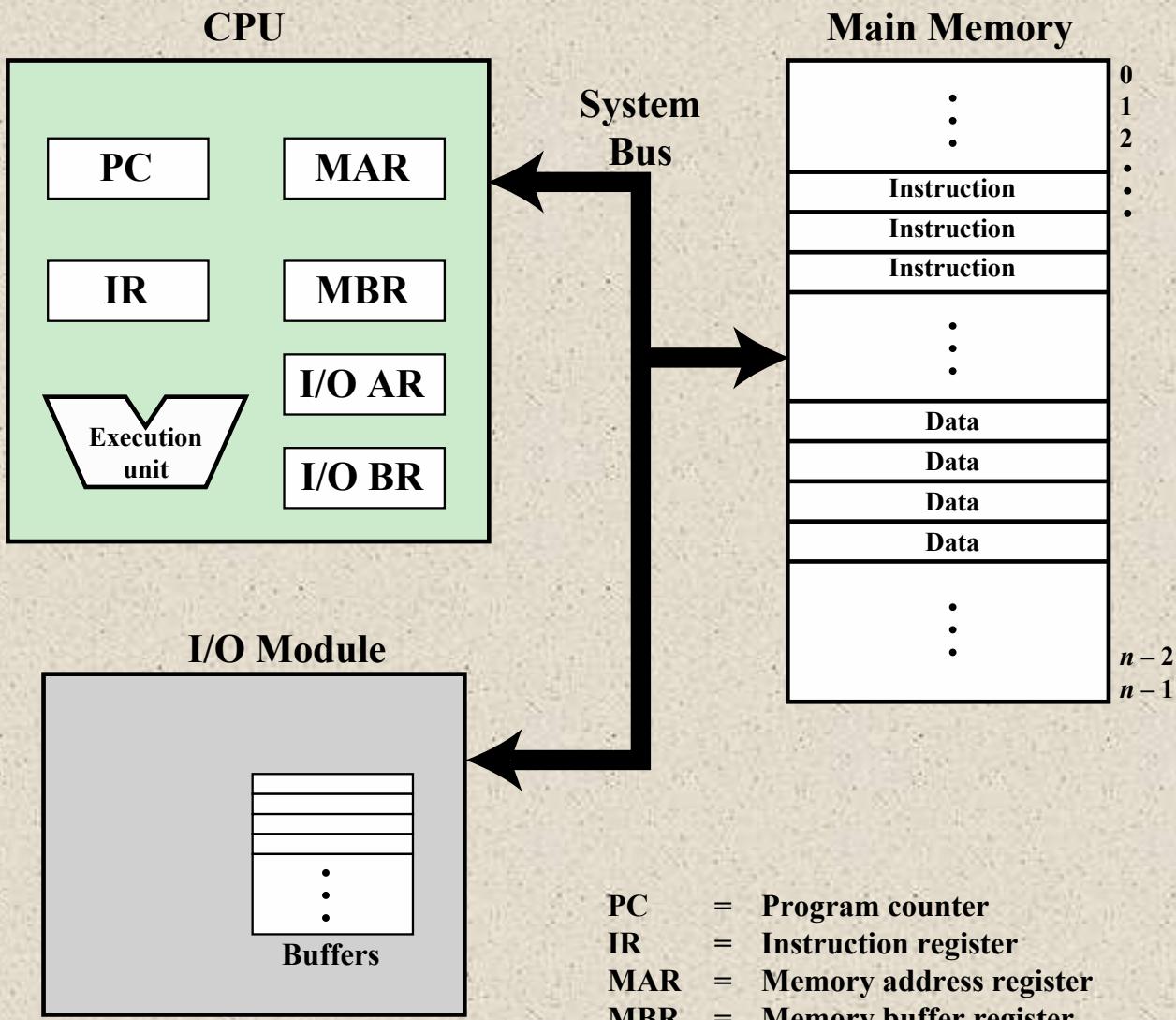
- Specifies a particular I/O device

I/O buffer register (I/OBR)

- Used for the exchange of data between an I/O module and the CPU

MAR

MBR



PC	=	Program counter
IR	=	Instruction register
MAR	=	Memory address register
MBR	=	Memory buffer register
I/O AR	=	Input/output address register
I/O BR	=	Input/output buffer register

Figure 3.2 Computer Components: Top-Level View

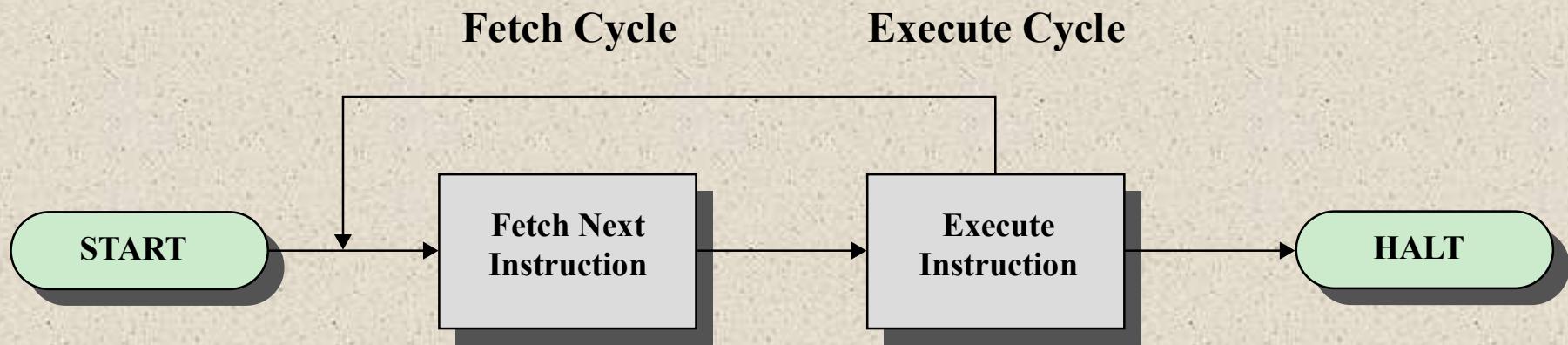
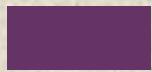


Figure 3.3 Basic Instruction Cycle



Fetch Cycle

Ambil Siklus

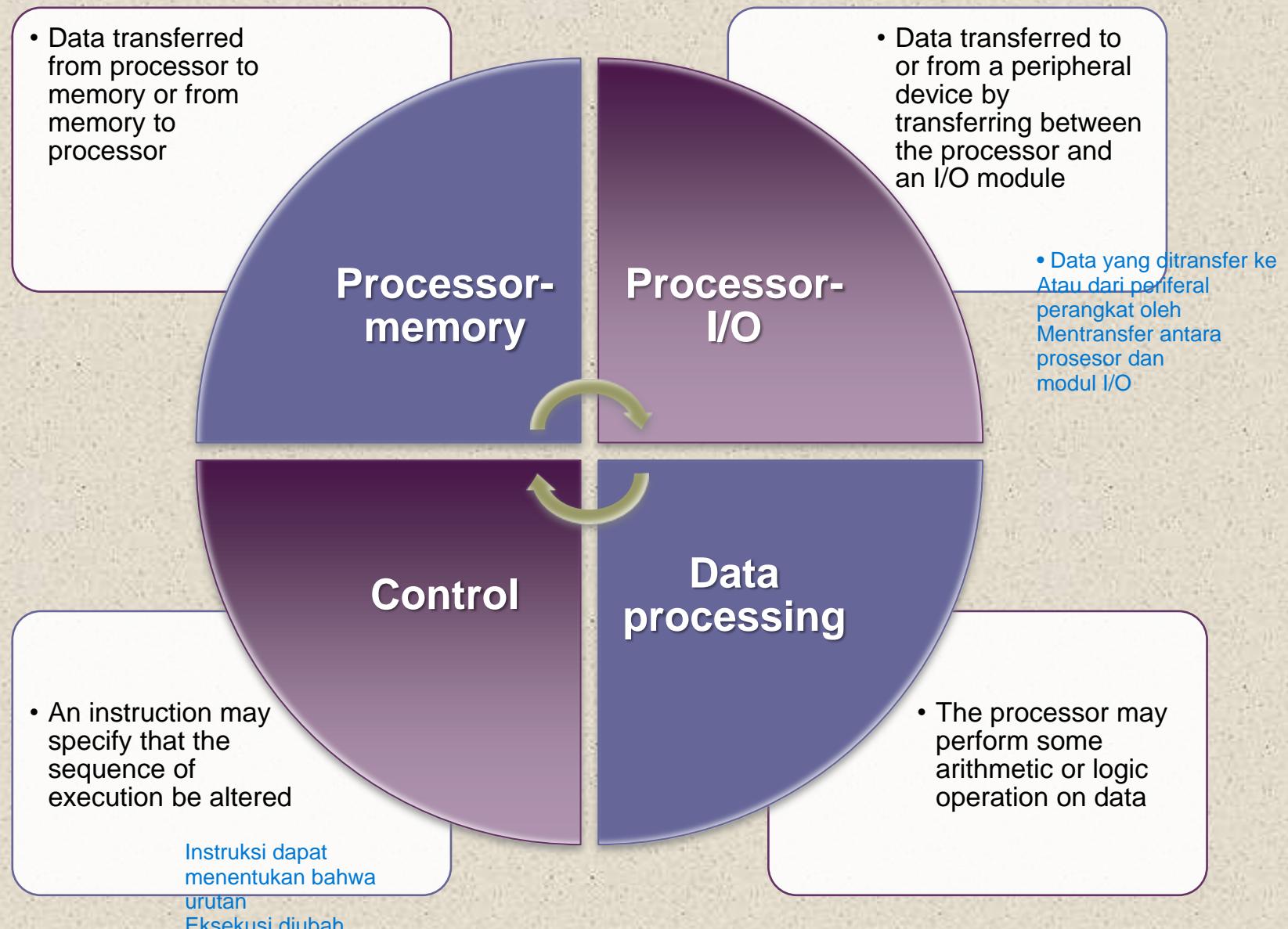
- At the beginning of each instruction cycle the processor fetches an instruction from memory
- The program counter (PC) holds the address of the instruction to be fetched next
- The processor increments the PC after each instruction fetch so that it will fetch the next instruction in sequence
- The fetched instruction is loaded into the instruction register (IR)
- The processor interprets the instruction and performs the required action

Prosesor meningkatkan PC setelah setiap instruksi mengambil sehingga akan mengambil instruksi berikutnya secara berurutan

Prosesor menafsirkan instruksi dan melakukan tindakan yang diperlukan



Action Categories



0	3 4	15
Opcode		Address

(a) Instruction format

0	1	15
S		Magnitude

(b) Integer format

Program Counter (PC) = Address of instruction

Instruction Register (IR) = Instruction being executed

Accumulator (AC) = Temporary storage

(c) Internal CPU registers

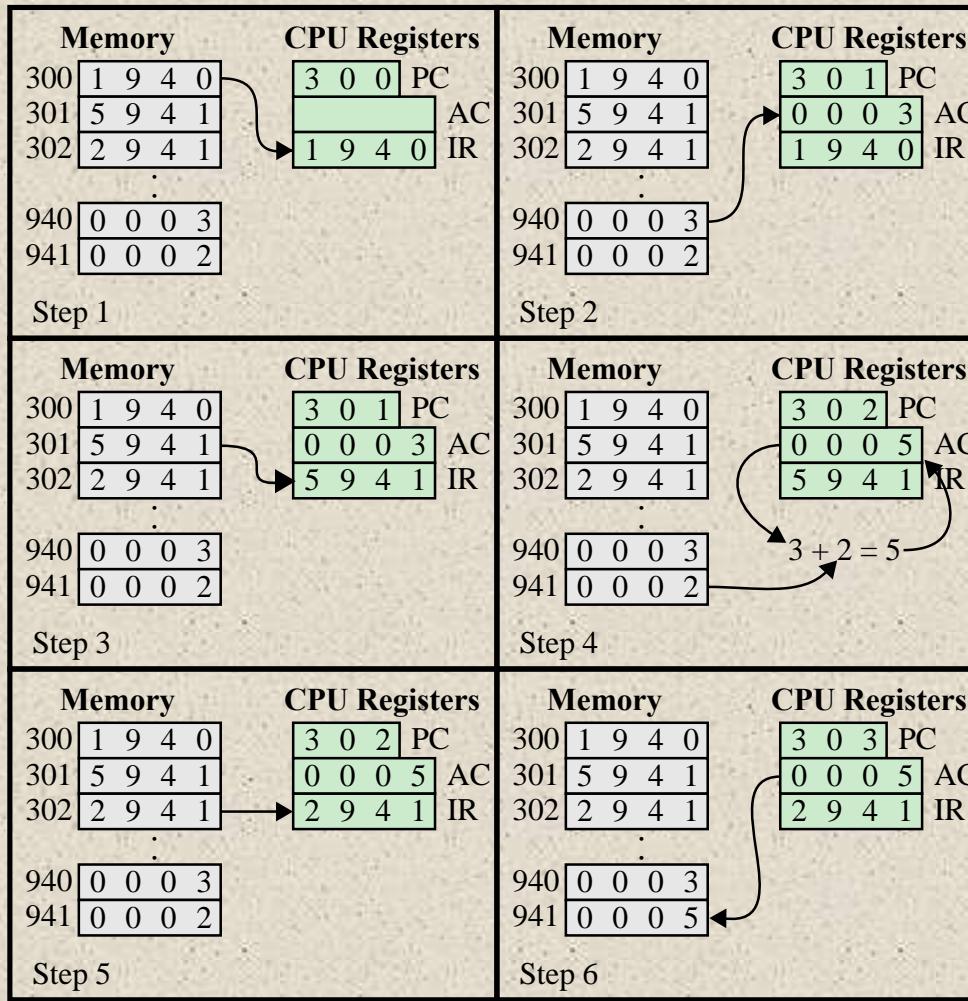
0001 = Load AC from Memory

0010 = Store AC to Memory

0101 = Add to AC from Memory

(d) Partial list of opcodes

Figure 3.4 Characteristics of a Hypothetical Machine



**Figure 3.5 Example of Program Execution
(contents of memory and registers in hexadecimal)**

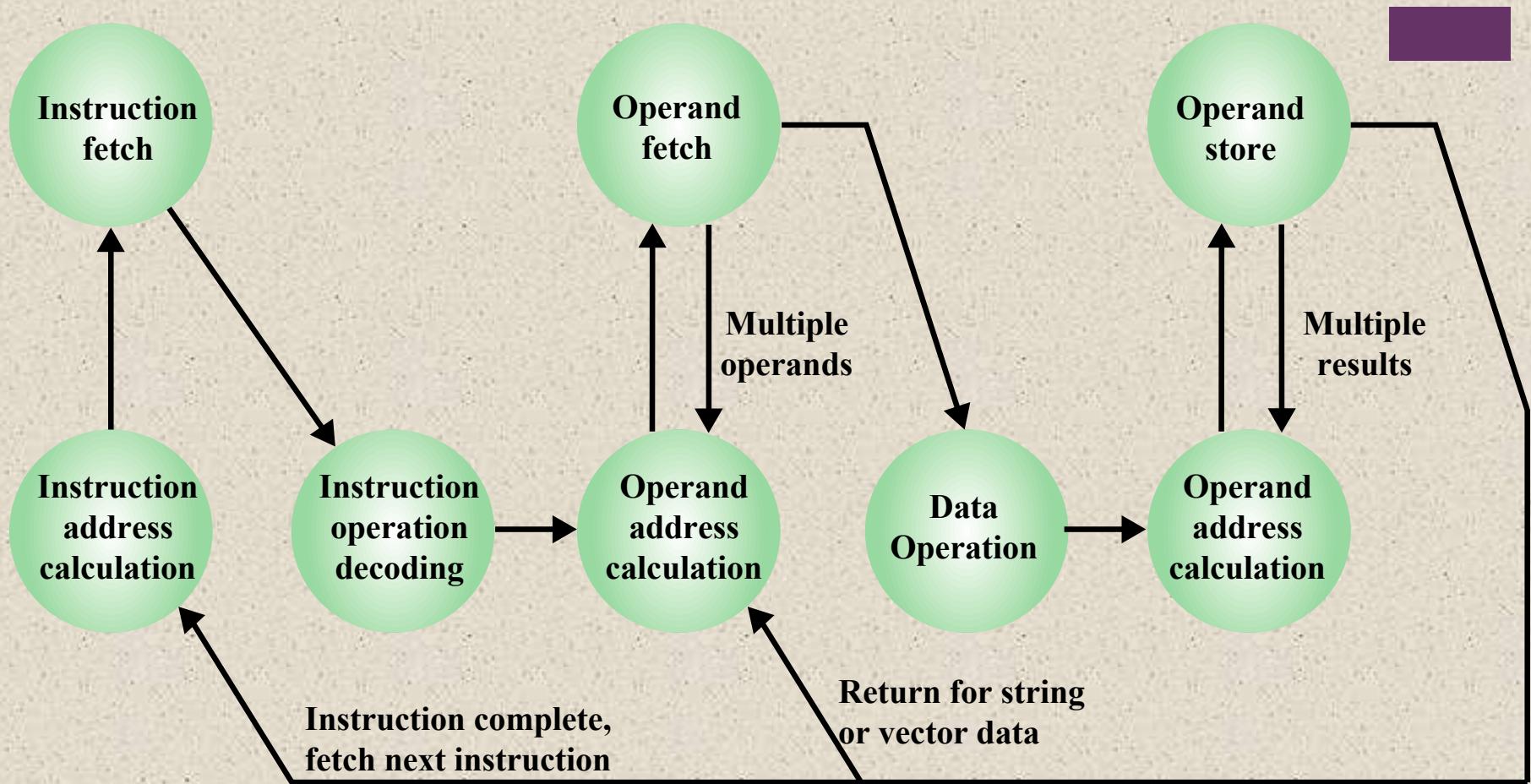


Figure 3.6 Instruction Cycle State Diagram

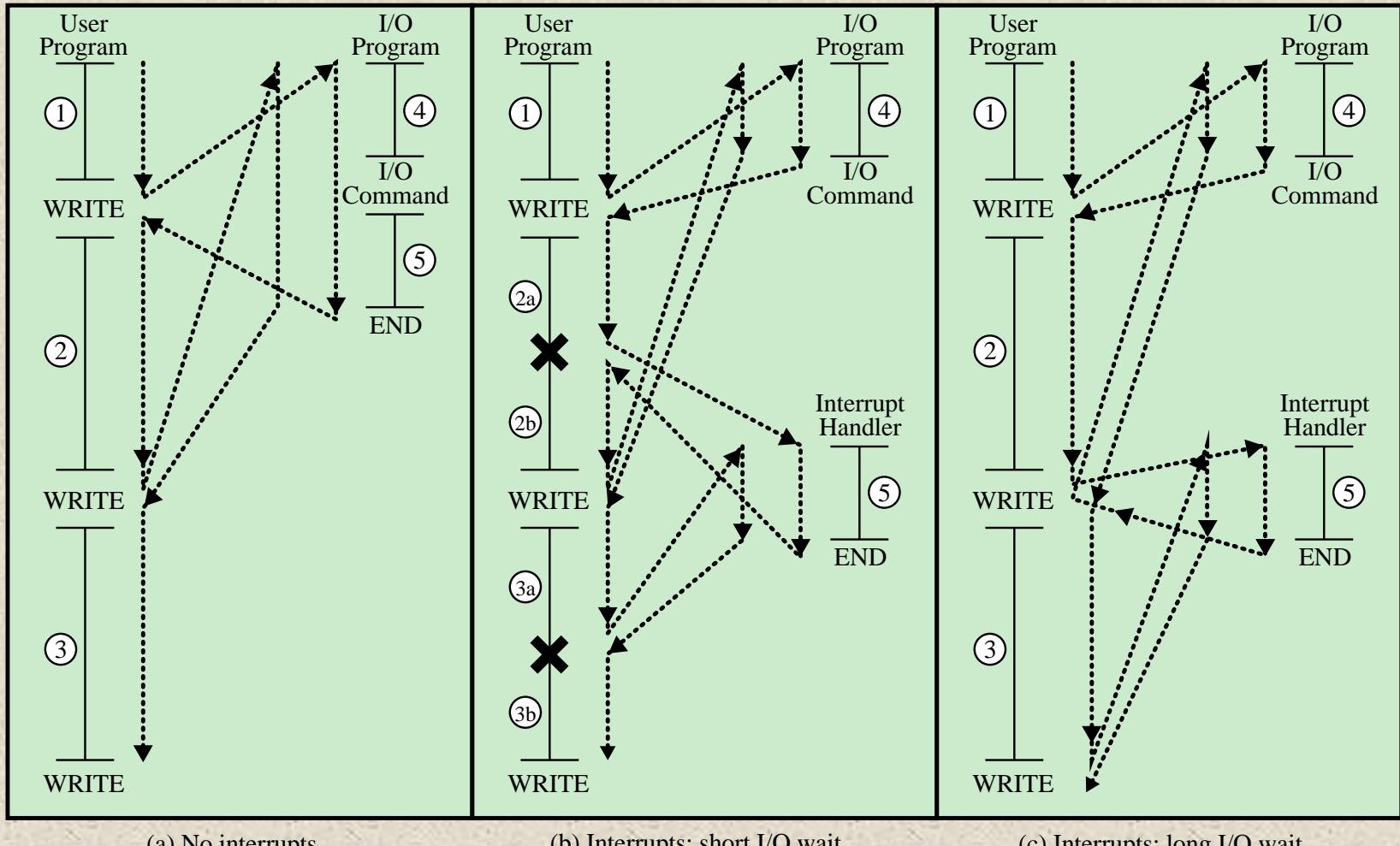


Program	Generated by some condition that occurs as a result of an instruction execution, such as arithmetic overflow, division by zero, attempt to execute an illegal machine instruction, or reference outside a user's allowed memory space.
Timer	Generated by a timer within the processor. This allows the operating system to perform certain functions on a regular basis.
I/O	Generated by an I/O controller, to signal normal completion of an operation, request service from the processor, or to signal a variety of error conditions.
Hardware failure	Generated by a failure such as power failure or memory parity error.

Dihasilkan oleh kegagalan seperti kegagalan daya atau kesalahan paritas memori

Table 3.1

Classes of Interrupts



X = interrupt occurs during course of execution of user program

Figure 3.7 Program Flow of Control Without and With Interrupts

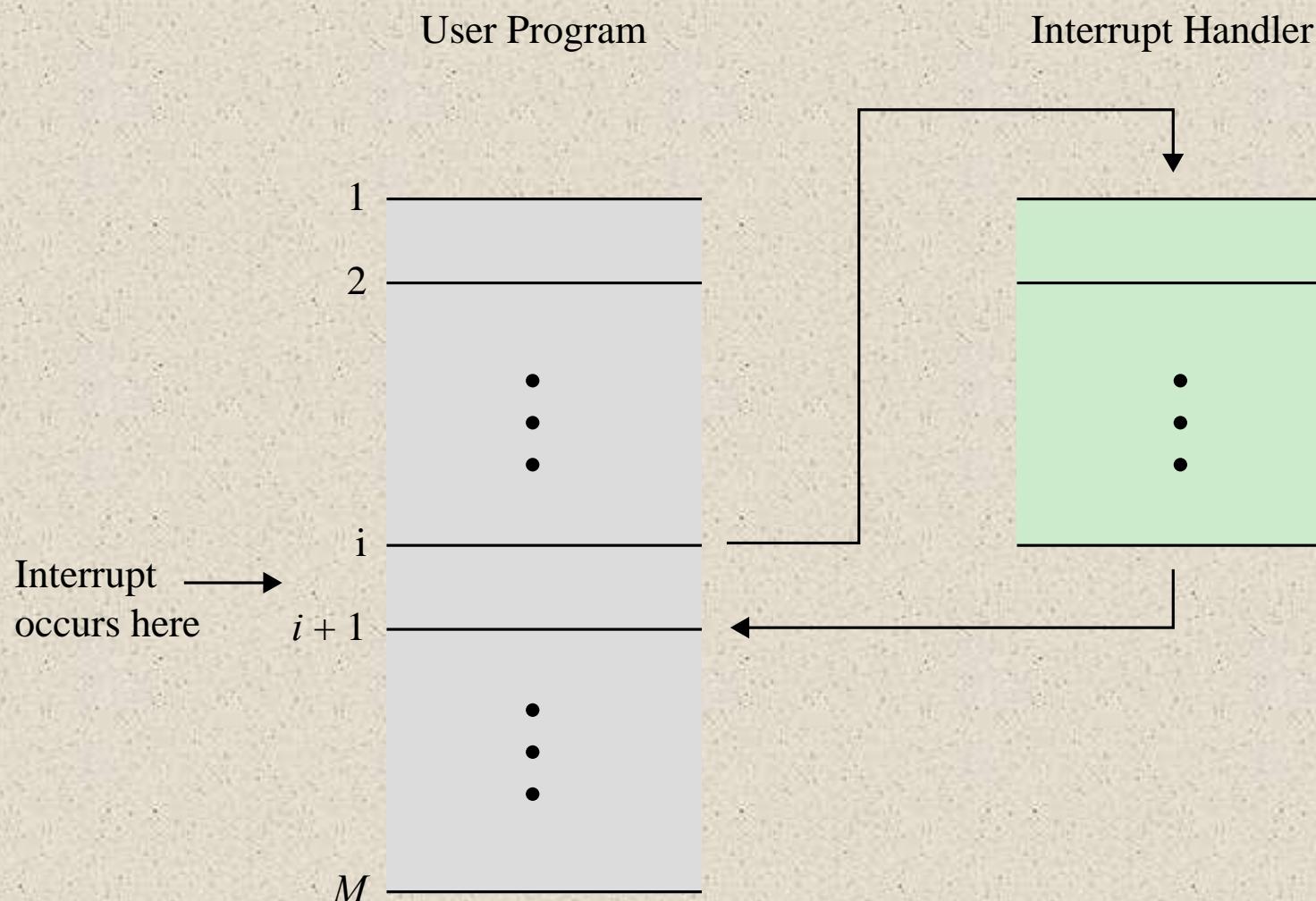


Figure 3.8 Transfer of Control via Interrupts

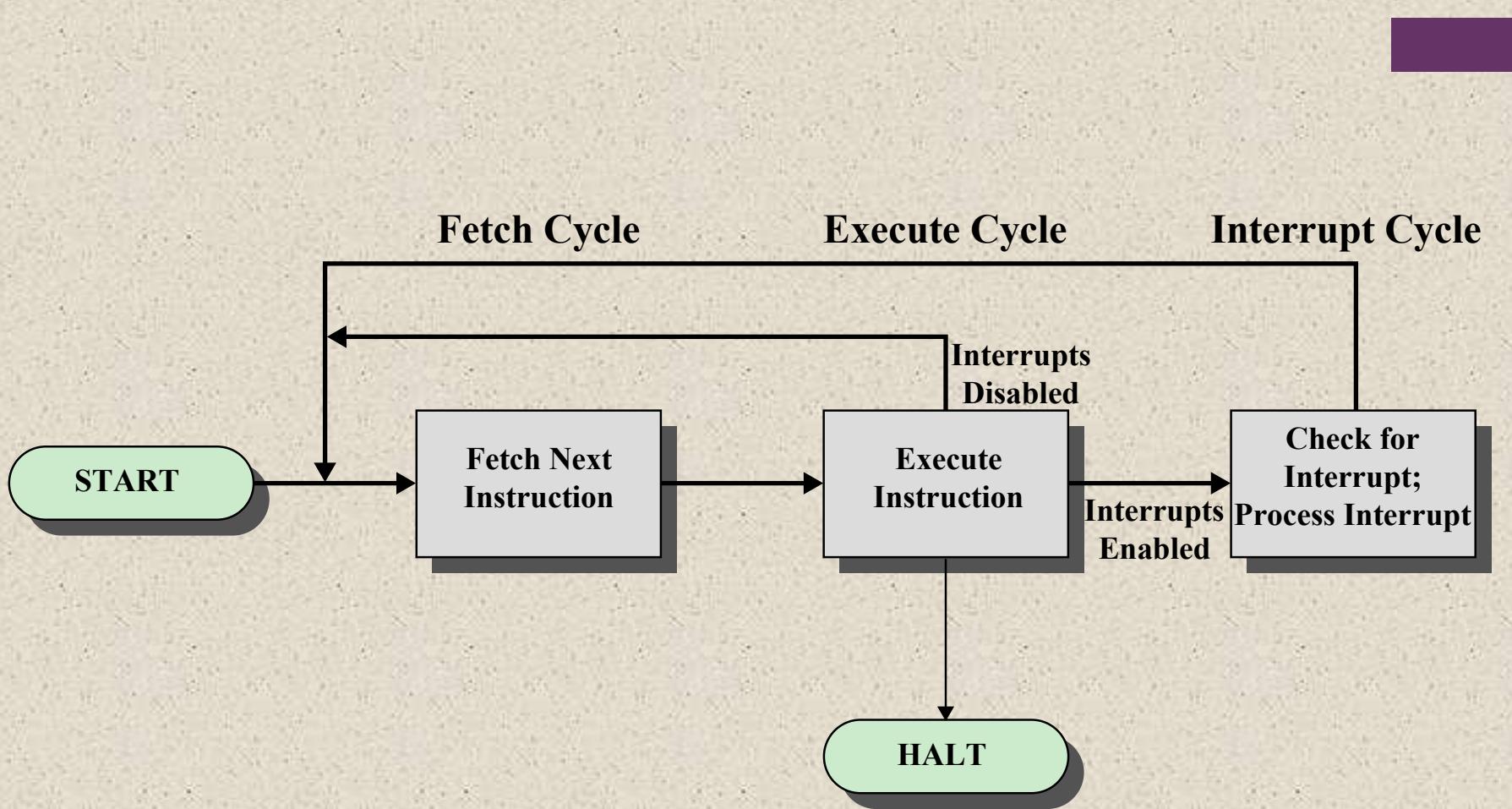


Figure 3.9 Instruction Cycle with Interrupts

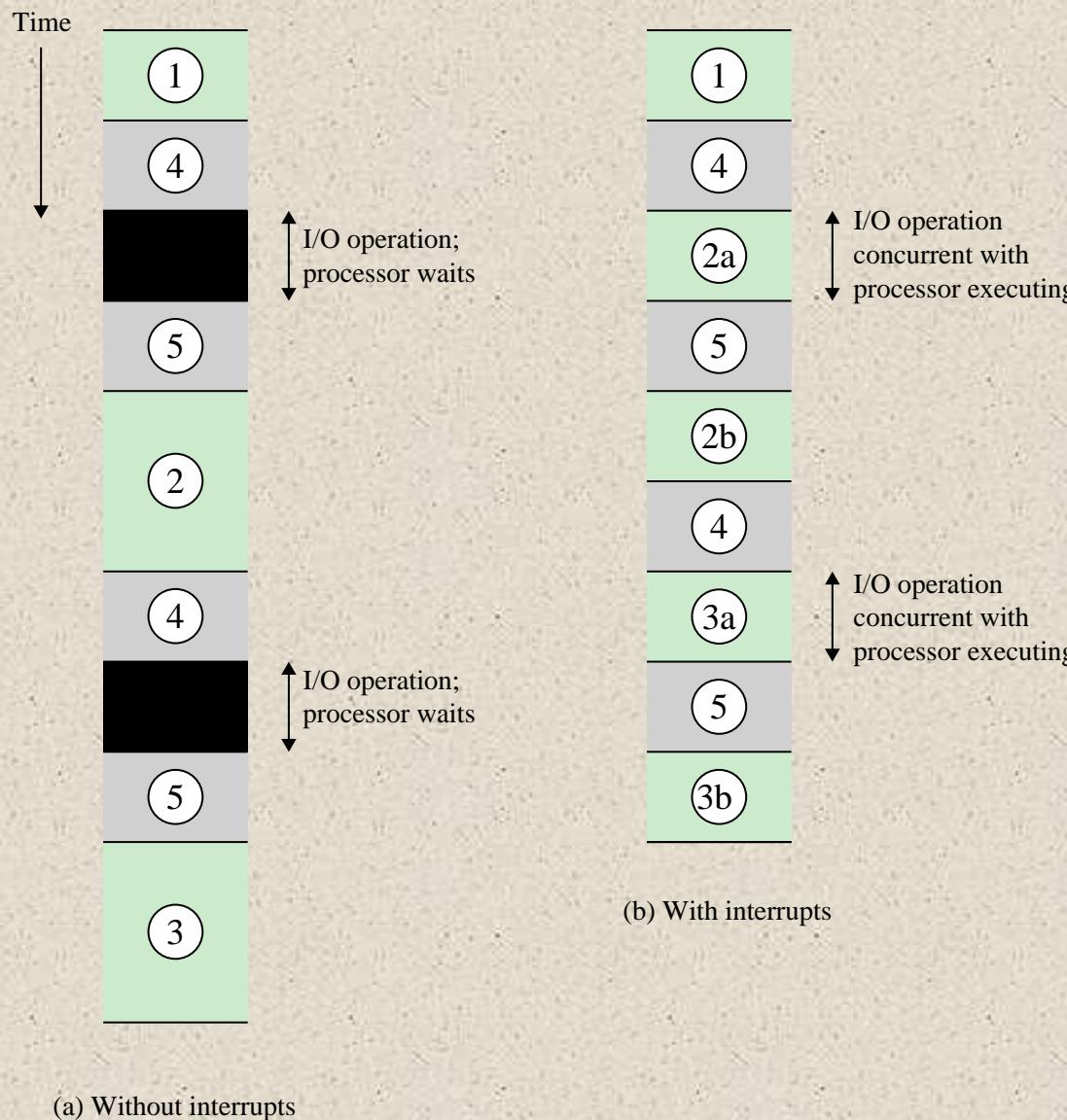


Figure 3.10 Program Timing: Short I/O Wait

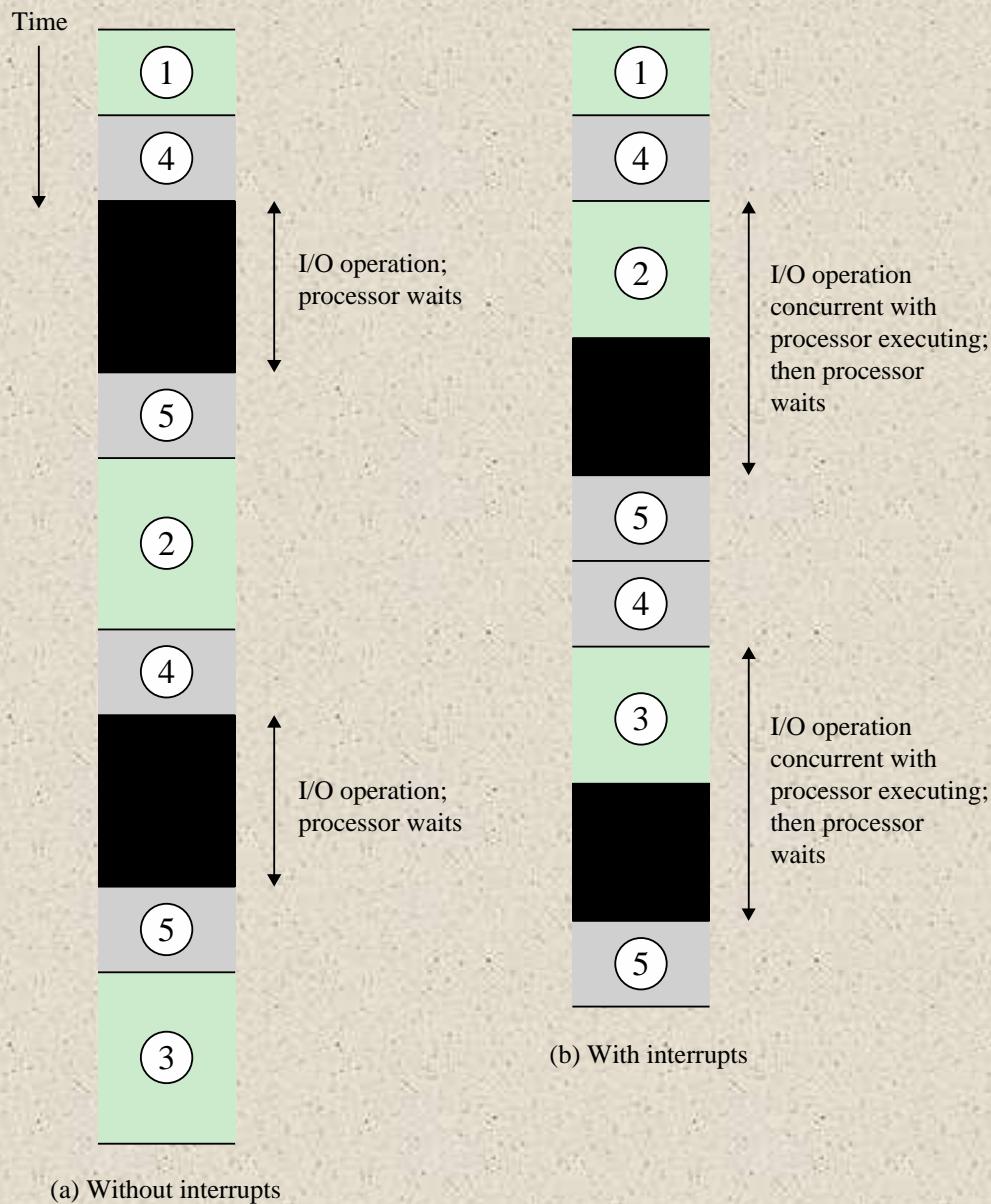


Figure 3.11 Program Timing: Long I/O Wait

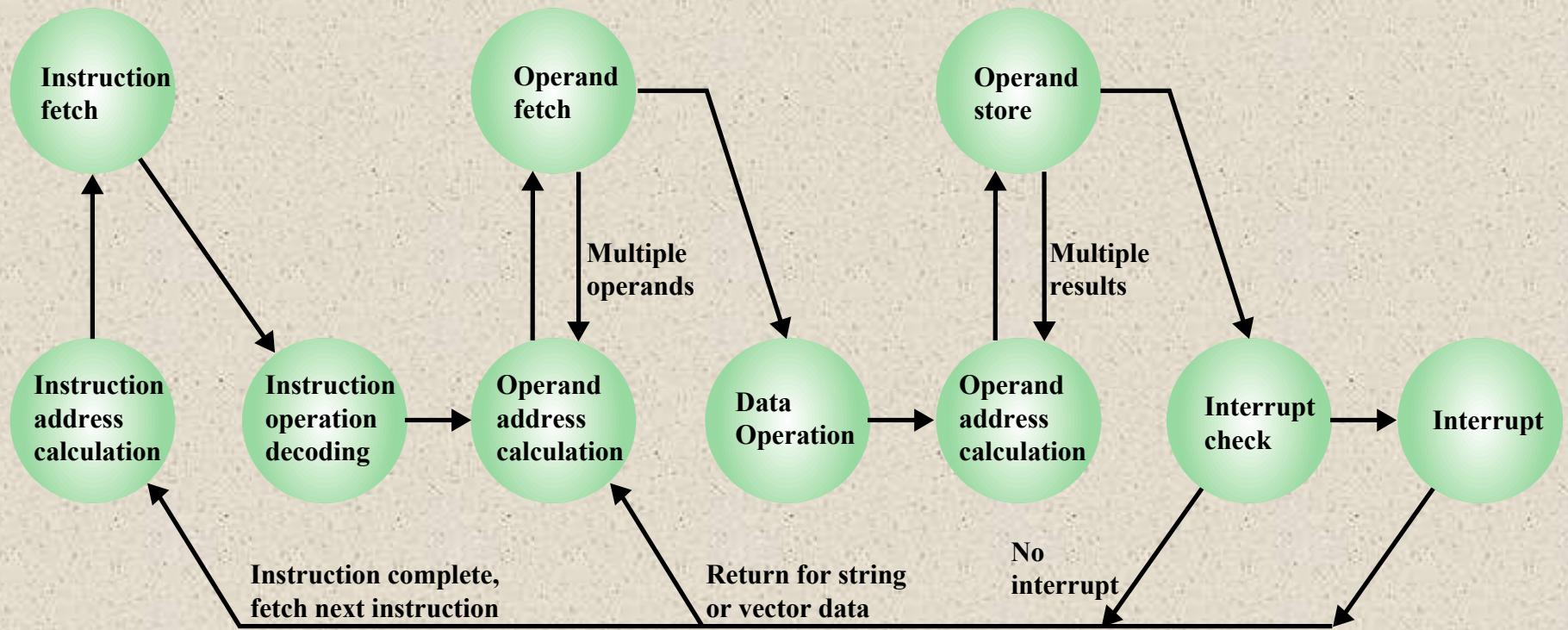
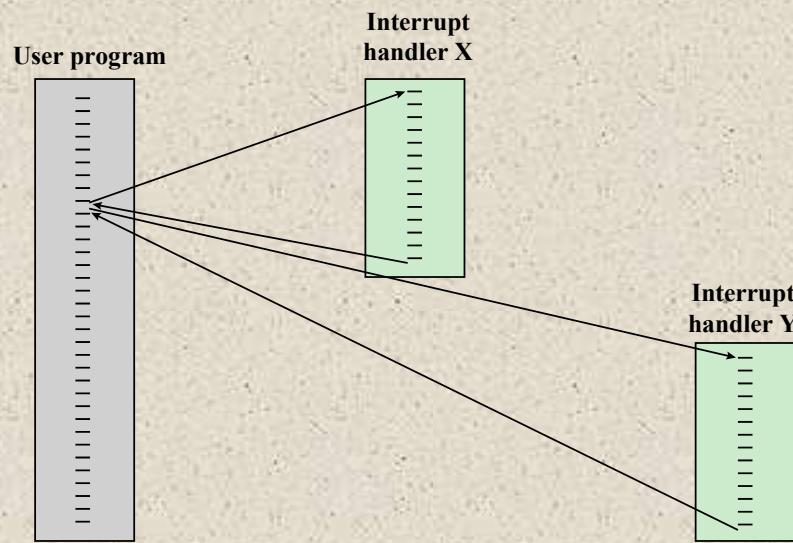
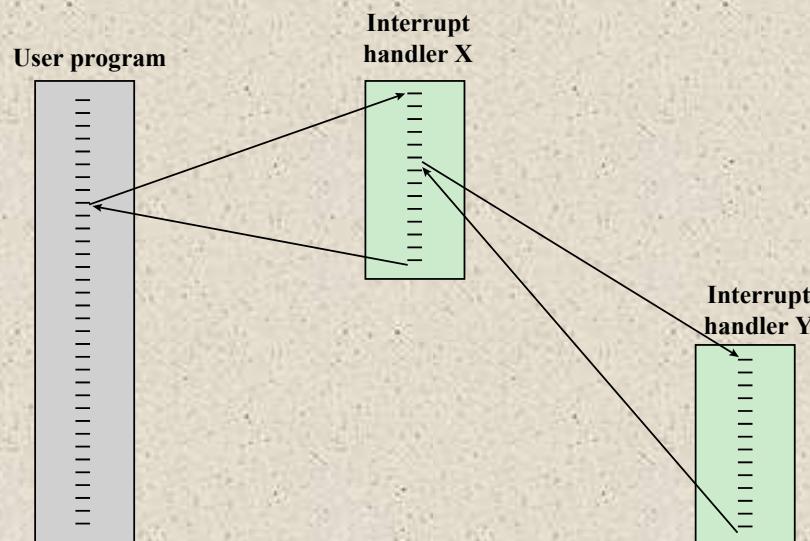


Figure 3.12 Instruction Cycle State Diagram, With Interrupts



(a) Sequential interrupt processing



(b) Nested interrupt processing

Figure 3.13 Transfer of Control with Multiple Interrupts

User program

Printer
interrupt service routine

Communication
interrupt service routine

Disk
interrupt service routine

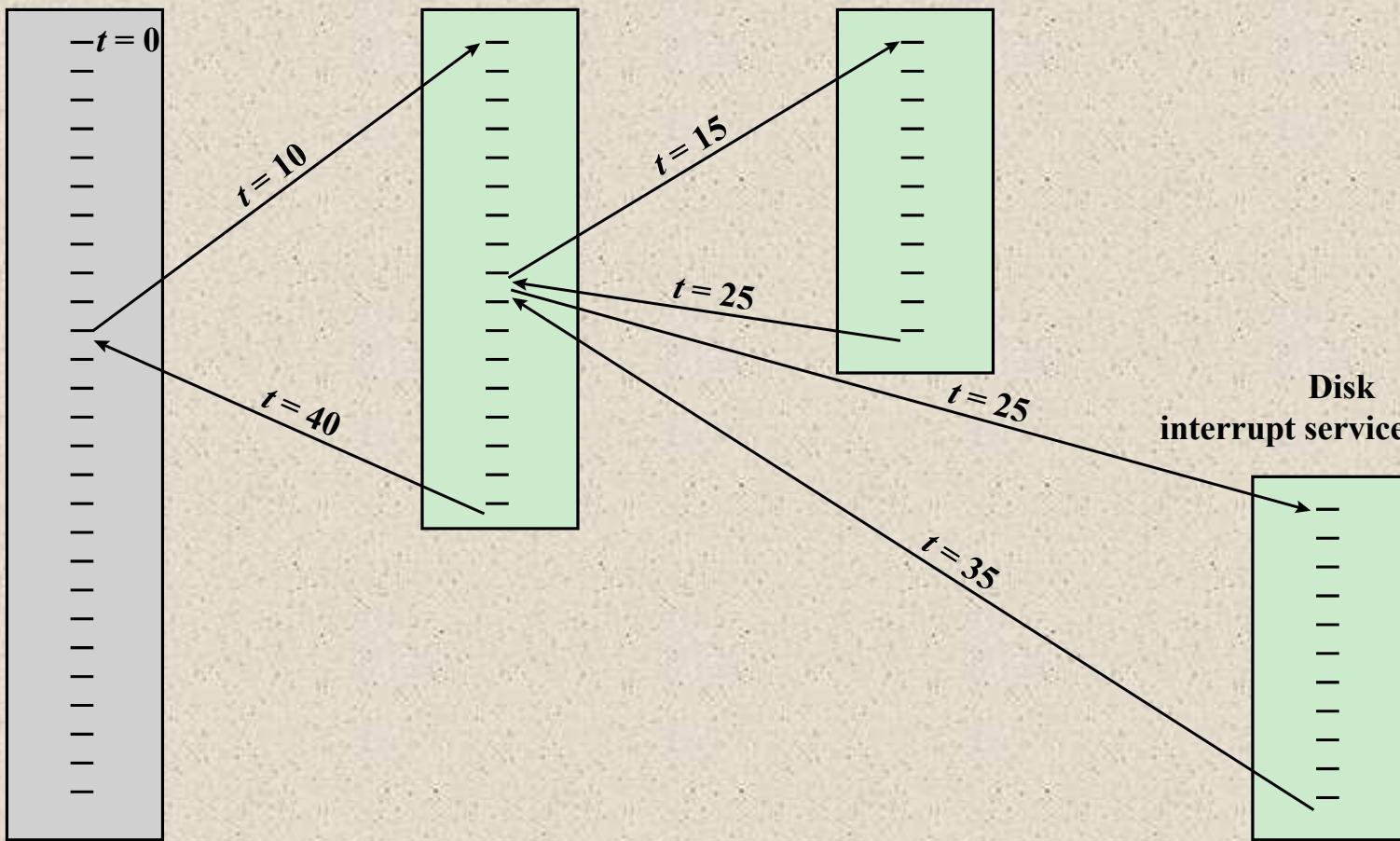


Figure 3.14 Example Time Sequence of Multiple Interrupts



I/O Function

- I/O module 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
- 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)

Modul I/O dapat bertukar data secara langsung dengan prosesor
Prosesor dapat membaca data dari atau menulis data ke modul I / O
Prosesor mengidentifikasi perangkat tertentu yang dikendalikan oleh modul I/O khusus
Instruksi I/O daripada instruksi referensi memori

Dalam beberapa kasus, diinginkan untuk memungkinkan pertukaran I / O terjadi.
langsung dengan memori
Prosesor memberikan kepada modul I / O otoritas untuk membaca dari atau menulis ke memori sehingga

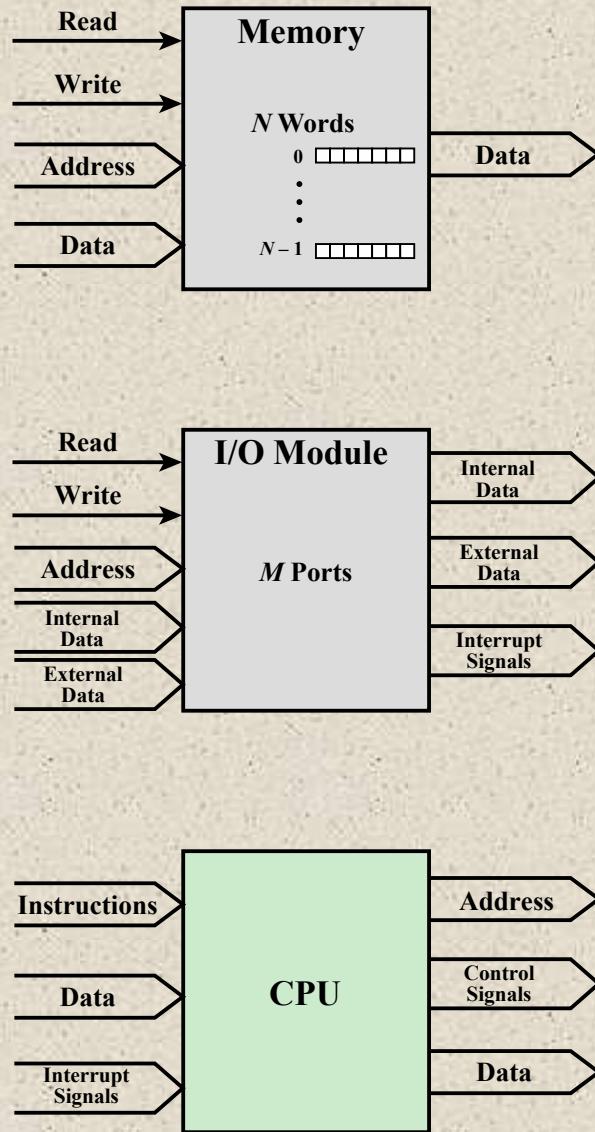


Figure 3.15 Computer Modules

The interconnection structure must support the following types of transfers:

Struktur interkoneksi harus mendukung Jenis transfer berikut:

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

A communication pathway connecting two or more devices

- Key characteristic is that it is a shared transmission medium

Signals transmitted by any one device are available for reception by all other devices attached to the bus

- If two devices transmit during the same time period their signals will overlap and become garbled



Typically consists of multiple communication lines

- Each line is capable of transmitting signals representing binary 1 and binary 0

Computer systems contain a number of different buses that provide pathways between components at various levels of the computer system hierarchy



System bus

- A bus that connects major computer components (processor, memory, I/O)

Sistem komputer berisi Sejumlah bus yang berbeda yang menyediakan jalur antara komponen di Berbagai tingkatan dari hierarki sistem komputer

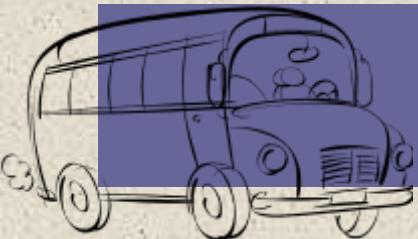
The most common computer interconnection structures are based on the use of one or more system buses

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 number of lines determines how many bits can be transferred at a time
- The width of the data bus is a key factor in determining overall system performance

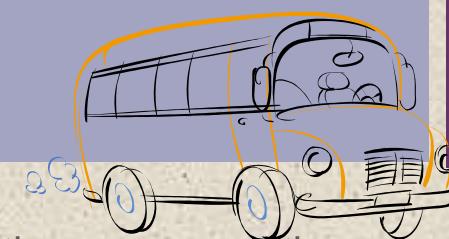


+ Address Bus



Digunakan untuk menunjuk sumber atau Tujuan dari data pada bus data
Jika prosesor ingin membaca Kata data dari memori itu Menempatkan alamat yang diinginkan kata pada baris alamat

Control Bus



- Used to designate the source or destination of the data on the data bus
- If the processor wishes to read a word 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
- 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
- Timing signals indicate the validity of data and address information
- Command signals specify operations to be performed

Lebar menentukan maksimum kemungkinan kapasitas memori dari sistem

Juga digunakan untuk mengatasi port I/O tingkat yang lebih tinggi digunakan

Digunakan untuk mengontrol akses dan penggunaan data dan baris alamat Karena data dan baris alamat dibagi oleh semua komponen di sana Harus menjadi sarana untuk mengendalikan pakai

Sinyal kontrol mengirimkan keduanya informasi perintah dan waktu di antara modul sistem

Sinyal waktu menunjukkan validitas informasi data dan alamat Sinyal perintah menentukan operasi untuk dilakukan

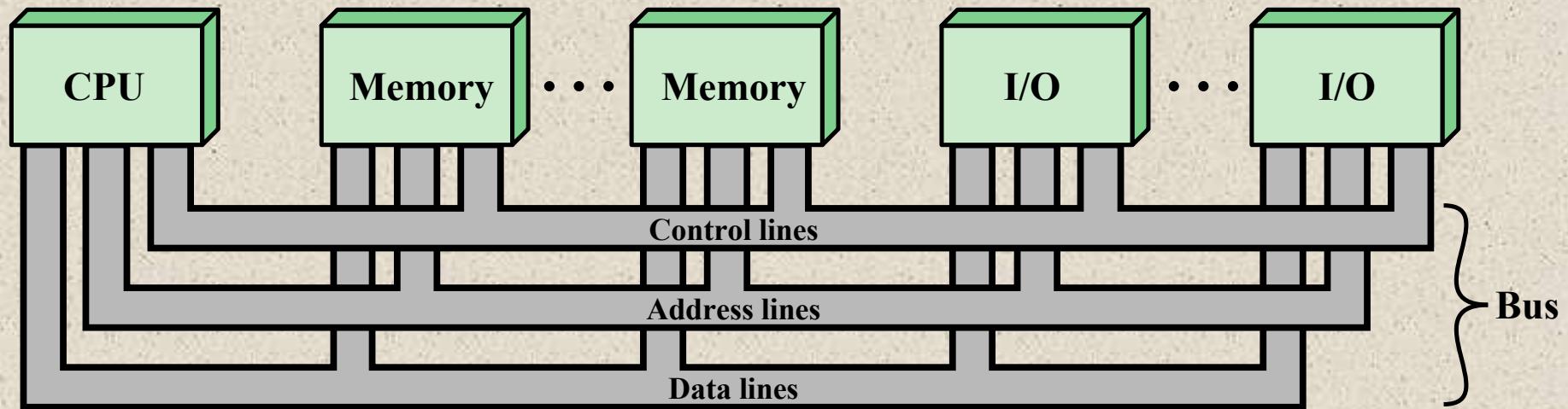


Figure 3.16 Bus Interconnection Scheme

Point-to-Point Interconnect

Principal reason for change was the electrical constraints encountered with increasing the frequency of wide synchronous buses

At higher and higher data rates it becomes increasingly difficult to perform the synchronization and arbitration functions in a timely fashion

A conventional shared bus on the same chip magnified the difficulties of increasing bus data rate and reducing bus latency to keep up with the processors

Has lower latency, higher data rate, and better scalability

+ Quick Path Interconnect

- Introduced in 2008
- Multiple direct connections
 - Direct pairwise connections to other components eliminating the need for arbitration found in shared transmission systems
- Layered protocol architecture
 - These processor level interconnects use a layered protocol architecture rather than the simple use of control signals found in shared bus arrangements
- Packetized data transfer
 - Data are sent as a sequence of packets each of which includes control headers and error control codes

QPI



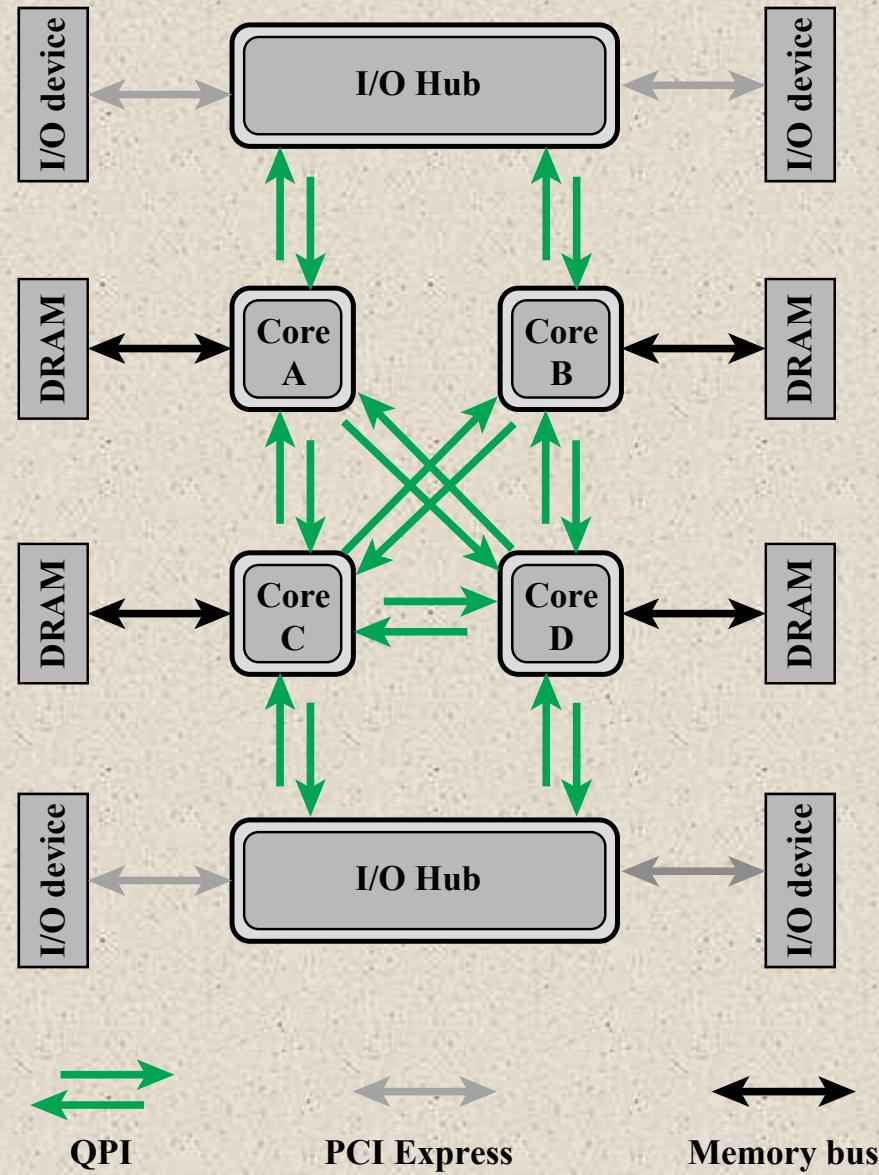


Figure 3.17 Multicore Configuration Using QPI

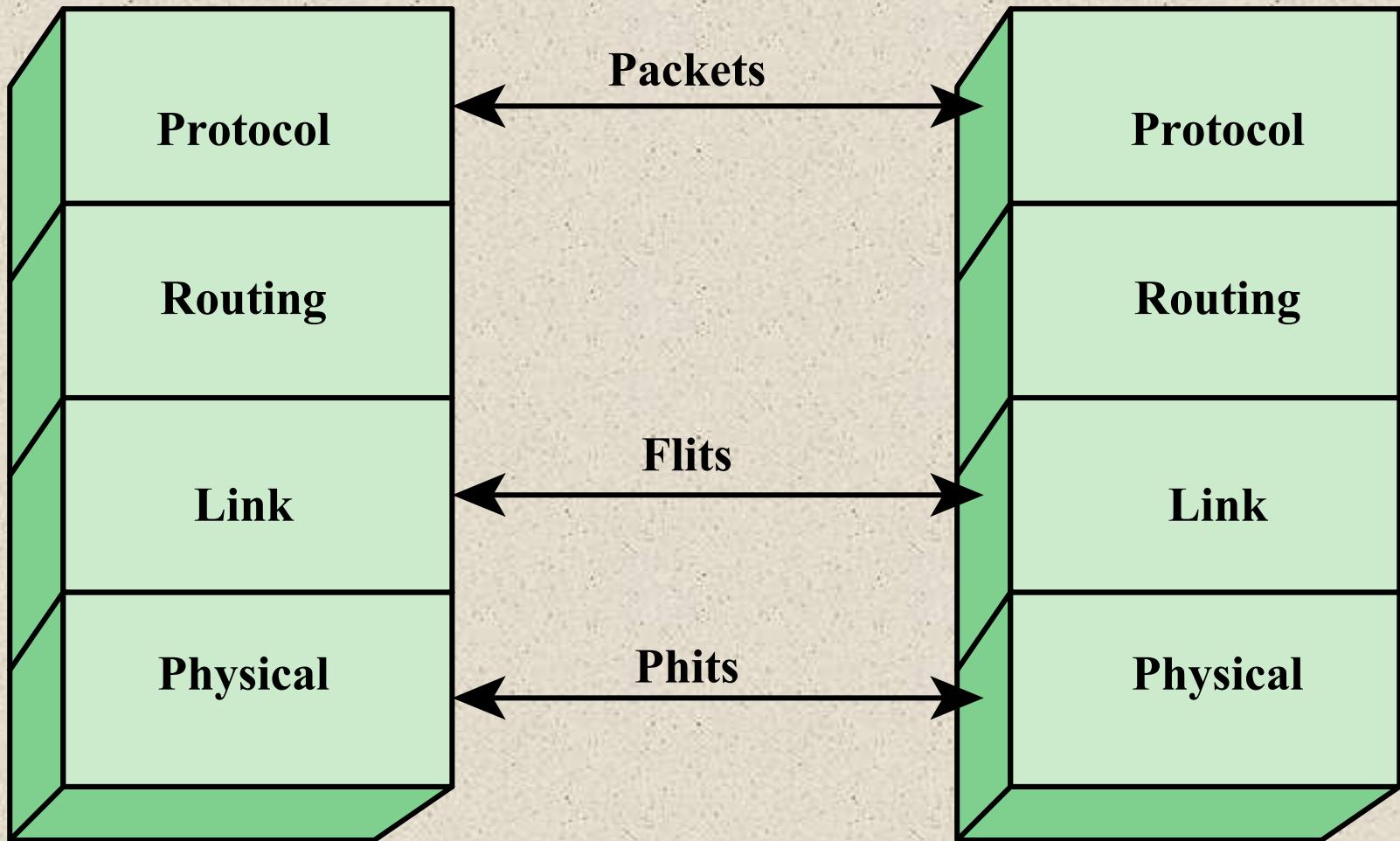


Figure 3.18 QPI Layers

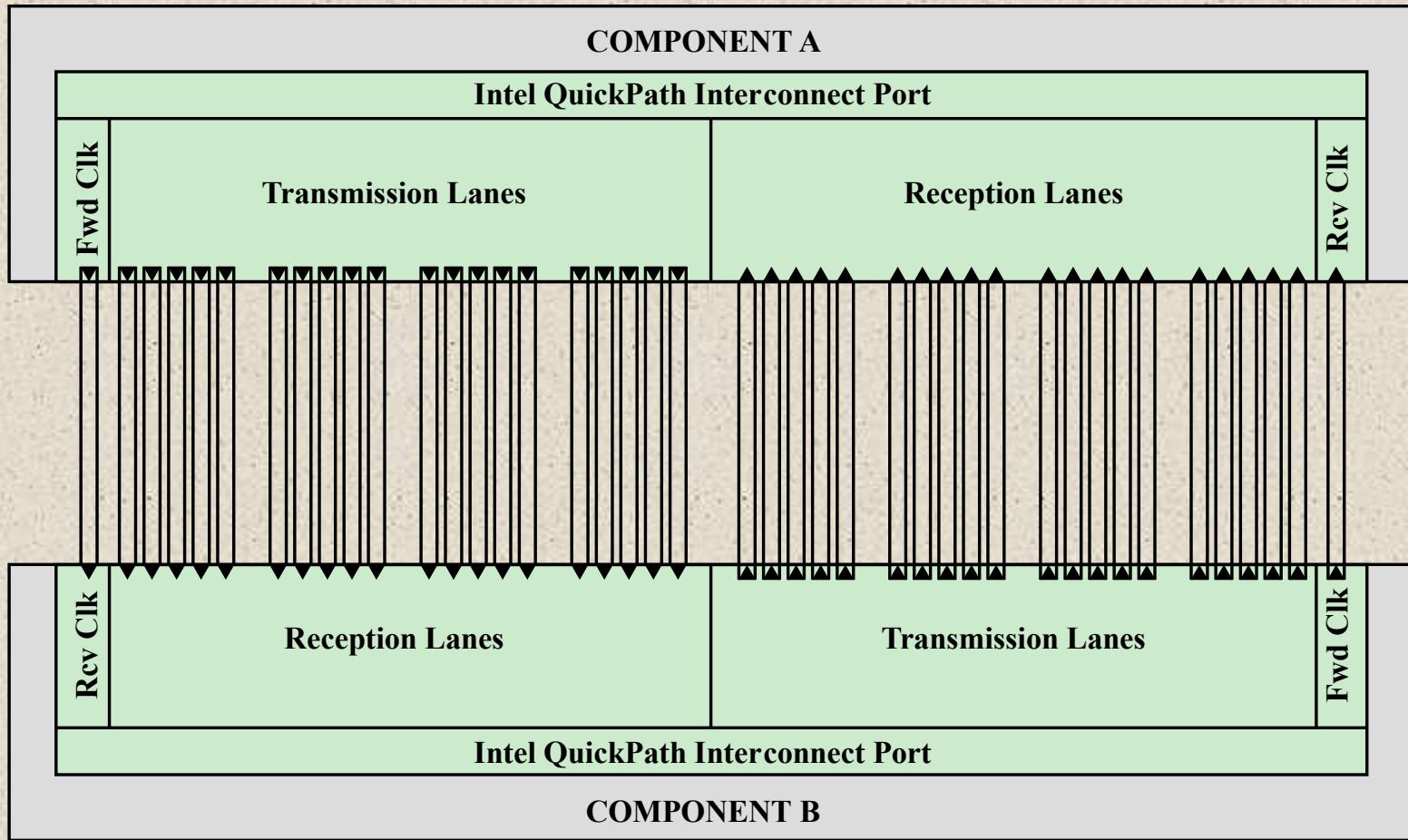


Figure 3.19 Physical Interface of the Intel QPI Interconnect

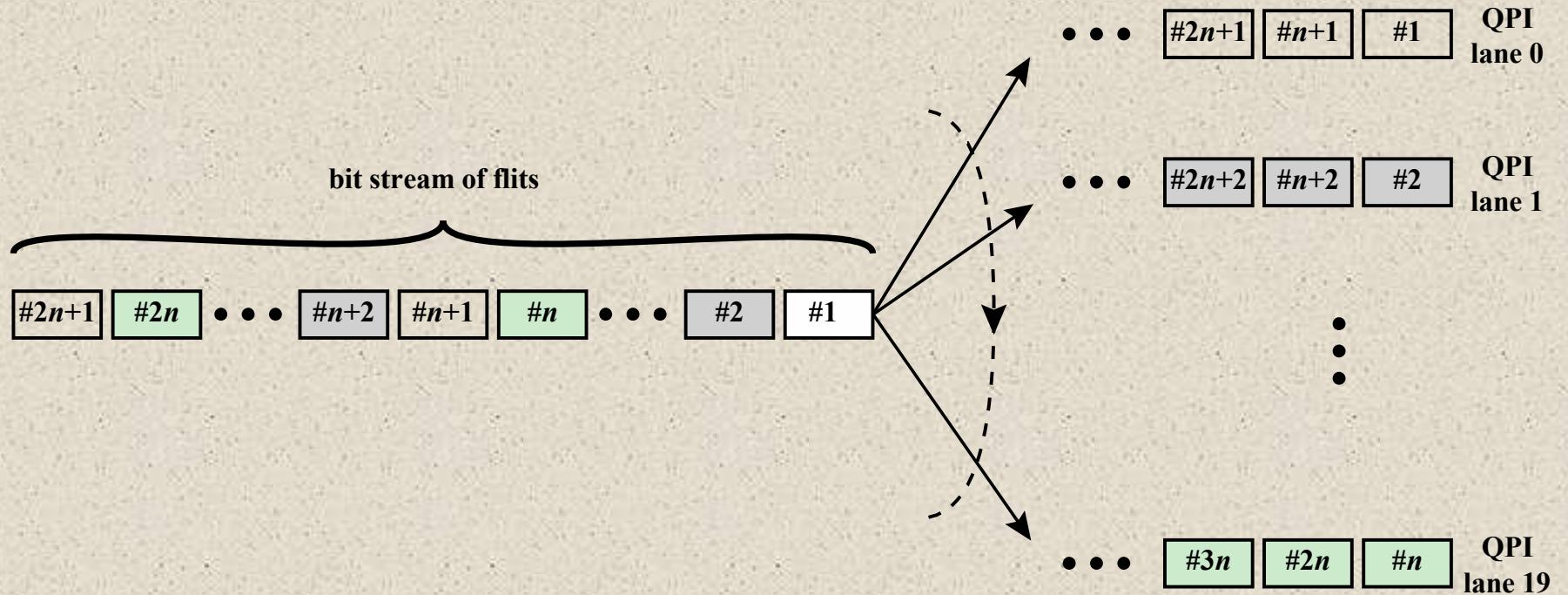


Figure 3.20 QPI Multilane Distribution

QPI Link Layer

- Performs two key functions: *flow control* and *error control*
 - Operate on the level of the flit (flow control unit)
 - Each flit consists of a 72-bit message payload and an 8-bit error control code called a *cyclic redundancy check* (CRC)
- Flow control function
 - Needed to ensure that a sending QPI entity does not overwhelm a receiving QPI entity by sending data faster than the receiver can process the data and clear buffers for more incoming data
- Error control function
 - Detects and recovers from bit errors, and so isolates higher layers from experiencing bit errors

QPI Routing and Protocol Layers

Routing Layer

- Used to determine the course that a packet will traverse across the available system interconnects
- Defined by firmware and describe the possible paths that a packet can follow

Protocol Layer

- Packet is defined as the unit of transfer
- One key function performed at this level is a cache coherency protocol which deals with making sure that main memory values held in multiple caches are consistent
- A typical data packet payload is a block of data being sent to or from a cache

Peripheral Component Interconnect (PCI)

- A popular high bandwidth, processor independent bus that can function as a mezzanine or peripheral bus
 - Bandwidth tinggi yang populer, prosesor bus independen yang dapat berfungsi sebagai bus mezzanine atau periferal
- Delivers better system performance for high speed I/O subsystems
- PCI Special Interest Group (SIG)
 - Created to develop further and maintain the compatibility of the PCI specifications
- PCI Express (PCIe)
 - Point-to-point interconnect scheme intended to replace bus-based schemes such as PCI
 - Key requirement is high capacity to support the needs of higher data rate I/O devices, such as Gigabit Ethernet
 - Another requirement deals with the need to support time dependent data streams

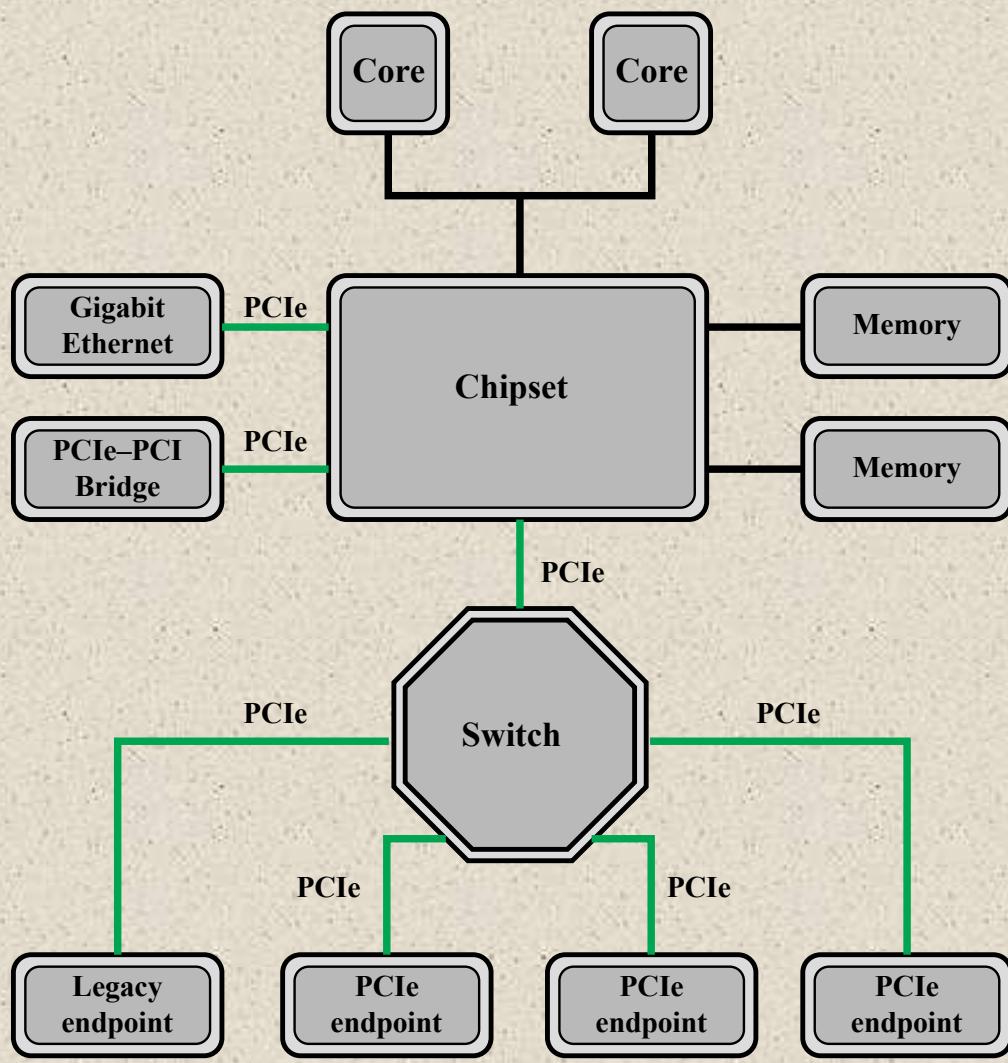


Figure 3.21 Typical Configuration Using PCIe

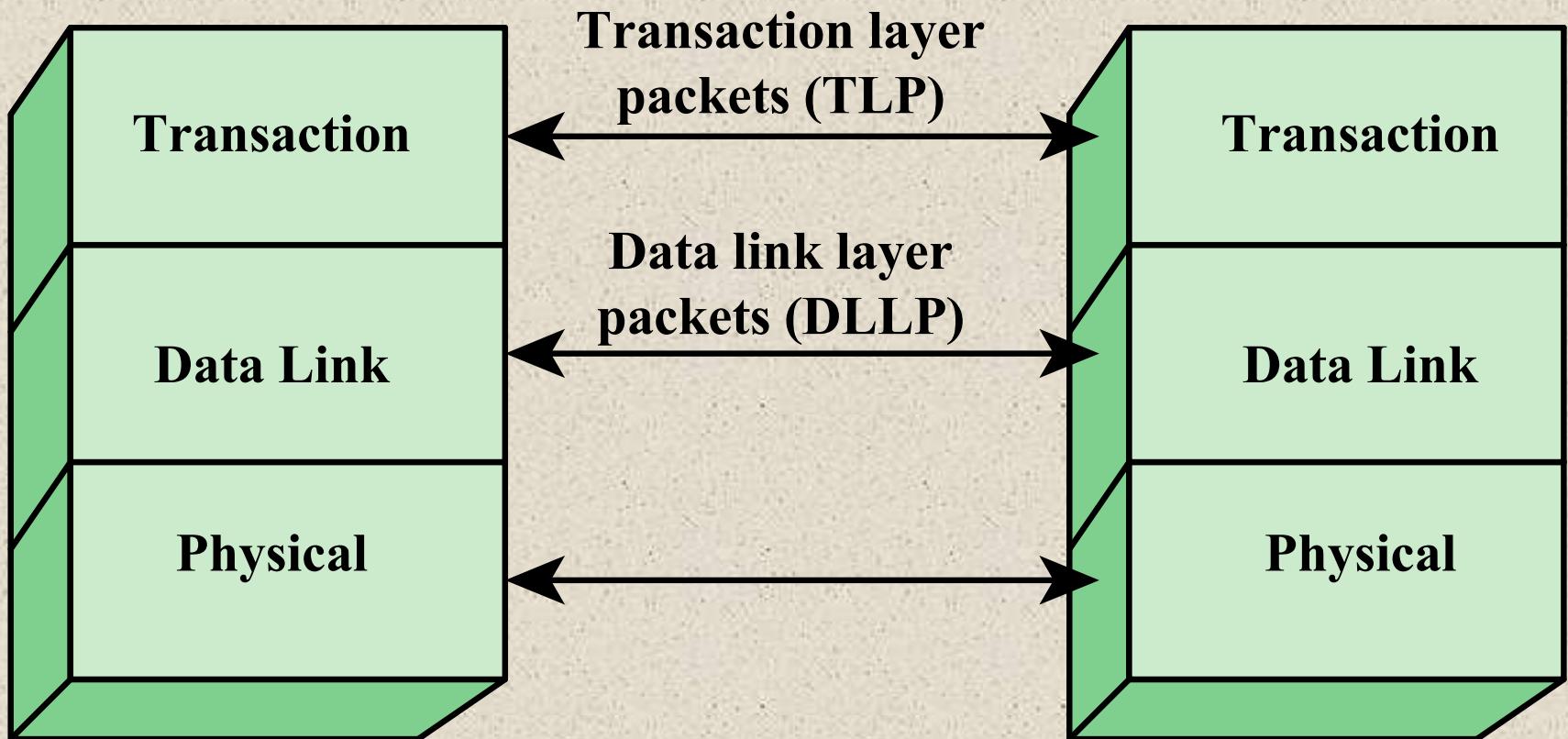


Figure 3.22 PCIe Protocol Layers

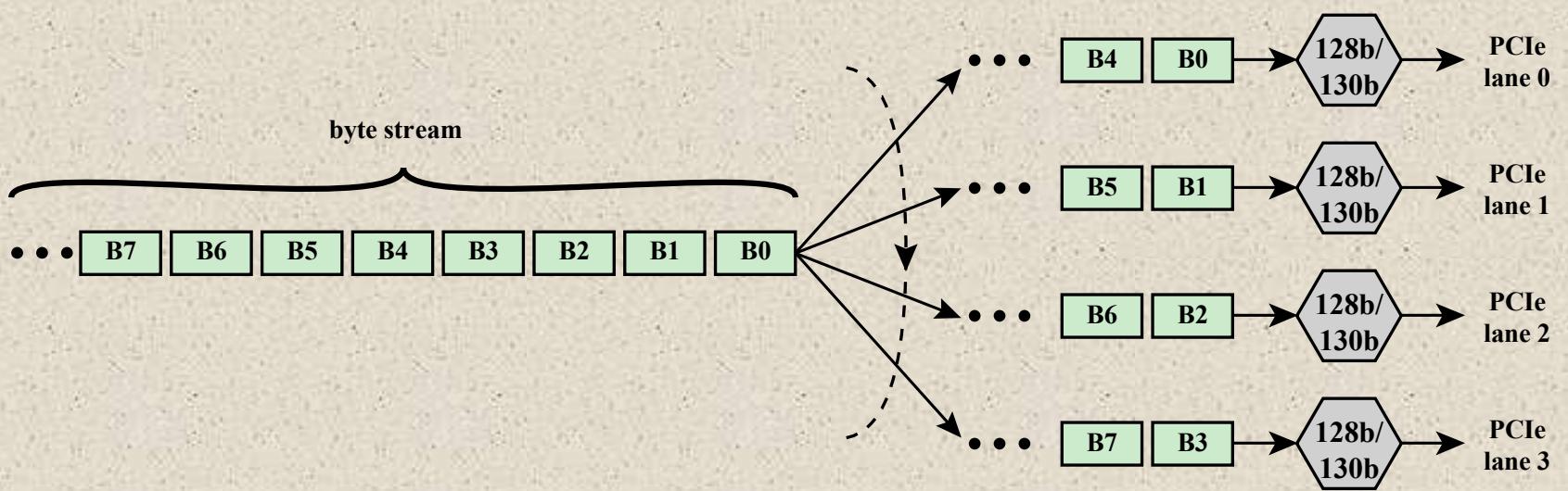


Figure 3.23 PCIe Multilane Distribution

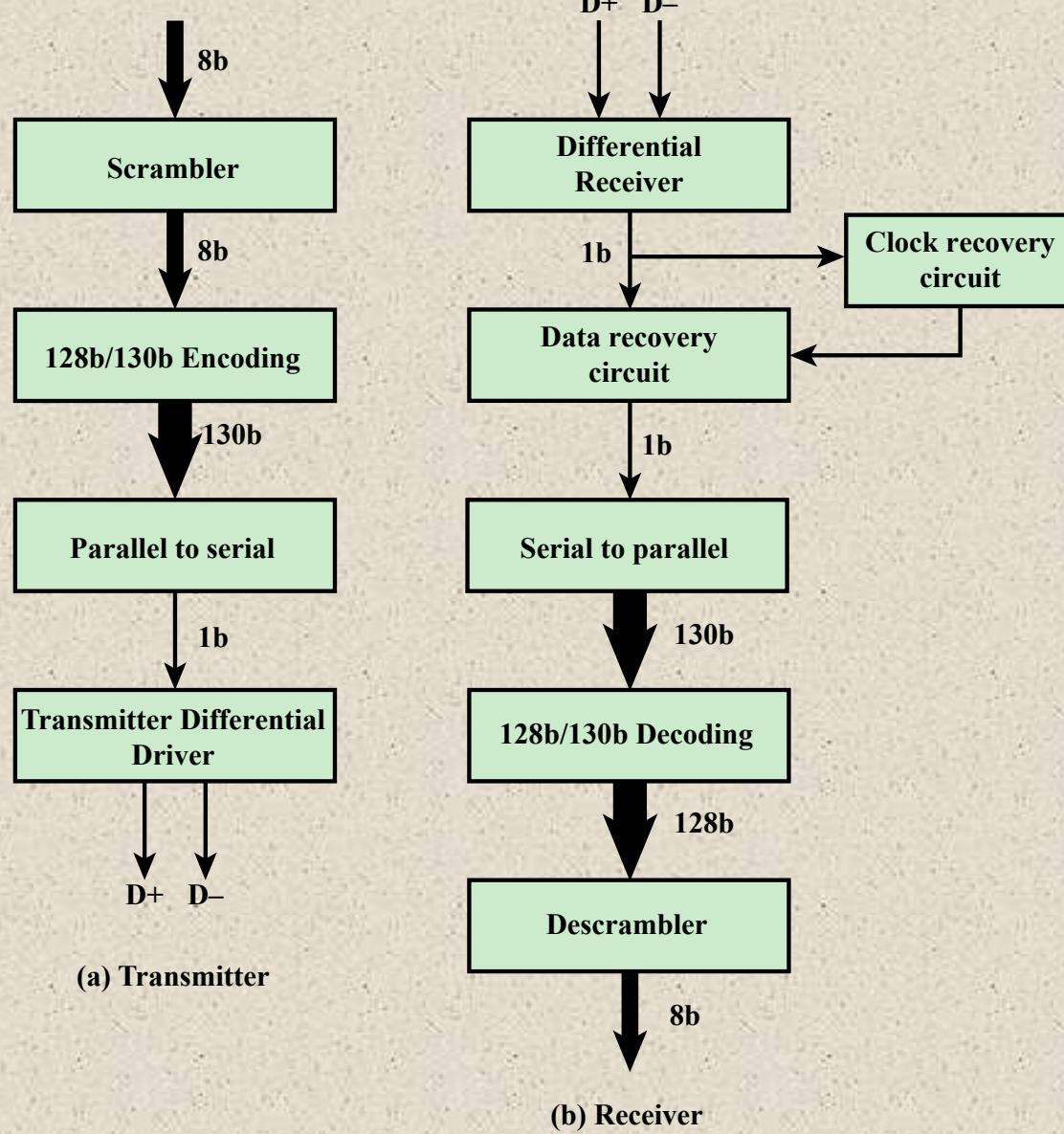


Figure 3.24 PCIe Transmit and Receive Block Diagrams



PCIe Transaction Layer (TL)



- Receives read and write requests from the software above the TL and creates request packets for transmission to a destination via the link layer
- Most transactions use a *split transaction* technique
 - A request packet is sent out by a source PCIe device which then waits for a response called a *completion packet*
- TL messages and some write transactions are posted transactions (meaning that no response is expected)
- TL packet format supports 32-bit memory addressing and extended 64-bit memory addressing



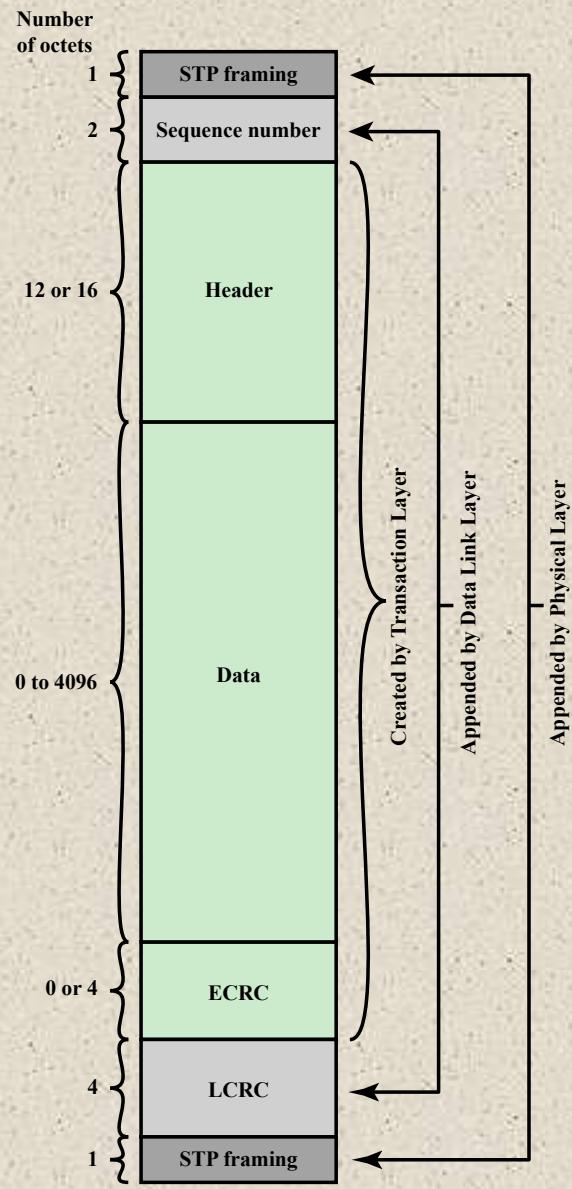
The TL supports four address spaces:

- Memory
 - The memory space includes system main memory and PCIe I/O devices
 - Certain ranges of memory addresses map into I/O devices
- Configuration
 - This address space enables the TL to read/write configuration registers associated with I/O devices
- I/O
 - This address space is used for legacy PCI devices, with reserved address ranges used to address legacy I/O devices
- Message
 - This address space is for control signals related to interrupts, error handling, and power management

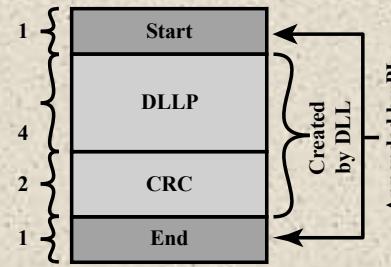
Table 3.2

PCIe TLP Transaction Types

Address Space	TLP Type	Purpose
Memory	Memory Read Request	Transfer data to or from a location in the system memory map.
	Memory Read Lock Request	
	Memory Write Request	
I/O	I/O Read Request	Transfer data to or from a location in the system memory map for legacy devices.
	I/O Write Request	
Configuration	Config Type 0 Read Request	Transfer data to or from a location in the configuration space of a PCIe device.
	Config Type 0 Write Request	
	Config Type 1 Read Request	
	Config Type 1 Write Request	
Message	Message Request	Provides in-band messaging and event reporting.
	Message Request with Data	
Memory, I/O, Configuration	Completion	Returned for certain requests.
	Completion with Data	
	Completion Locked	
	Completion Locked with Data	



(a) Transaction Layer Packet



(b) Data Link Layer Packet

Figure 3.25 PCIe Protocol Data Unit Format

+ Summary

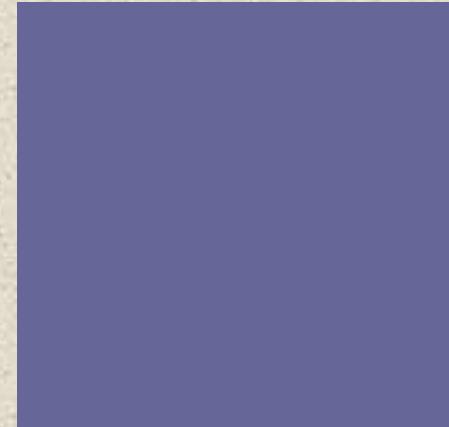
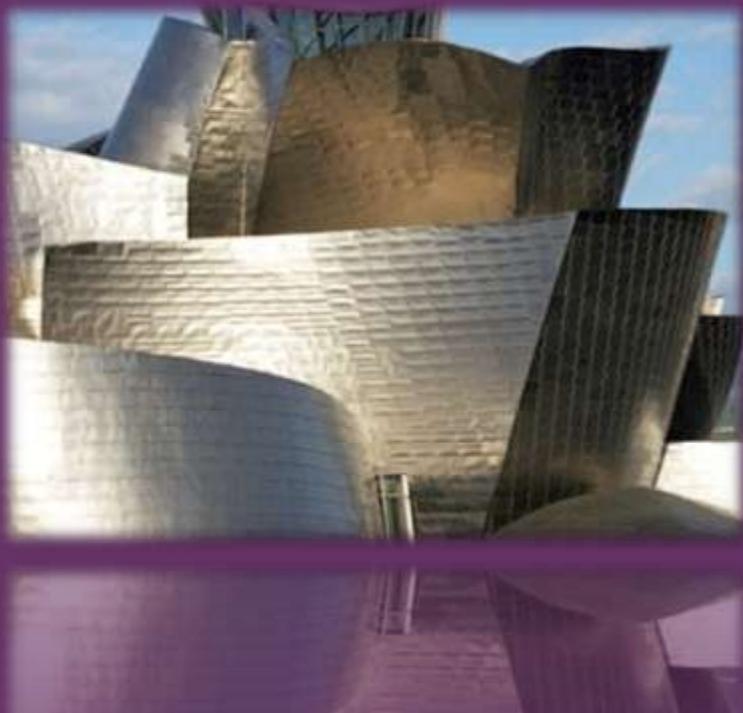
Chapter 3

- Computer components
- Computer function
 - Instruction fetch and execute
 - Interrupts
 - I/O function
- Interconnection structures
- Bus interconnection

A Top-Level View of Computer Function and Interconnection

- Point-to-point interconnect
 - QPI physical layer
 - QPI link layer
 - QPI routing layer
 - QPI protocol layer
- PCI express
 - PCI physical and logical architecture
 - PCIe physical layer
 - PCIe transaction layer
 - PCIe data link layer

+



William Stallings
Computer Organization
and Architecture
10th Edition

+ Chapter 4

Cache Memory



Location Internal (e.g. processor registers, cache, main memory) External (e.g. optical disks, magnetic disks, tapes)	Performance Access time Cycle time Transfer rate
Capacity Number of words Number of bytes	Physical Type Semiconductor Magnetic Optical Magneto-optical
Unit of Transfer Word Block	Physical Characteristics Volatile/nonvolatile Erasable/nonerasable
Access Method Sequential Direct Random Associative	Organization Memory modules

Table 4.1
Key Characteristics of Computer Memory Systems

Characteristics of Memory Systems

Lokasi

Mengacu pada apakah memori bersifat internal dan eksternal ke komputer

Memori internal sering disamakan dengan memori utama

Prosesor membutuhkan memori lokalnya sendiri, dalam bentuk register

Cache adalah bentuk lain dari memori internal

Memori eksternal terdiri dari perangkat penyimpanan periferal yang dapat diakses oleh prosesor melalui pengontrol I/O

■ Location

- Refers to whether memory is internal and external to the computer
- Internal memory is often equated with main memory
- Processor requires its own local memory, in the form of registers
- Cache is another form of internal memory
- External memory consists of peripheral storage devices that are accessible to the processor via I/O controllers

■ Capacity

- Memory is typically expressed in terms of bytes

Kapasitas

Memori biasanya dinyatakan dalam hal byte

Unit transfer

Untuk memori internal, unit transfer sama dengan jumlah
saluran listrik masuk dan keluar dari modul memori

■ Unit of transfer

- For internal memory the unit of transfer is equal to the number of electrical lines into and out of the memory module

Method of Accessing Units of Data

Sequential access

Memory is organized into units of data called records

Akses harus dilakukan dalam urutan linear spesifik
Access must be made in a specific linear sequence

Access time is variable

Direct access

Involves a shared read-write mechanism

Blok individu atau Catatan memiliki keunikan alamat berdasarkan lokasi fisik

Individual blocks or records have a unique address based on physical location

Access time is variable

Random access

Each addressable location in memory has a unique, physically wired-in addressing mechanism

Waktu untuk mengakses a Lokasi yang diberikan adalah independen dari urutan sebelumnya mengakses dan konstan

The time to access a given location is independent of the sequence of prior accesses and is constant

Any location can be selected at random and directly addressed and accessed

Main memory and some cache systems are random access

Associative

A word is retrieved based on a portion of its contents rather than its address

Each location has its own addressing mechanism and retrieval time is constant independent of location or prior access patterns

Cache memories may employ associative access

Setiap lokasi memiliki sendiri mekanisme pengalaman dan waktu pengambilan adalah konstanta independen dari lokasi atau akses sebelumnya Pola

Capacity and Performance:

The two most important characteristics of memory

Three performance parameters are used:

Access time (latency)

- For random-access memory it is the time it takes to perform a read or write operation
- For non-random-access memory it is the time it takes to position the read-write mechanism at the desired location

Memory cycle time

- Access time plus any additional time required before second access can commence
- Additional time may be required for transients to die out on signal lines or to regenerate data if they are read destructively
- Concerned with the system bus, not the processor

Transfer rate

- The rate at which data can be transferred into or out of a memory unit
- For random-access memory it is equal to $1/(\text{cycle time})$



Memory

- The most common forms are:

- Semiconductor memory
- Magnetic surface memory
- Optical
- Magneto-optical

Beberapa karakteristik fisik penyimpanan data adalah penting:

Memori volatil

Informasi meluruh secara alami atau hilang ketika daya listrik dimatikan

Memori nonvolatile

Setelah dicatat, informasi tetap tanpa kerusakan sampai sengaja diubah

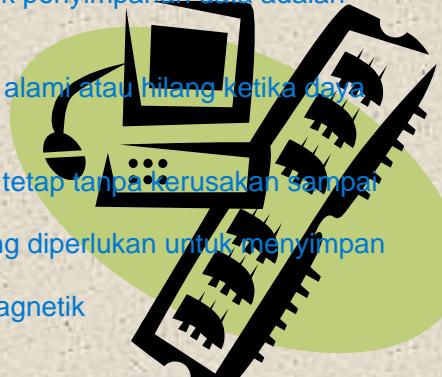
Tidak ada daya listrik yang diperlukan untuk menyimpan informasi

Kenangan permukaan magnetik

Apakah nonvolatile

Memori semikonduktor

Mungkin mudah menguap atau nonvolatile



- Several physical characteristics of data storage are important:

- Volatile memory
 - Information decays naturally or is lost when electrical power is switched off
- Nonvolatile memory
 - Once recorded, information remains without deterioration until deliberately changed
 - No electrical power is needed to retain information
- Magnetic-surface memories
 - Are nonvolatile
- Semiconductor memory
 - May be either volatile or nonvolatile
- Nonerasable memory
 - Cannot be altered, except by destroying the storage unit
 - Semiconductor memory of this type is known as read-only memory (ROM)

Memori yang tidak dapat disembuhkan

Tidak dapat diubah, kecuali dengan menghancurkan unit penyimpanan

Memori semikonduktor jenis ini dikenal sebagai memori read-only (ROM)

Untuk memori akses acak, organisasi adalah masalah desain utama

Organisasi mengacu pada susunan fisik bit untuk membentuk kata-kata

- For random-access memory the organization is a key design issue
 - Organization refers to the physical arrangement of bits to form words

Memory Hierarchy

- Design constraints on a computer's memory can be summed up by three questions:
 - How much, how fast, how expensive
 - There is a trade-off among capacity, access time, and cost
 - Faster access time, greater cost per bit
 - Greater capacity, smaller cost per bit
 - Greater capacity, slower access time
 - The way out of the memory dilemma is not to rely on a single memory component or technology, but to employ a memory hierarchy
- Kendala desain pada memori komputer dapat dijumlaskan dengan tiga pertanyaan:
Berapa banyak, seberapa cepat, seberapa mahal
Ada trade-off antara kapasitas, waktu akses, dan biaya
Waktu akses yang lebih cepat, biaya per bit yang lebih besar
Kapasitas yang lebih besar, biaya per bit yang lebih kecil
Kapasitas yang lebih besar, waktu akses lebih lambat
Jalan keluar dari dilema memori adalah tidak bergantung pada satu komponen memori atau teknologi, tetapi untuk menggunakan memori hierarki

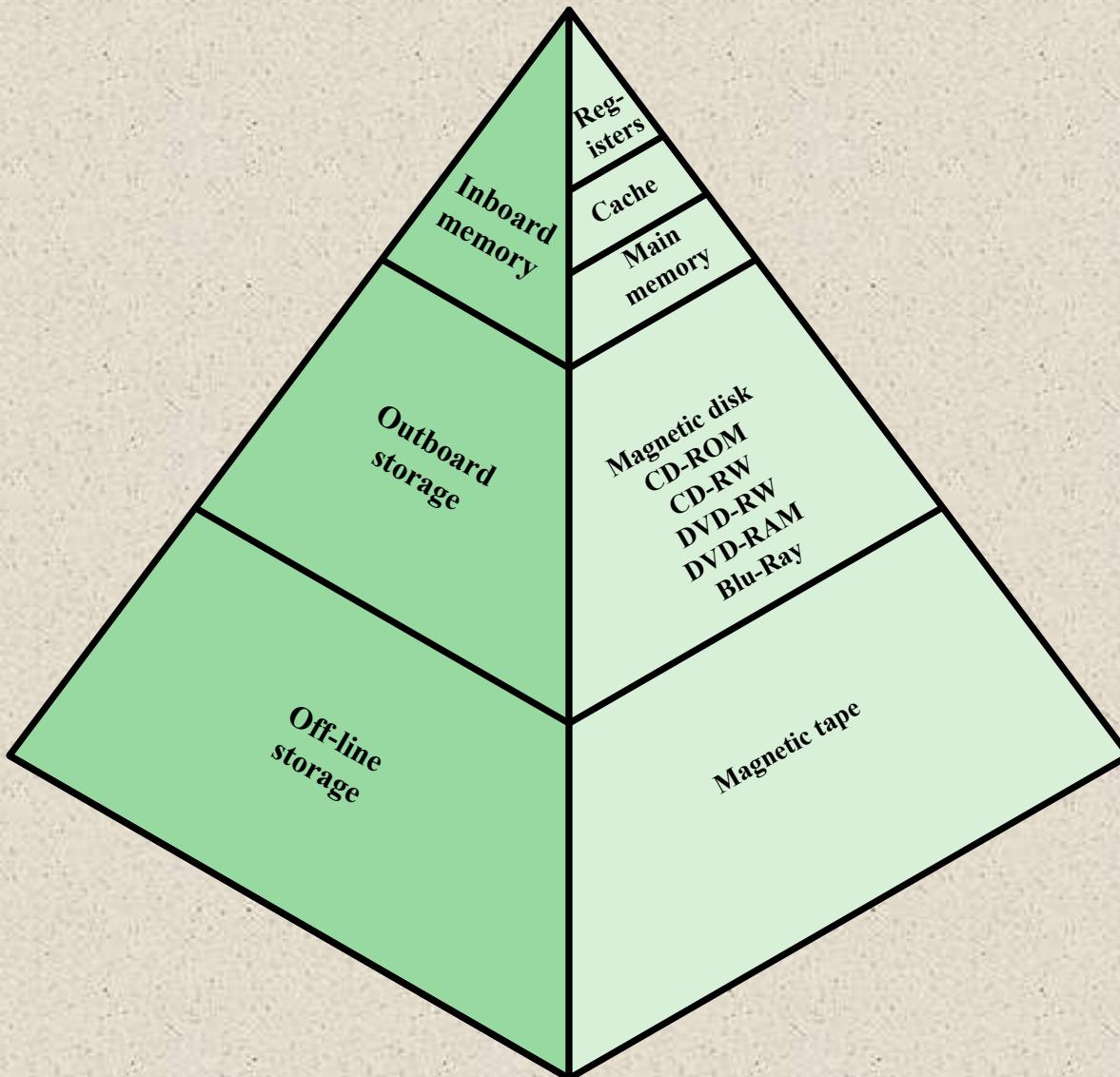


Figure 4.1 The Memory Hierarchy

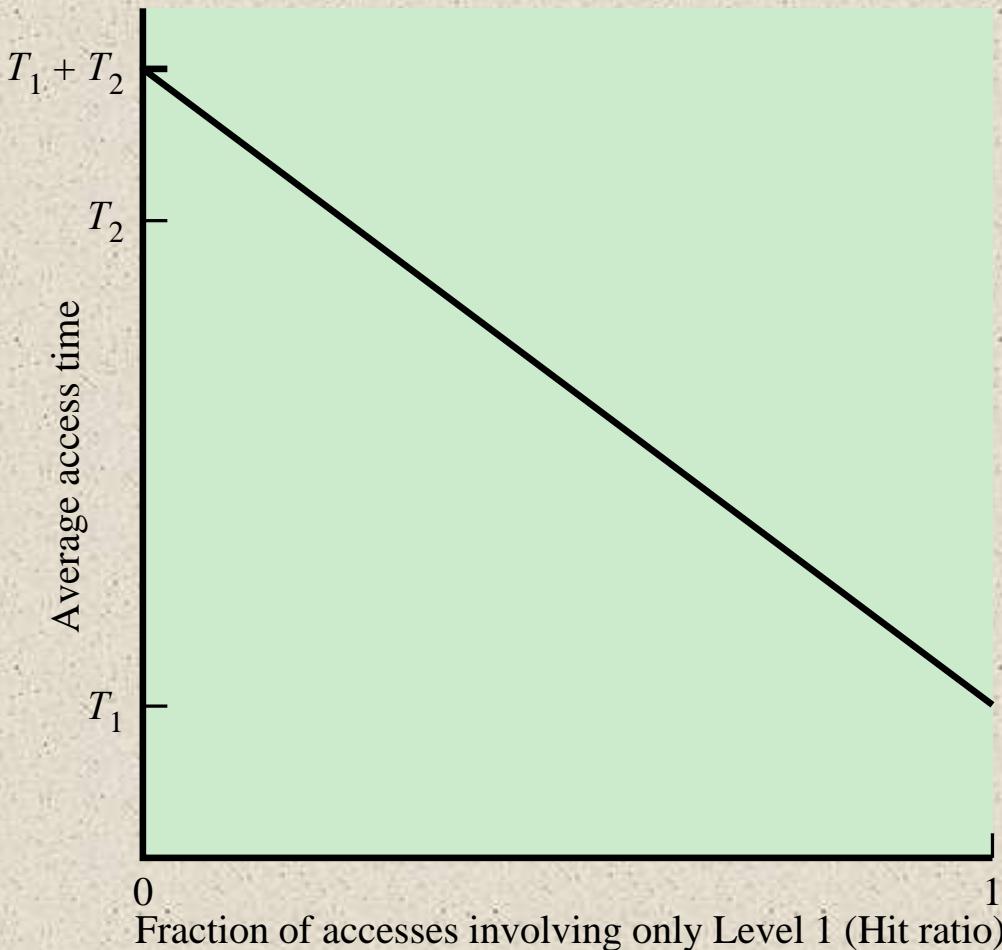


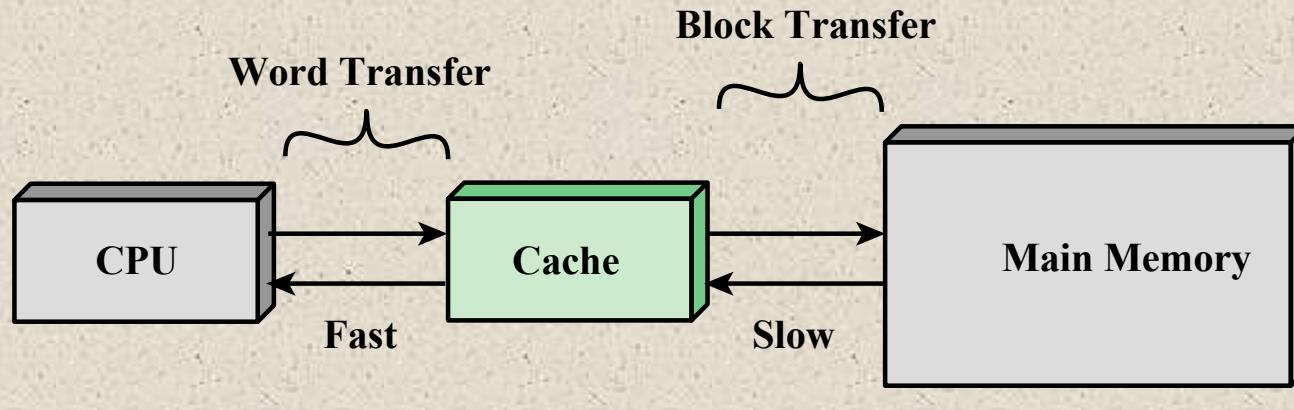
Figure 4.2 Performance of a Simple Two-Level Memory



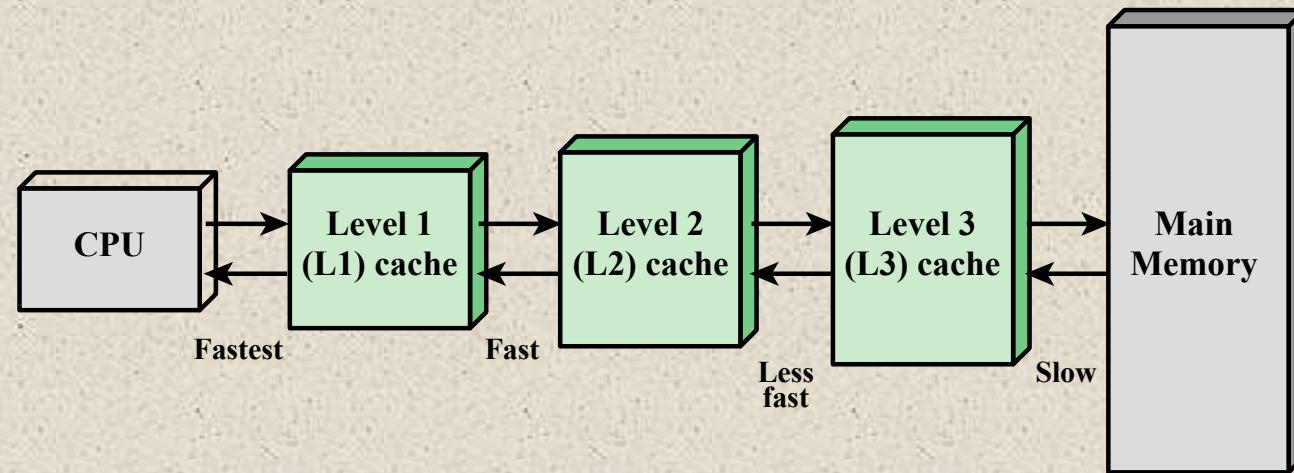
Memory

- The use of three levels exploits the fact that semiconductor memory comes in a variety of types which differ in speed and cost
- Data are stored more permanently on external mass storage devices
- External, nonvolatile memory is also referred to as **secondary memory** or **auxiliary memory**
 - Penggunaan tiga tingkat mengeksplorasi fakta bahwa semikonduktor memori datang dalam berbagai jenis yang berbeda dalam kecepatan dan biaya
 - Data disimpan lebih permanen pada penyimpanan massal eksternal Perangkat
 - Memori eksternal dan nonvolatile juga disebut sebagai sekunder, memori atau memori tambahan
- Disk cache
 - A portion of main memory can be used as a buffer to hold data temporarily that is to be read out to disk
 - A few large transfers of data can be used instead of many small transfers of data
 - Data can be retrieved rapidly from the software cache rather than slowly from the disk

Cache disk
Sebagian dari memori utama dapat digunakan sebagai buffer untuk menyimpan data sementara itu harus dibacakan ke disk
Beberapa transfer data besar dapat digunakan alih-alih banyak transfer data
Data dapat diambil dengan cepat dari cache perangkat lunak daripada perlahan-lahan dari disk



(a) Single cache



(b) Three-level cache organization

Figure 4.3 Cache and Main Memory

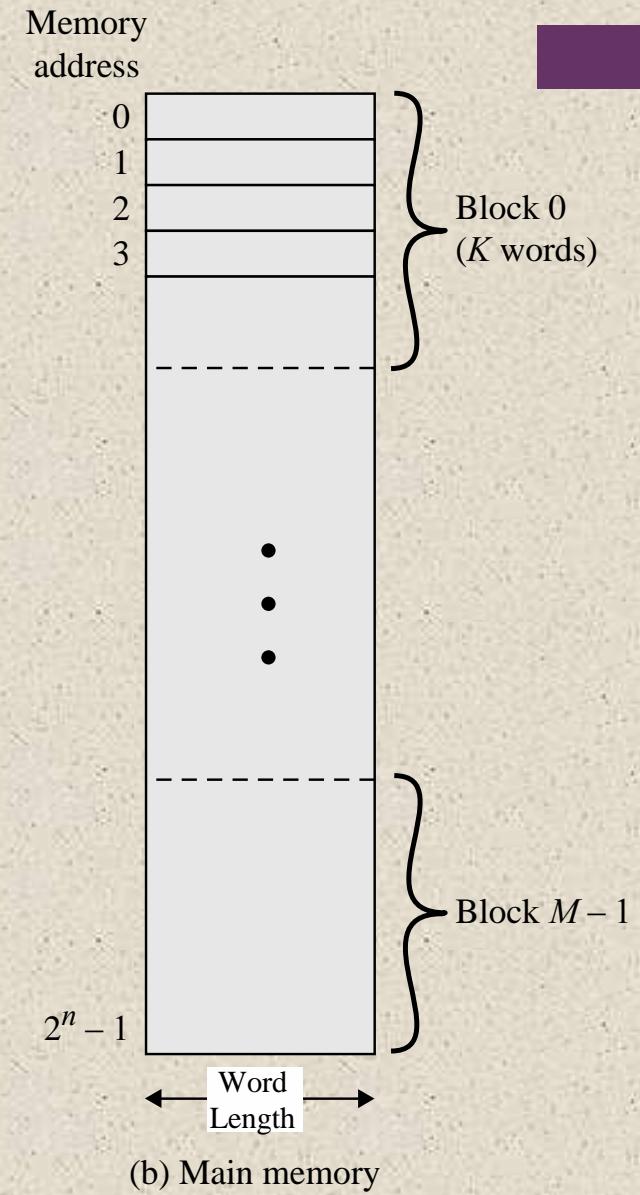
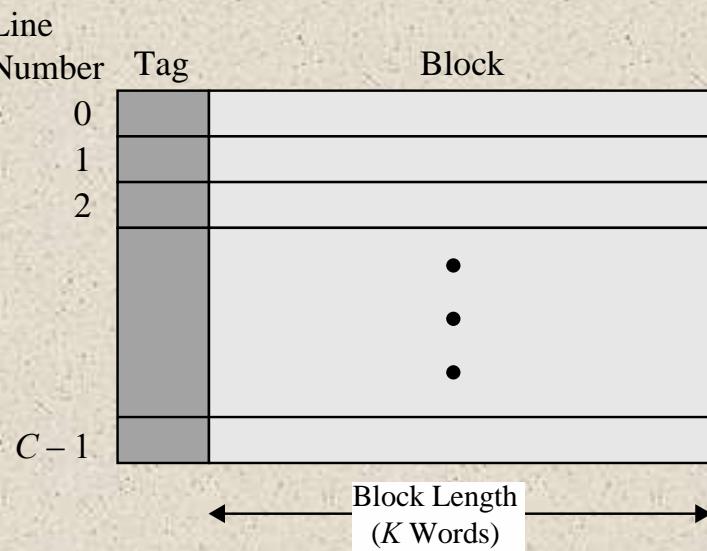


Figure 4.4 Cache/Main-Memory Structure

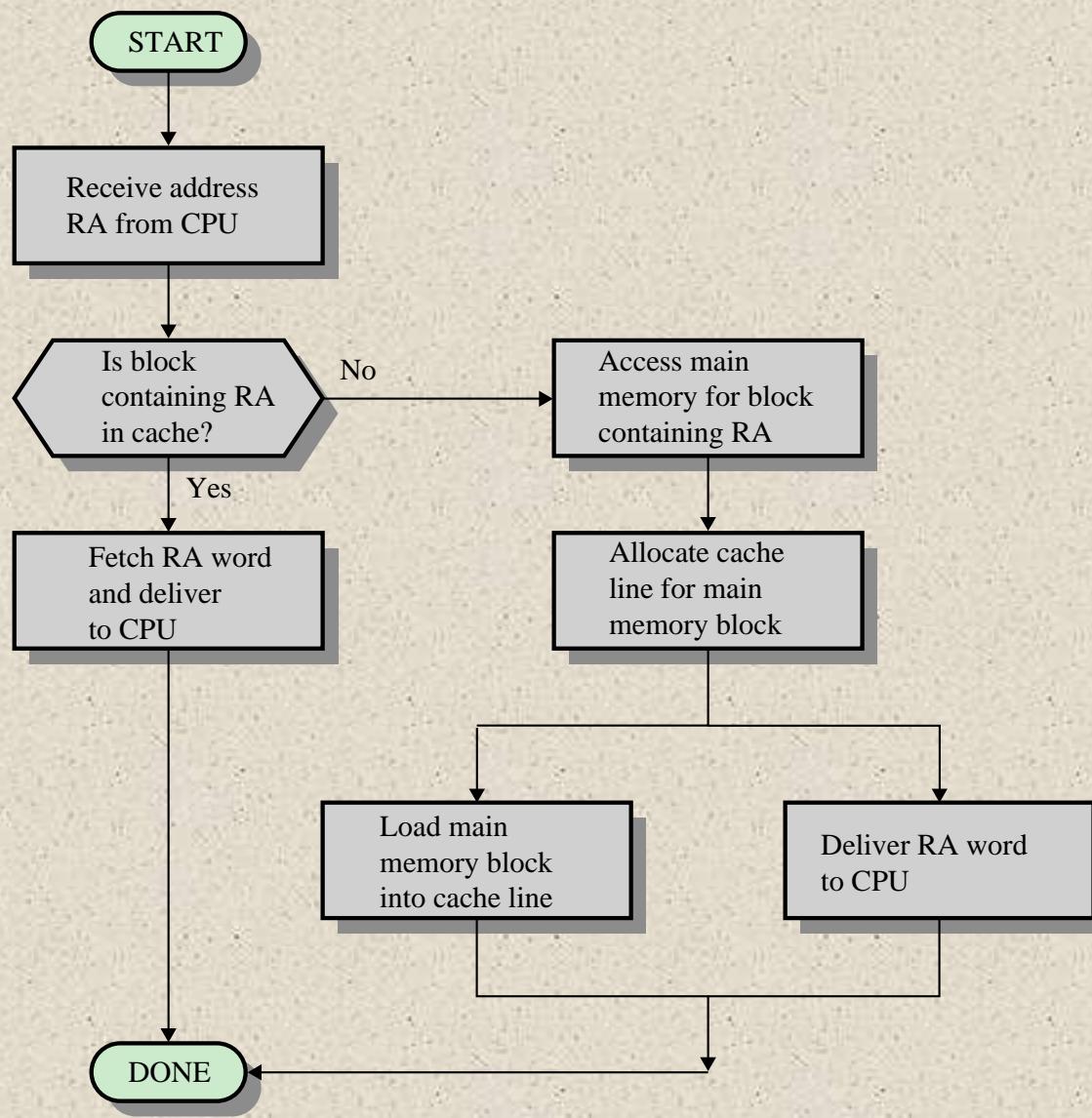


Figure 4.5 Cache Read Operation

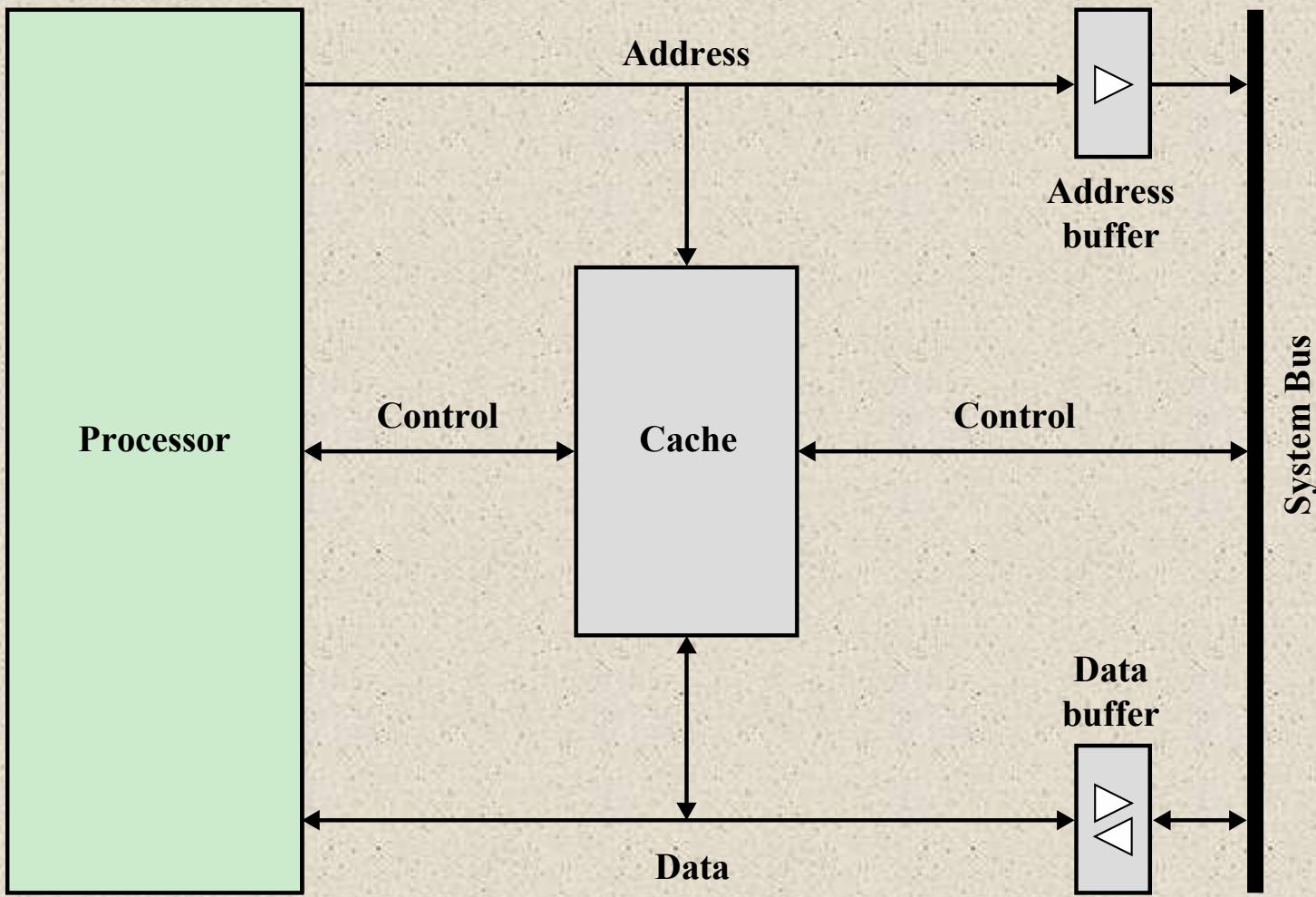


Figure 4.6 Typical Cache Organization

Cache Addresses

Logical
Physical

Cache Size

Mapping Function

Direct
Associative
Set Associative

Replacement Algorithm

Least recently used (LRU)
First in first out (FIFO)
Least frequently used (LFU)
Random

Write Policy

Write through
Write back

Line Size

Number of caches

Single or two level
Unified or split

Cache Addresses

Virtual Memory



■ Virtual memory

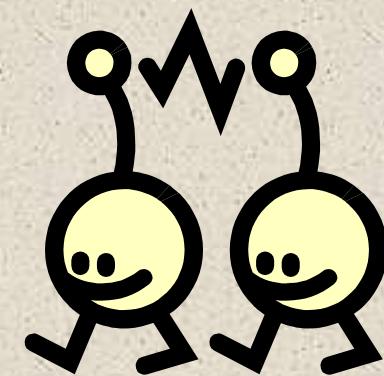
- Facility that allows programs to address memory from a logical point of view, without regard to the amount of main memory physically available
- When used, the address fields of machine instructions contain virtual addresses
- For reads to and writes from main memory, a hardware memory management unit (MMU) translates each virtual address into a physical address in main memory

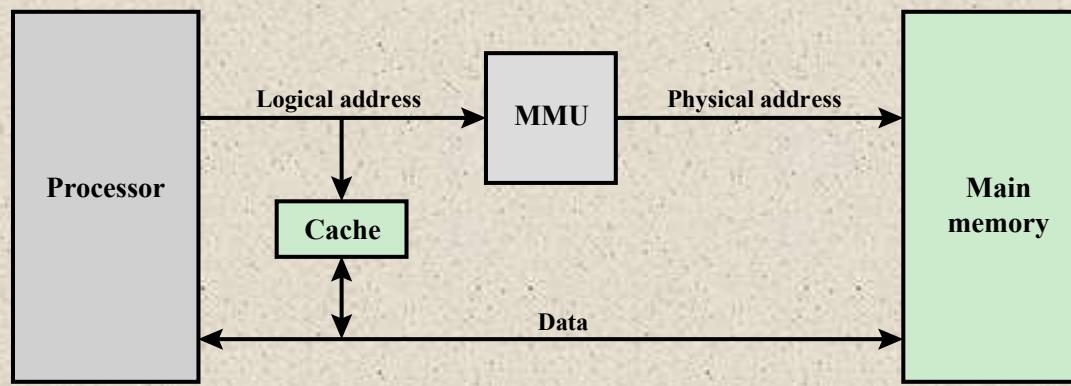
Memori virtual

Fasilitas yang memungkinkan program untuk mengatasi memori dari titik logis pandangan, tanpa memperhatikan jumlah memori utama secara fisik tersedia

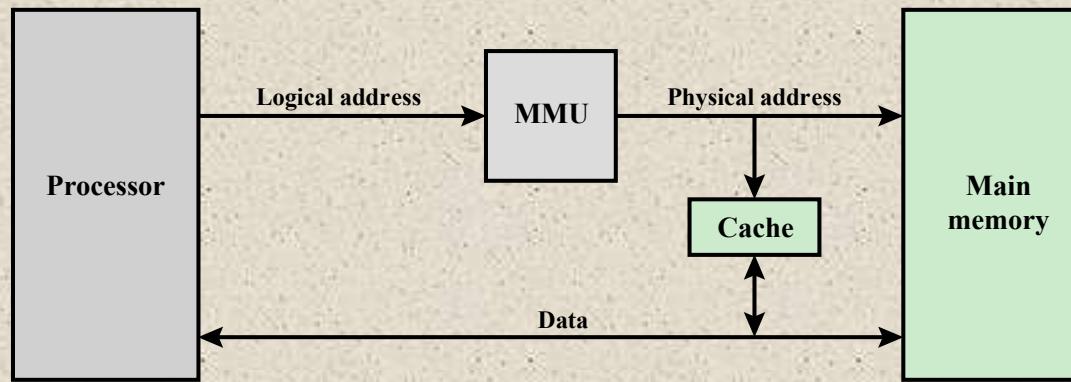
Saat digunakan, bidang alamat instruksi mesin berisi alamat virtual

Untuk membaca dan menulis dari memori utama, memori perangkat keras unit manajemen (MMU) menerjemahkan setiap alamat virtual menjadi alamat fisik dalam memori utama





(a) Logical Cache



(b) Physical Cache

Figure 4.7 Logical and Physical Caches

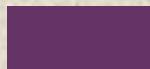


Table 4.3
**Cache Sizes
of Some
Processors**

Processor	Type	Year of Introduction	L1 Cache	L2 cache	L3 Cache
IBM 360/85	Mainframe	1968	16 to 32 kB	—	—
PDP-11/70	Minicomputer	1975	1 kB	—	—
VAX 11/780	Minicomputer	1978	16 kB	—	—
IBM 3033	Mainframe	1978	64 kB	—	—
IBM 3090	Mainframe	1985	128 to 256 kB	—	—
Intel 80486	PC	1989	8 kB	—	—
Pentium	PC	1993	8 kB/8 kB	256 to 512 KB	—
PowerPC 601	PC	1993	32 kB	—	—
PowerPC 620	PC	1996	32 kB/32 kB	—	—
PowerPC G4	PC/server	1999	32 kB/32 kB	256 KB to 1 MB	2 MB
IBM S/390 G6	Mainframe	1999	256 kB	8 MB	—
Pentium 4	PC/server	2000	8 kB/8 kB	256 KB	—
IBM SP	High-end server/ supercomputer	2000	64 kB/32 kB	8 MB	—
CRAY MTab	Supercomputer	2000	8 kB	2 MB	—
Itanium	PC/server	2001	16 kB/16 kB	96 KB	4 MB
Itanium 2	PC/server	2002	32 kB	256 KB	6 MB
IBM POWER5	High-end server	2003	64 kB	1.9 MB	36 MB
CRAY XD-1	Supercomputer	2004	64 kB/64 kB	1MB	—
IBM POWER6	PC/server	2007	64 kB/64 kB	4 MB	32 MB
IBM z10	Mainframe	2008	64 kB/128 kB	3 MB	24-48 MB
Intel Core i7 EE 990	Workstation/ server	2011	6 ´ 32 kB/32 kB	1.5 MB	12 MB
IBM zEnterprise 196	Mainframe/ Server	2011	24 ´ 64 kB/ 128 kB	24 ´ 1.5 MB	24 MB L3 192 MB L4

^a Two values separated by a slash refer to instruction and data caches.

^b Both caches are instruction only; no data caches.

(Table can be found on page 134 in the textbook.)

Mapping Function

- Because there are fewer cache lines than main memory blocks, an algorithm is needed for mapping main memory blocks into cache lines
- Three techniques can be used:

Direct

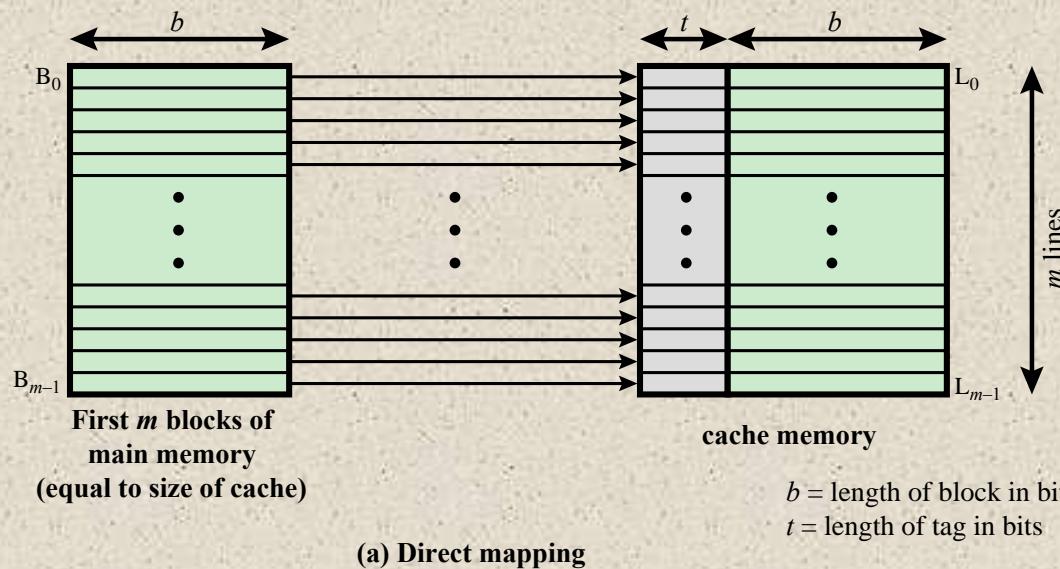
- The simplest technique
- Maps each block of main memory into only one possible cache line

Associative

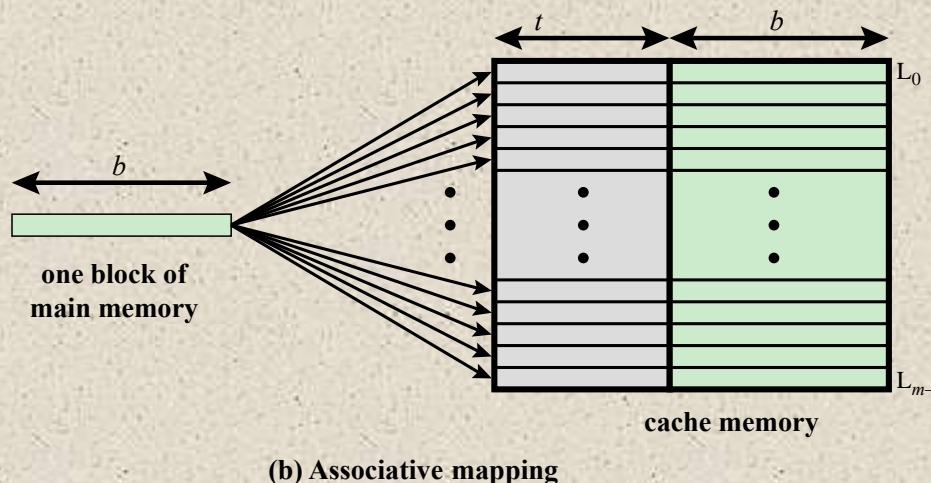
- Permits each main memory block to be loaded into any line of the cache
 - The cache control logic interprets a memory address simply as a Tag and a Word field
 - To determine whether a block is in the cache, the cache control logic must simultaneously examine every line's Tag for a match

Set Associative

- A compromise that exhibits the strengths of both the direct and associative approaches while reducing their disadvantages



(a) Direct mapping



(b) Associative mapping

**Figure 4.8 Mapping From Main Memory to Cache:
Direct and Associative**

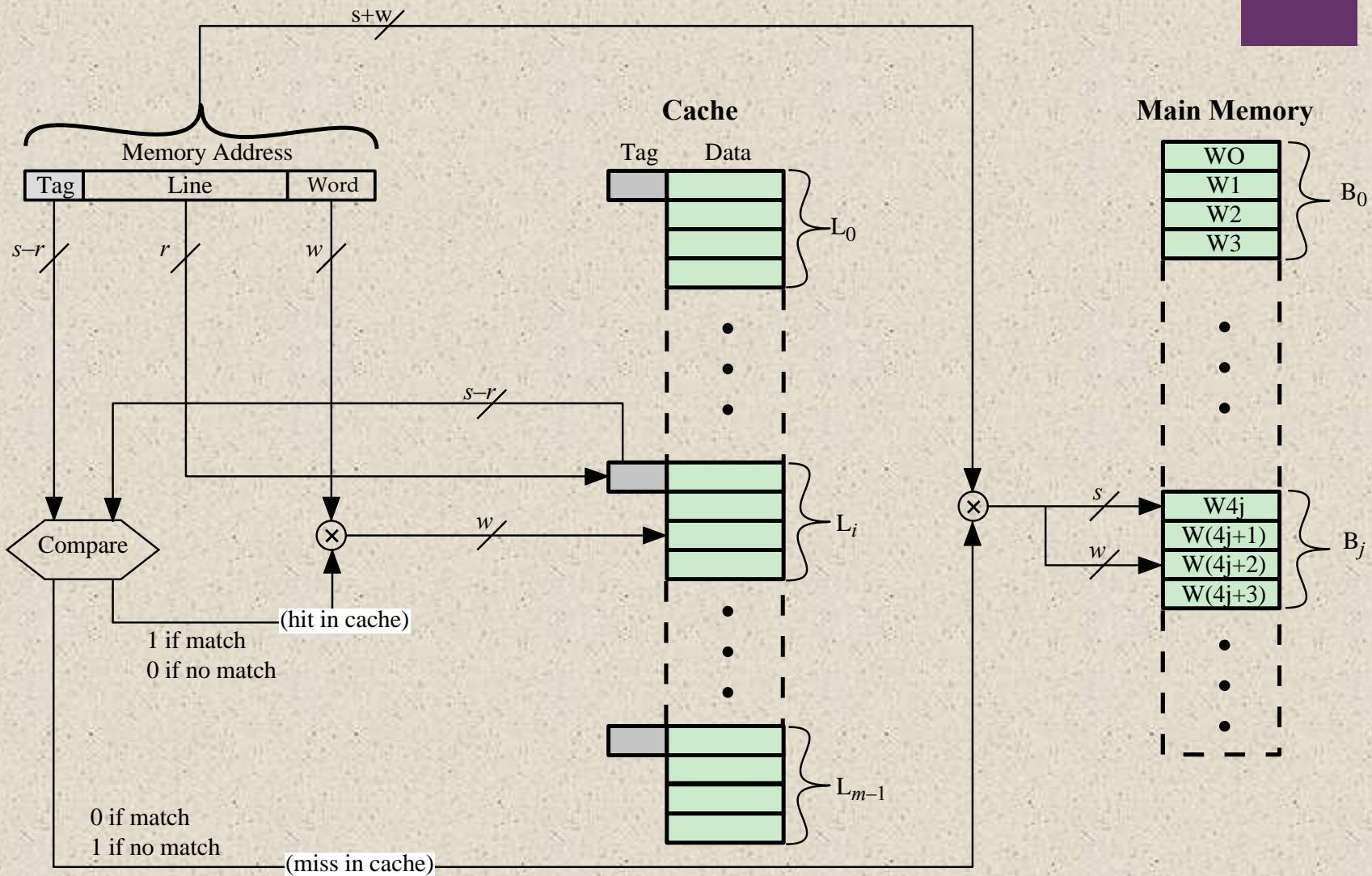


Figure 4.9 Direct-Mapping Cache Organization

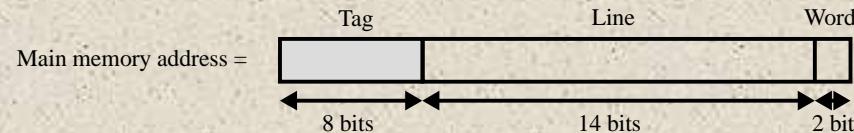
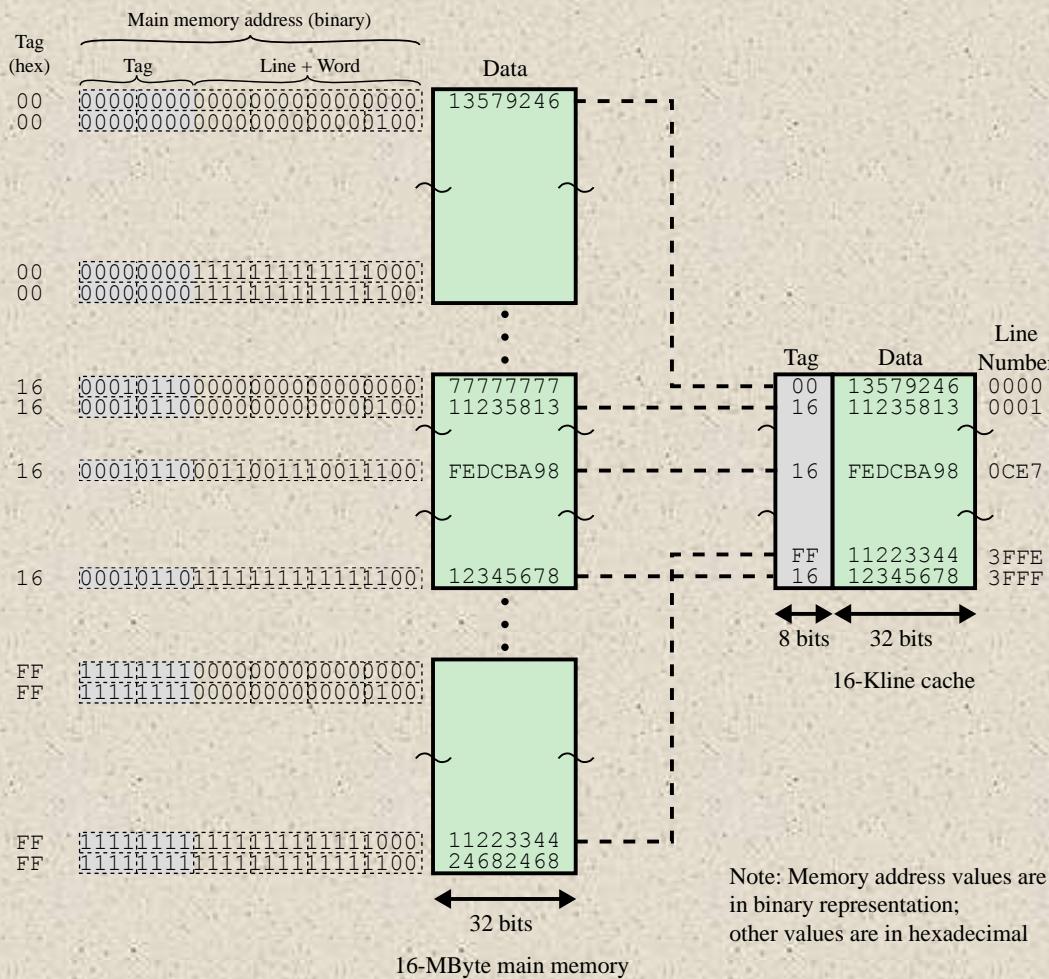
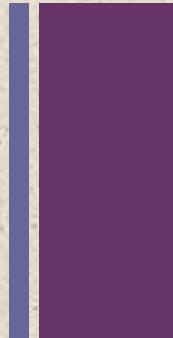


Figure 4.10 Direct Mapping Example

Direct Mapping Summary



- Address length = $(s + w)$ bits
- Number of addressable units = 2^{s+w} words or bytes
- Block size = line size = 2^w words or bytes
- Number of blocks in main memory = $2^{s+w}/2^w = 2^s$
- Number of lines in cache = $m = 2^r$
- Size of tag = $(s - r)$ bits





Victim Cache



- Originally proposed as an approach to reduce the conflict misses of direct mapped caches without affecting its fast access time
 - Fully associative cache
 - Typical size is 4 to 16 cache lines
 - Residing between direct mapped L1 cache and the next level of memory

Awalnya diusulkan sebagai pendekatan untuk mengurangi konflik meleset dari cache yang dipetakan langsung tanpa mempengaruhi puasanya

waktu akses

Cache asosiatif penuh

Ukuran tipikal adalah 4 hingga 16 baris cache

Berada di antara cache L1 yang dipetakan langsung dan tingkat ingatan

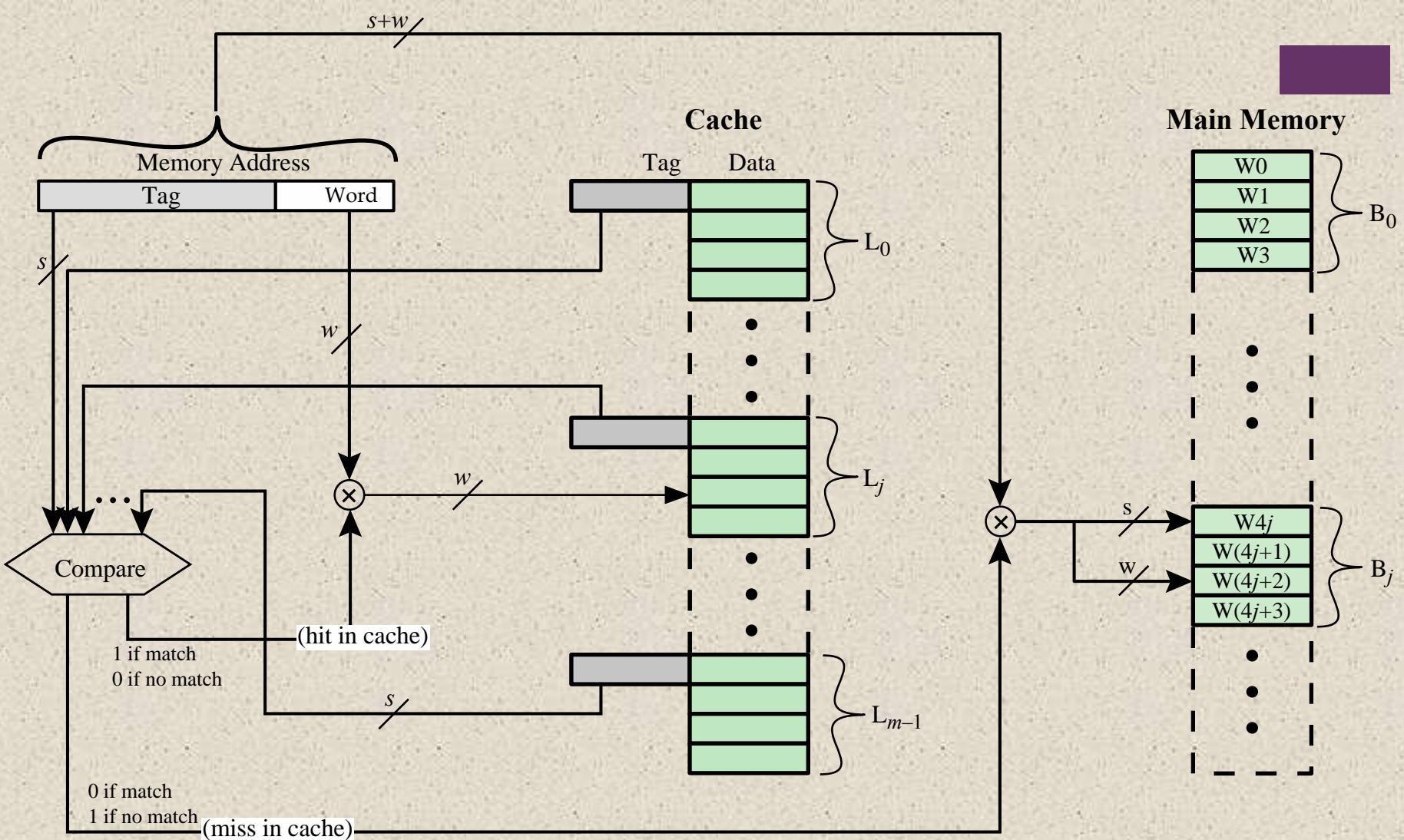


Figure 4.11 Fully Associative Cache Organization

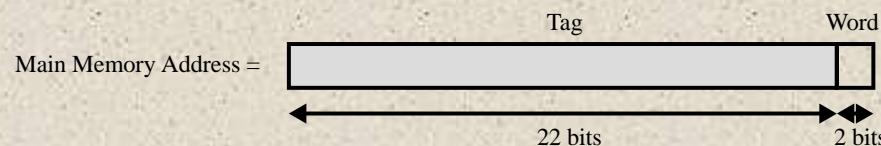
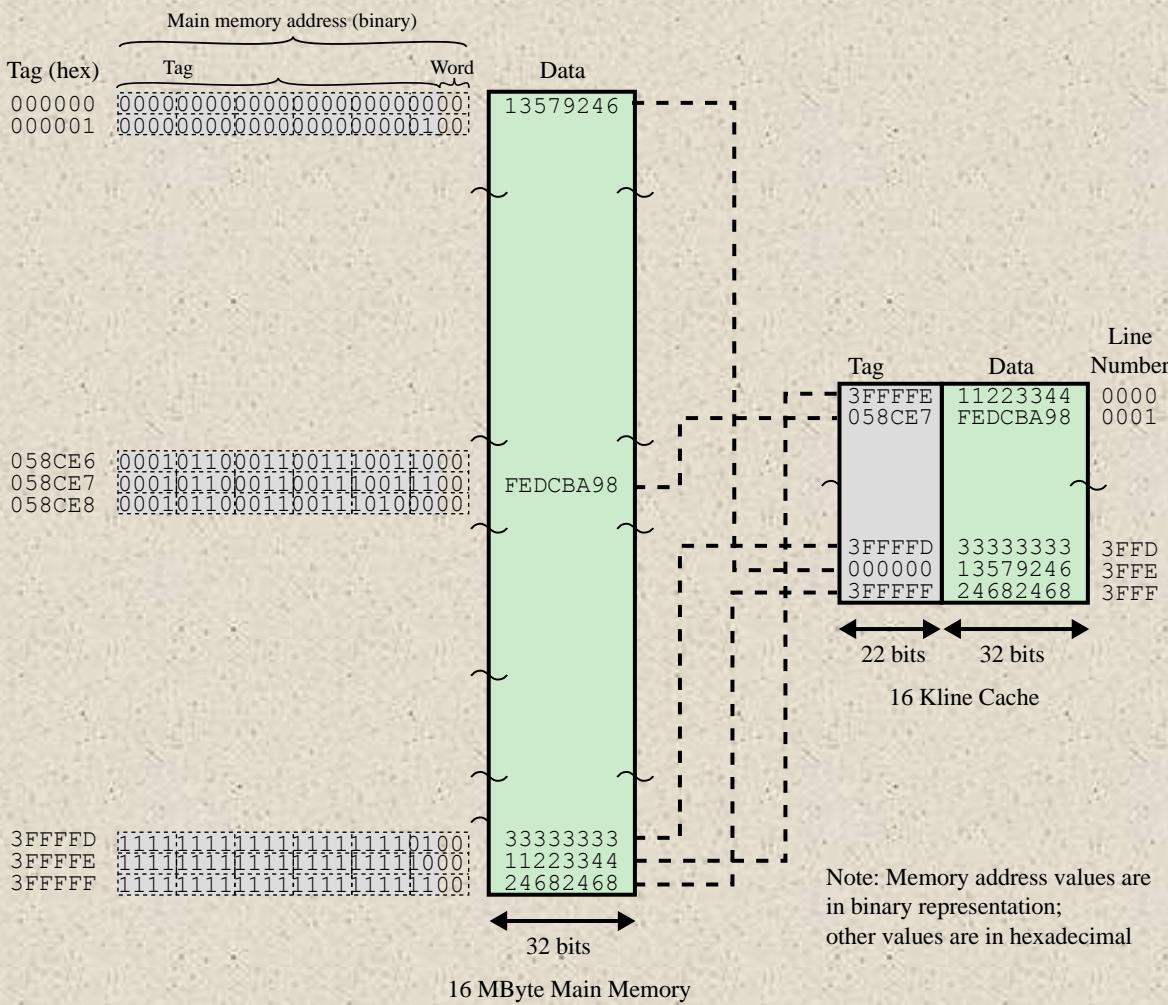


Figure 4.12 Associative Mapping Example

Associative Mapping Summary



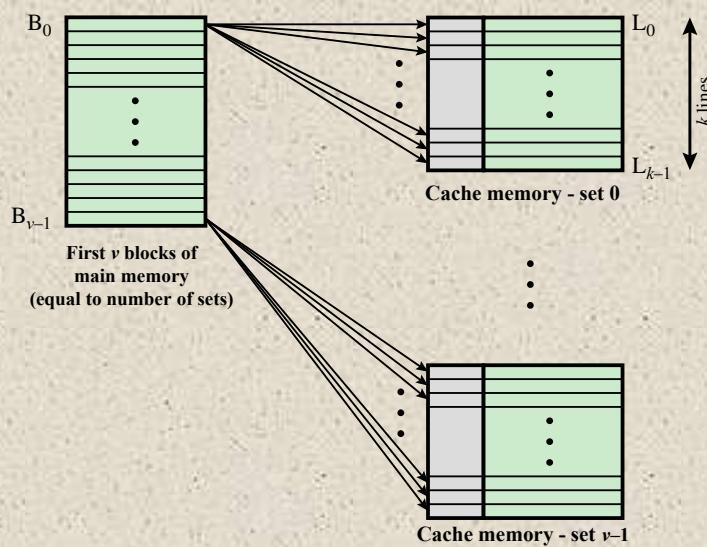
- Address length = $(s + w)$ bits
- Number of addressable units = 2^{s+w} words or bytes
- Block size = line size = 2^w words or bytes
- Number of blocks in main memory = $2^{s+w}/2^w = 2^s$
- Number of lines in cache = undetermined
- Size of tag = s bits





Set Associative Mapping

- Compromise that exhibits the strengths of both the direct and associative approaches while reducing their disadvantages
 - Cache consists of a number of sets
 - Each set contains a number of lines
 - A given block maps to any line in a given set
 - e.g. 2 lines per set
 - 2 way associative mapping
 - A given block can be in one of 2 lines in only one set
- Kompromi yang menunjukkan kekuatan baik langsung maupun pendekatan asosiatif sambil mengurangi kerugiannya
Cache terdiri dari sejumlah set
Setiap set berisi sejumlah baris
Peta blok tertentu ke baris apa pun dalam satu set tertentu
misalnya 2 baris per set
2 cara pemetaan asosiatif
Blok yang diberikan dapat berada di salah satu dari 2 baris hanya dalam satu set



(a) v associative-mapped caches

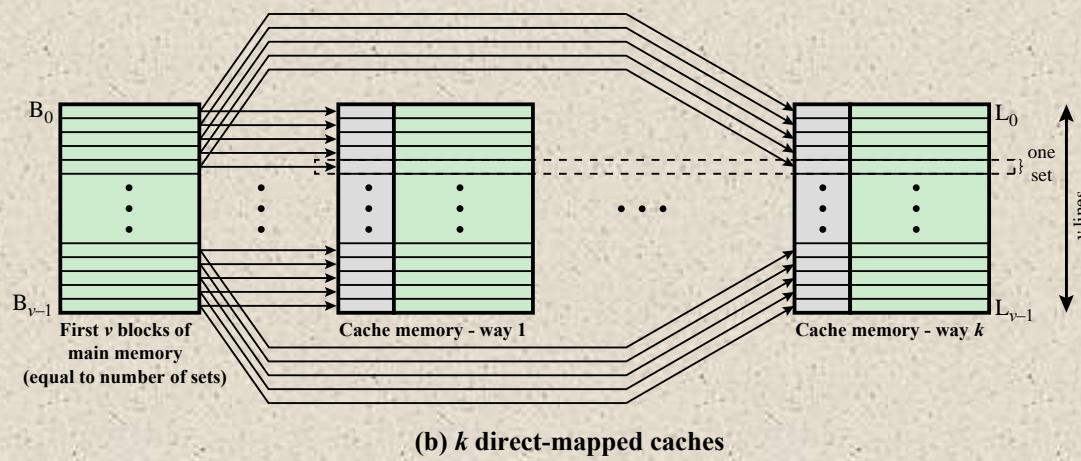


Figure 4.13 Mapping From Main Memory to Cache:
 k -way Set Associative

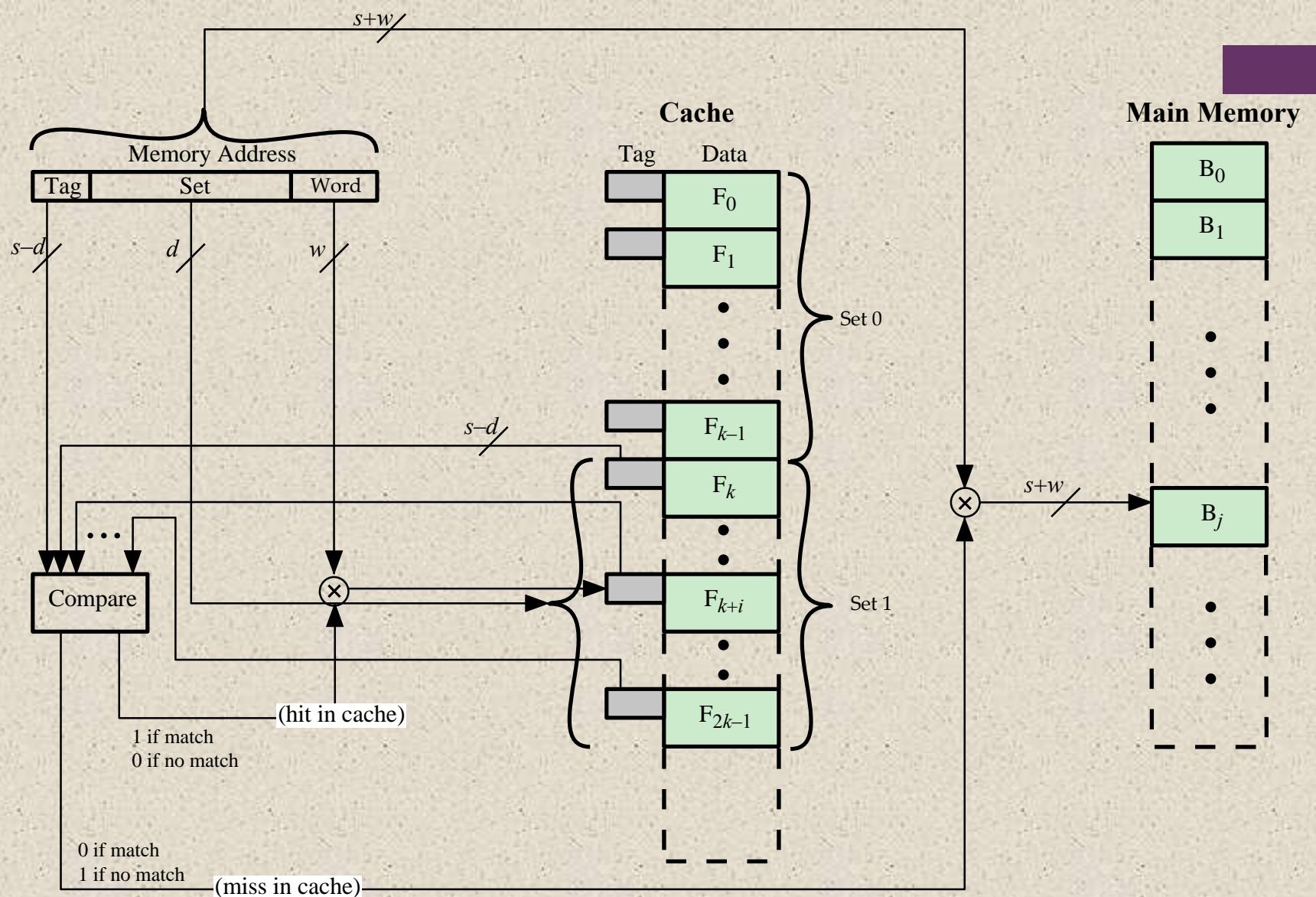


Figure 4.14 k -Way Set Associative Cache Organization



Set Associative Mapping Summary

- Address length = $(s + w)$ bits
- Number of addressable units = 2^{s+w} words or bytes
- Block size = line size = 2^w words or bytes
- Number of blocks in main memory = $2^{s+w}/2^w=2^s$
- Number of lines in set = k
- Number of sets = $v = 2^d$
- Number of lines in cache = $m=kv = k * 2^d$
- Size of cache = $k * 2^{d+w}$ words or bytes
- Size of tag = $(s - d)$ bits



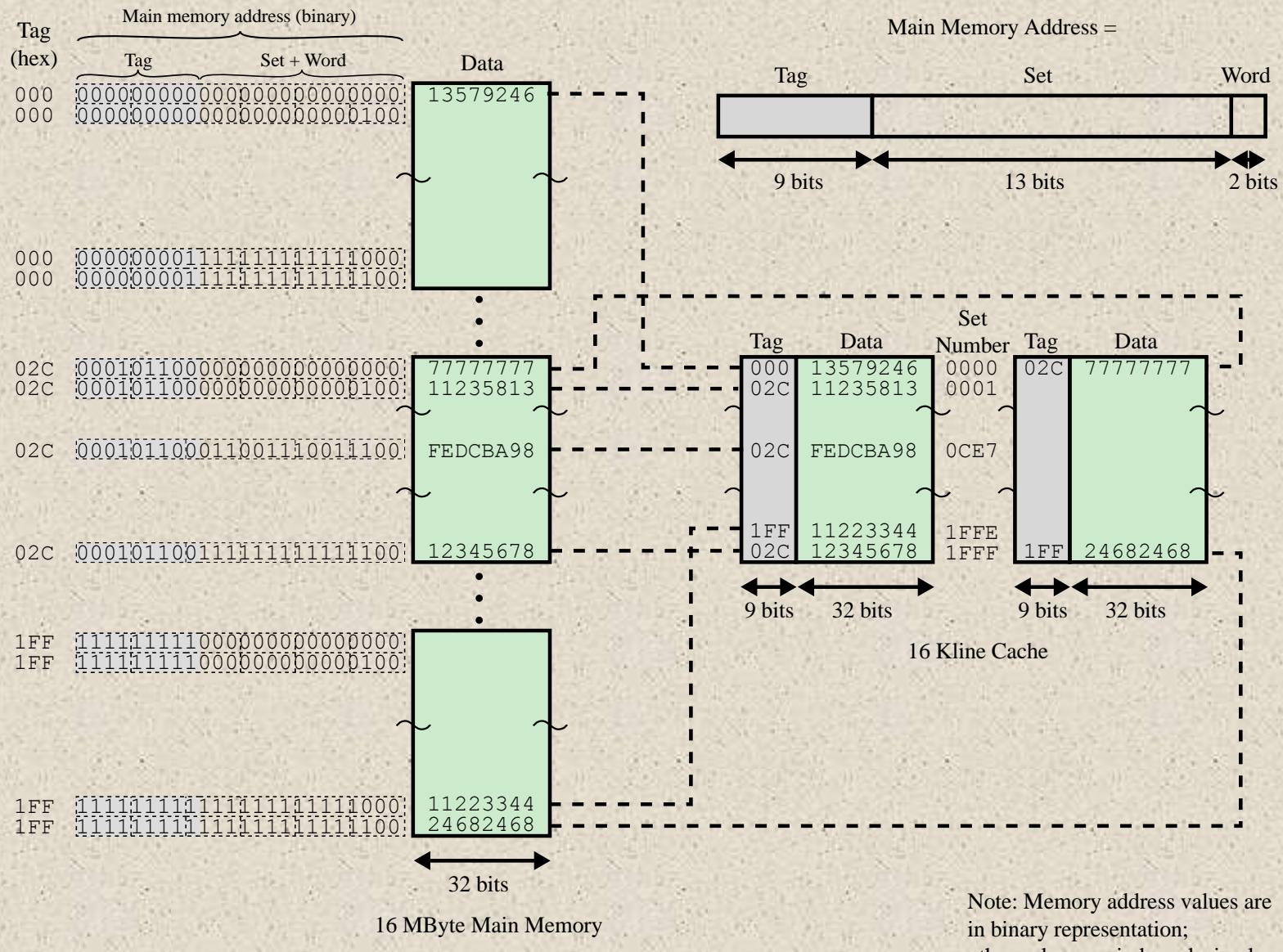


Figure 4.15 Two-Way Set Associative Mapping Example

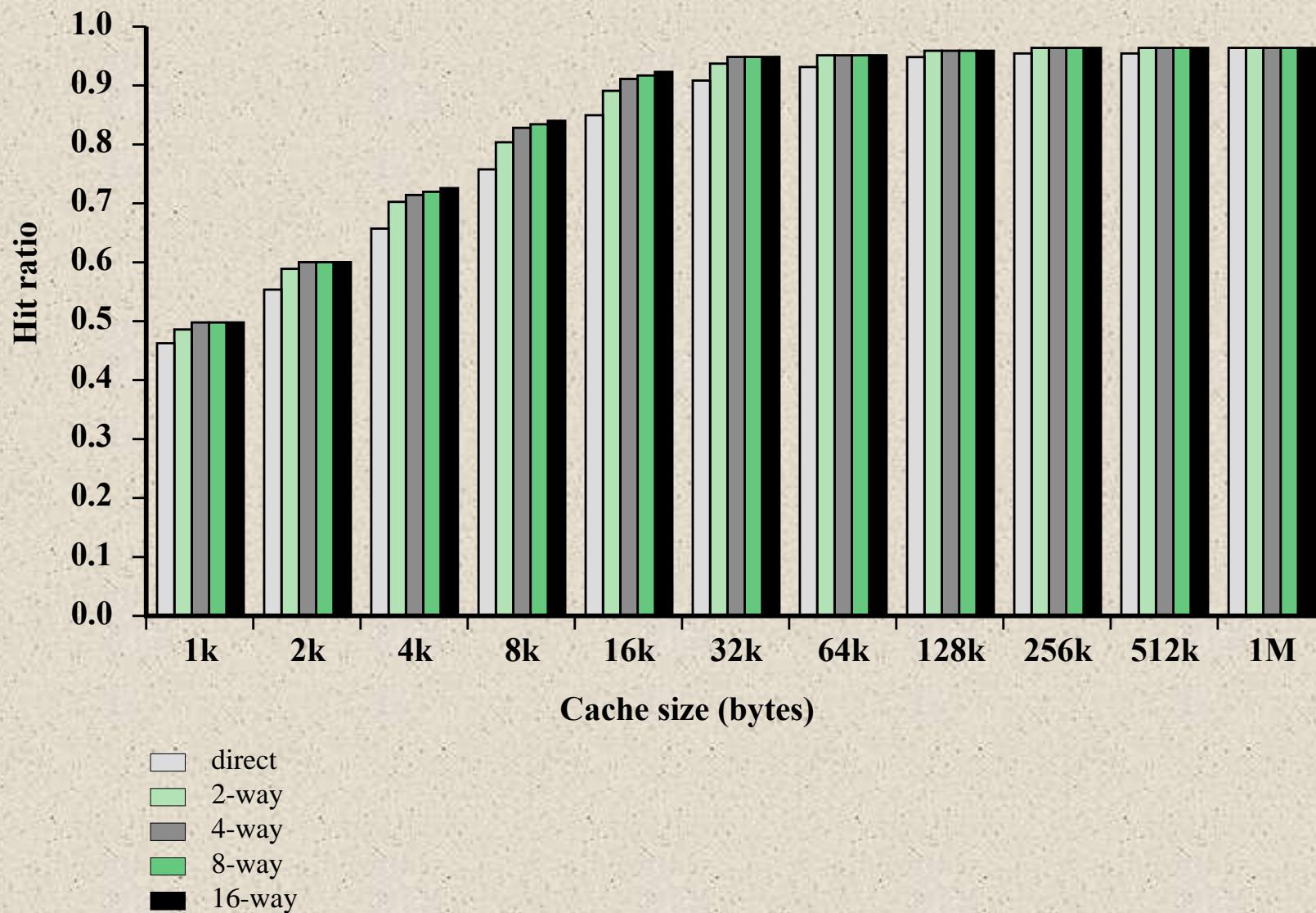
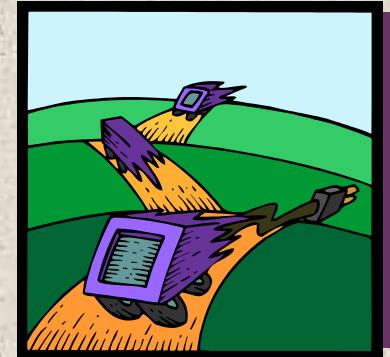


Figure 4.16 Varying Associativity over Cache Size

Replacement Algorithms



- Once the cache has been filled, when a new block is brought into the cache, one of the existing blocks must be replaced
- For direct mapping there is only one possible line for any particular block and no choice is possible
- For the associative and set-associative techniques a replacement algorithm is needed
- To achieve high speed, an algorithm must be implemented in hardware

Setelah cache diisi, ketika blok baru dibawa ke dalam cache, salah satu blok yang ada harus diganti
Untuk pemetaan langsung hanya ada satu jalur yang mungkin untuk setiap blok tertentu dan tidak ada pilihan yang mungkin
Untuk teknik asosiatif dan set-asosiatif a algoritma penggantian diperlukan
Untuk mencapai kecepatan tinggi, algoritma harus diimplementasikan dalam perangkat keras

+ The most common replacement algorithms are:

- Least recently used (LRU)
 - Most effective
 - Replace that block in the set that has been in the cache longest with no reference to it
 - Because of its simplicity of implementation, LRU is the most popular replacement algorithm
- First-in-first-out (FIFO)
 - Replace that block in the set that has been in the cache longest
 - Easily implemented as a round-robin or circular buffer technique
- Least frequently used (LFU)
 - Replace that block in the set that has experienced the fewest references
 - Could be implemented by associating a counter with each line

Write Policy

When a block that is resident in the cache is to be replaced there are two cases to consider:

If the old block in the cache has not been altered then it may be overwritten with a new block without first writing out the old block

If at least one write operation has been performed on a word in that line of the cache then main memory must be updated by writing the line of cache out to the block of memory before bringing in the new block

There are two problems to contend with:

More than one device may have access to main memory

A more complex problem occurs when multiple processors are attached to the same bus and each processor has its own local cache - if a word is altered in one cache it could conceivably invalidate a word in other caches

Write Through and Write Back

Tulis melalui

Teknik paling sederhana

Semua operasi penulisan dibuat untuk memori utama serta cache

Kerugian utama dari teknik ini adalah bahwa hal itu menghasilkan substansial lalu lintas memori dan dapat membuat kemacetan

- Write through
 - Simplest technique
 - All write operations are made to main memory as well as to the cache
 - The main disadvantage of this technique is that it generates substantial memory traffic and may create a bottleneck
- Write back
 - Minimizes memory writes
 - Updates are made only in the cache
 - Portions of main memory are invalid and hence accesses by I/O modules can be allowed only through the cache
 - This makes for complex circuitry and a potential bottleneck

Line Size

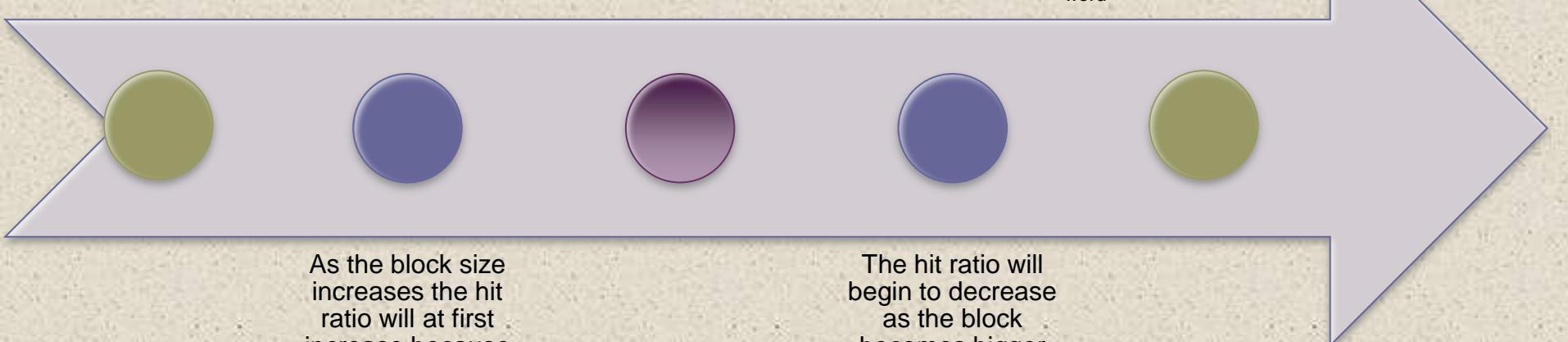


When a block of data is retrieved and placed in the cache not only the desired word but also some number of adjacent words are retrieved

As the block size increases more useful data are brought into the cache

Two specific effects come into play:

- Larger blocks reduce the number of blocks that fit into a cache
- As a block becomes larger each additional word is farther from the requested word



As the block size increases the hit ratio will at first increase because of the principle of locality

The hit ratio will begin to decrease as the block becomes bigger and the probability of using the newly fetched information becomes less than the probability of reusing the information that has to be replaced

Multilevel Caches

- As logic density has increased it has become possible to have a cache on the same chip as the processor
- The on-chip cache reduces the processor's external bus activity and speeds up execution time and increases overall system performance
 - When the requested instruction or data is found in the on-chip cache, the bus access is eliminated
 - On-chip cache accesses will complete appreciably faster than would even zero-wait state bus cycles
 - During this period the bus is free to support other transfers
- Two-level cache:
 - Internal cache designated as level 1 (L1)
 - External cache designated as level 2 (L2)
- Potential savings due to the use of an L2 cache depends on the hit rates in both the L1 and L2 caches
- The use of multilevel caches complicates all of the design issues related to caches, including size, replacement algorithm, and write policy

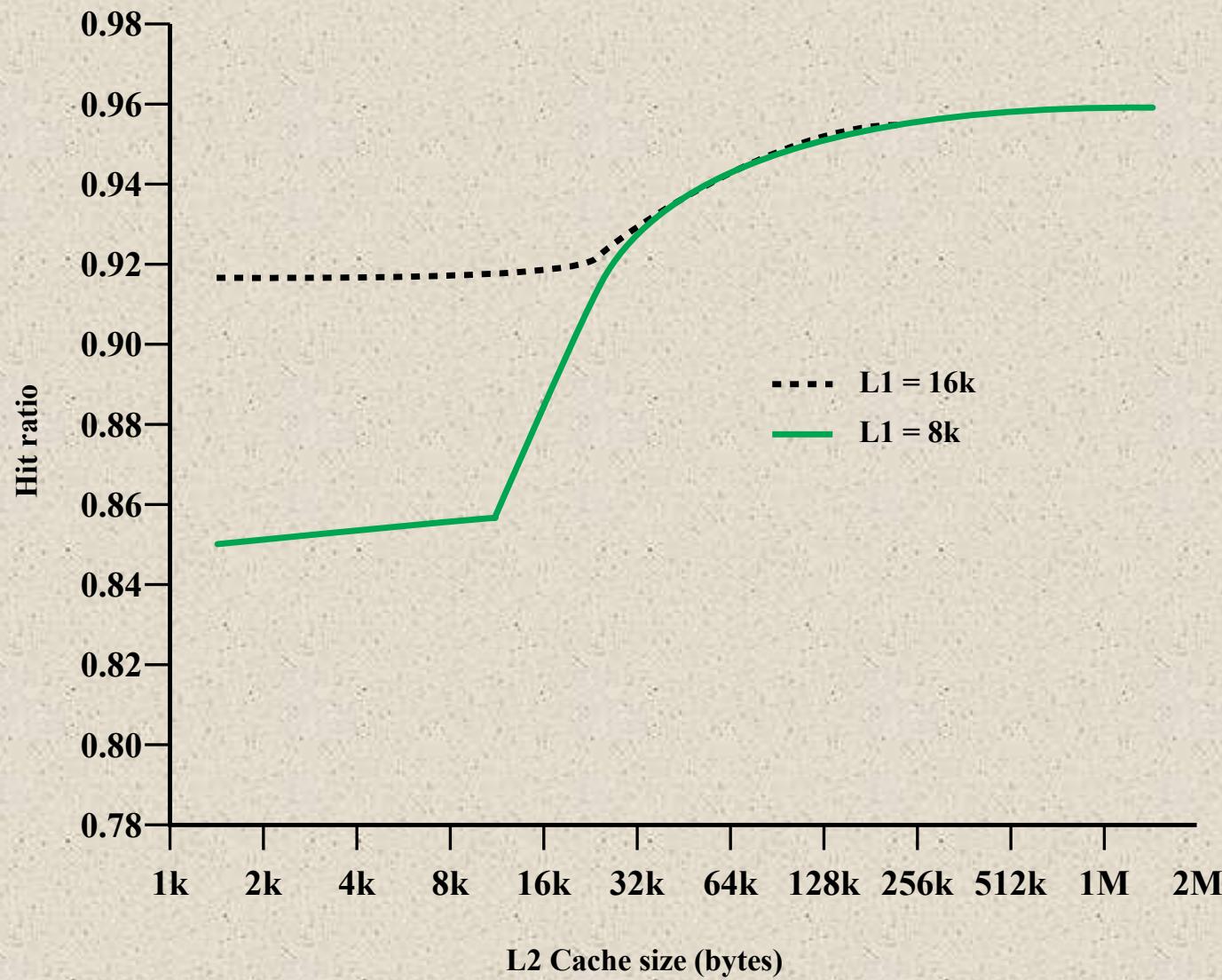


Figure 4.17 Total Hit Ratio (L1 and L2) for 8 Kbyte and 16 Kbyte L1



Unified Versus Split Caches

- Has become common to split cache:
 - One dedicated to instructions
 - One dedicated to data
 - Both exist at the same level, typically as two L1 caches
- Advantages of unified cache:
 - Higher hit rate
 - Balances load of instruction and data fetches automatically
 - Only one cache needs to be designed and implemented
- Trend is toward split caches at the L1 and unified caches for higher levels
- Advantages of split cache:
 - Eliminates cache contention between instruction fetch/decode unit and execution unit
 - Important in pipelining

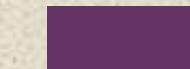


Table 4.4

Intel
Cache
Evolution

Problem	Solution	Processor on which Feature First Appears
External memory slower than the system bus.	Add external cache using faster memory technology.	386
Increased processor speed results in external bus becoming a bottleneck for cache access.	Move external cache on-chip, operating at the same speed as the processor.	486
Internal cache is rather small, due to limited space on chip	Add external L2 cache using faster technology than main memory	486
Contention occurs when both the Instruction Prefetcher and the Execution Unit simultaneously require access to the cache. In that case, the Prefetcher is stalled while the Execution Unit's data access takes place.	Create separate data and instruction caches.	Pentium
Increased processor speed results in external bus becoming a bottleneck for L2 cache access.	Create separate back-side bus that runs at higher speed than the main (front-side) external bus. The BSB is dedicated to the L2 cache.	Pentium Pro
	Move L2 cache on to the processor chip.	Pentium II
Some applications deal with massive databases and must have rapid access to large amounts of data. The on-chip caches are too small.	Add external L3 cache.	Pentium III
	Move L3 cache on-chip.	Pentium 4

(Table is on page 150 in the textbook.)

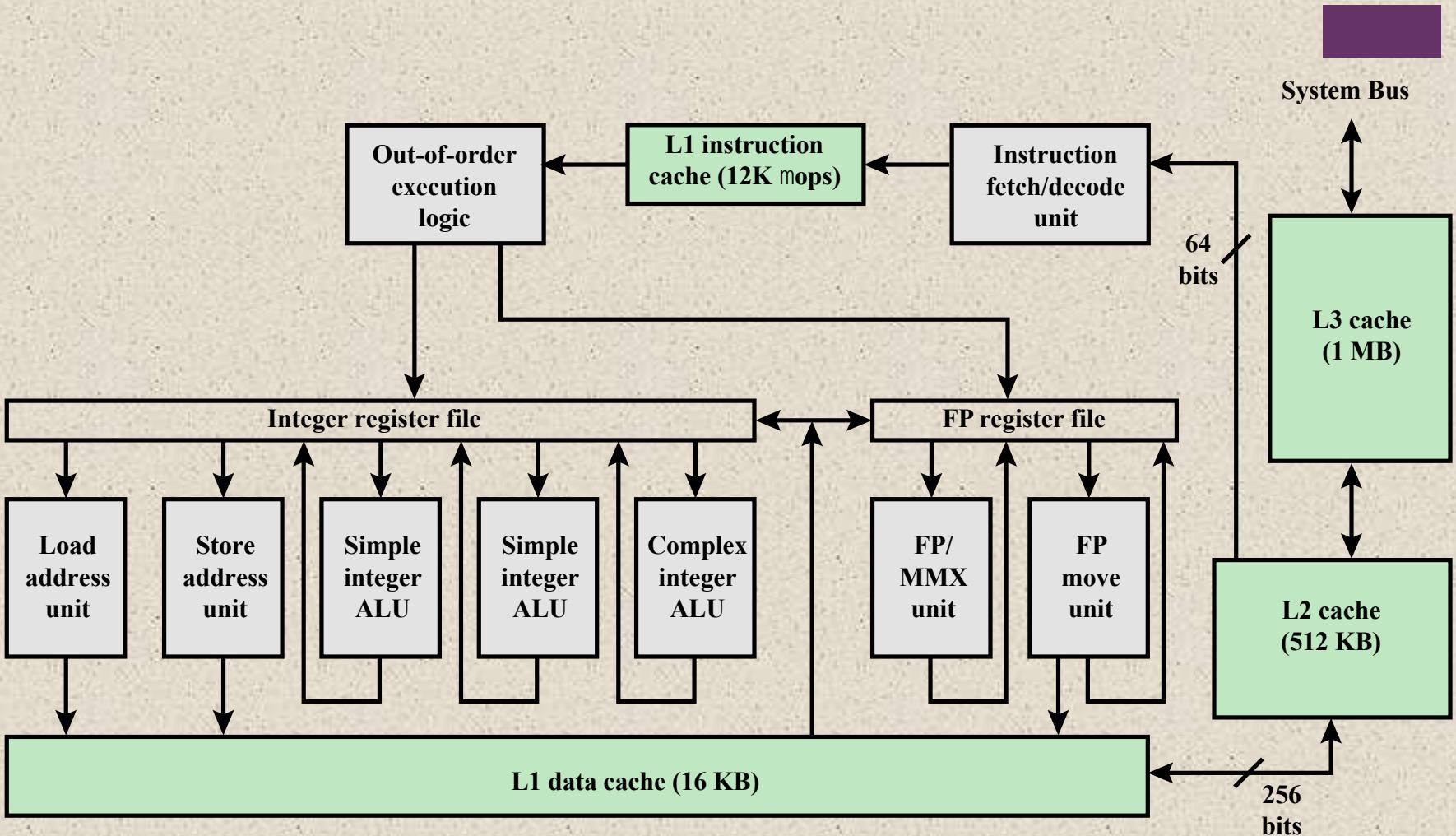


Figure 4.18 Pentium 4 Block Diagram

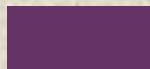


Table 4.5 Pentium 4 Cache Operating Modes

Control Bits		Operating Mode		
CD	NW	Cache Fills	Write Throughs	Invalidates
0	0	Enabled	Enabled	Enabled
1	0	Disabled	Enabled	Enabled
1	1	Disabled	Disabled	Disabled

Note: CD = 0; NW = 1 is an invalid combination.

+

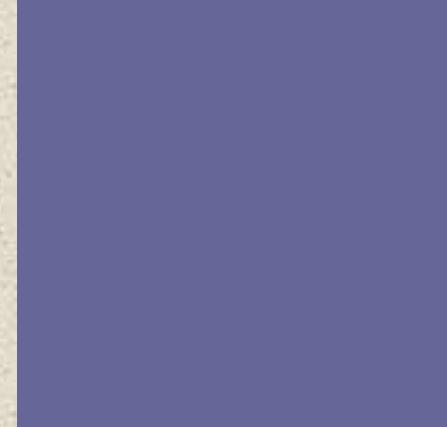
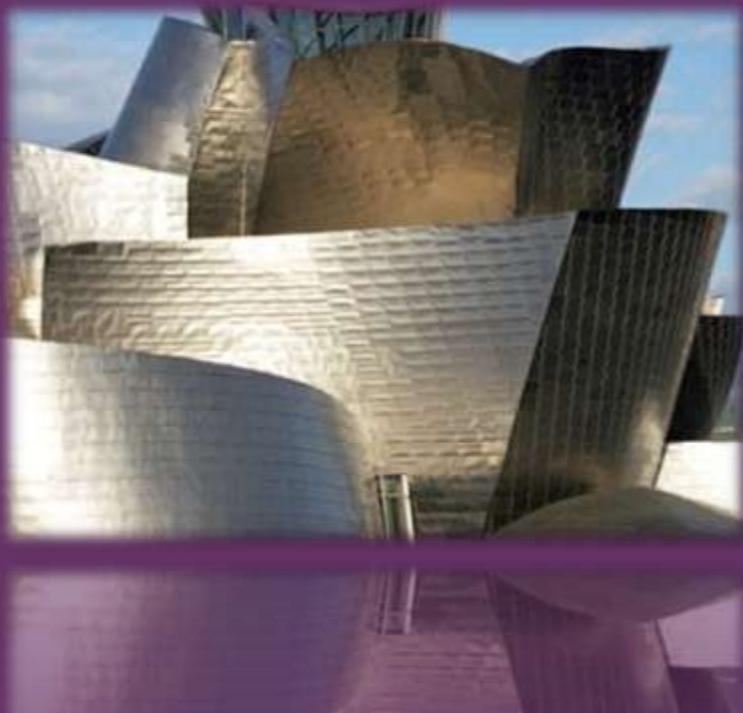
Summary

Chapter 4

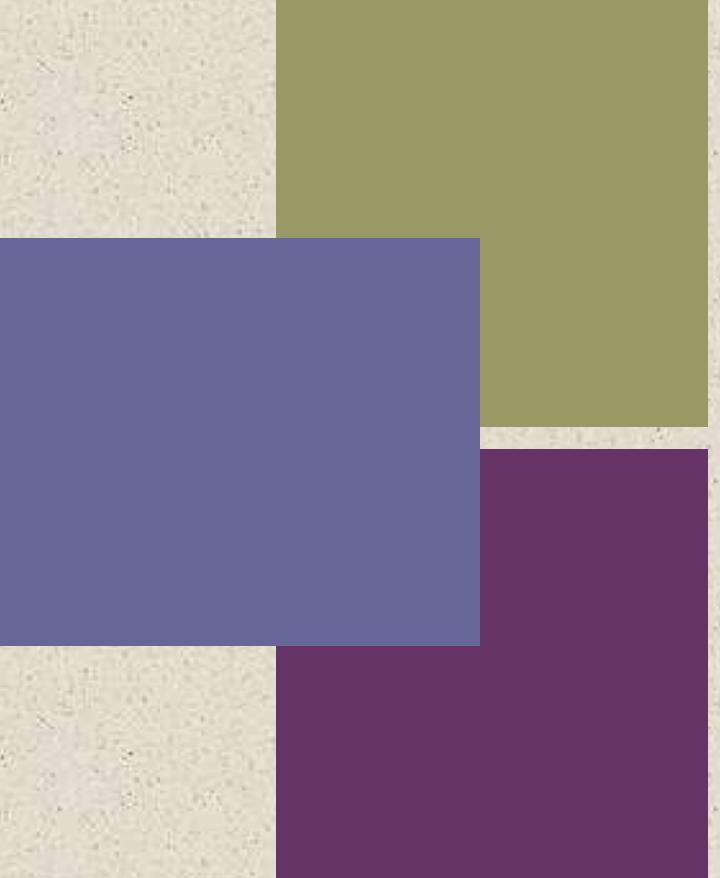
- Computer memory system overview
 - Characteristics of Memory Systems
 - Memory Hierarchy
- Cache memory principles
- Pentium 4 cache organization
- Elements of cache design
 - Cache addresses
 - Cache size
 - Mapping function
 - Replacement algorithms
 - Write policy
 - Line size
 - Number of caches

Cache Memory

+

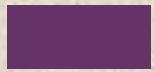


William Stallings
Computer Organization
and Architecture
10th Edition

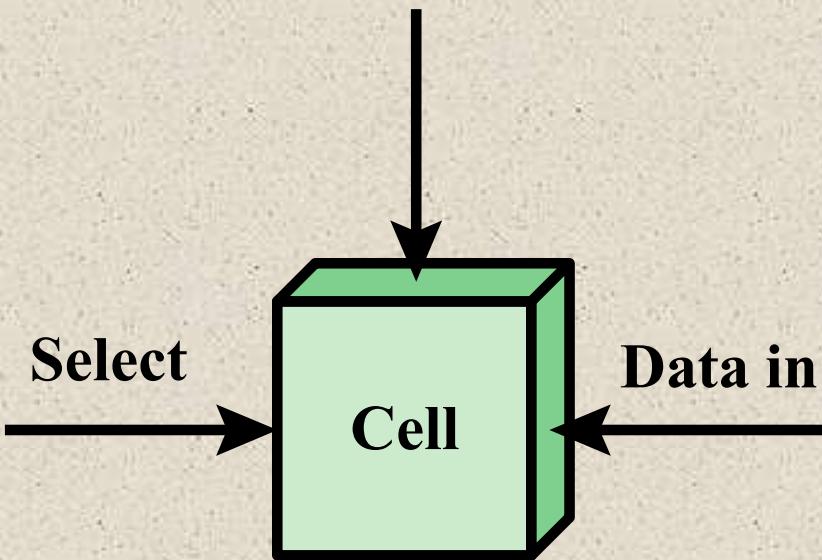


+ Chapter 5

Internal Memory

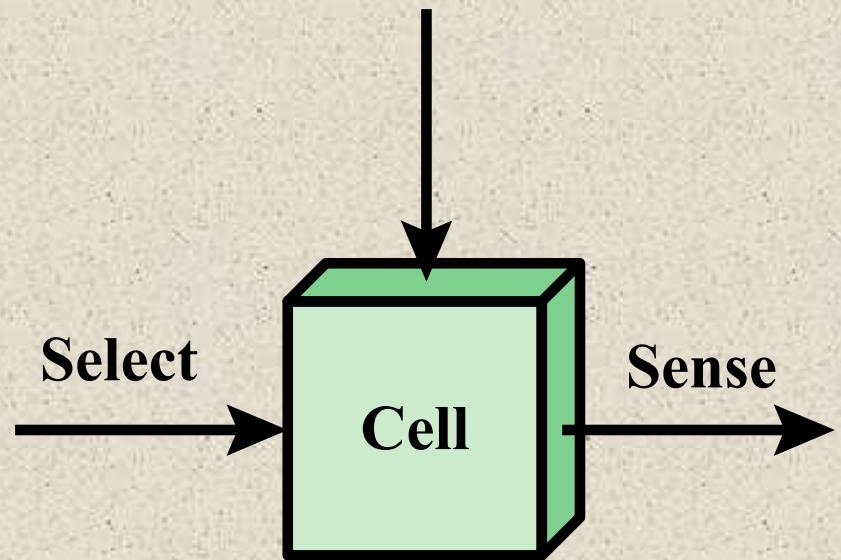


Control



(a) Write

Control



(b) Read

Figure 5.1 Memory Cell Operation



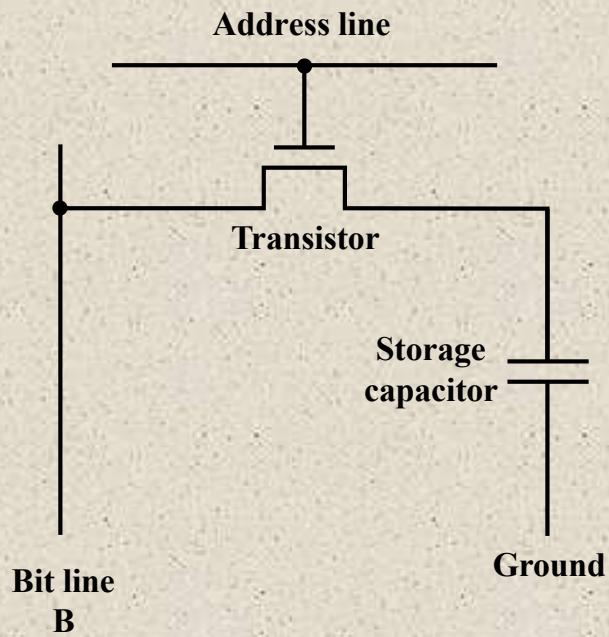
Memory Type	Category	Erasure	Write Mechanism	Volatility
Random-access memory (RAM)	Read-write memory	Electrically, byte-level	Electrically	Volatile
Read-only memory (ROM)	Read-only memory	Not possible	Masks	
Programmable ROM (PROM)		UV light, chip-level		Nonvolatile
Erasable PROM (EPROM)	Read-mostly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

Table 5.1
Semiconductor Memory Types

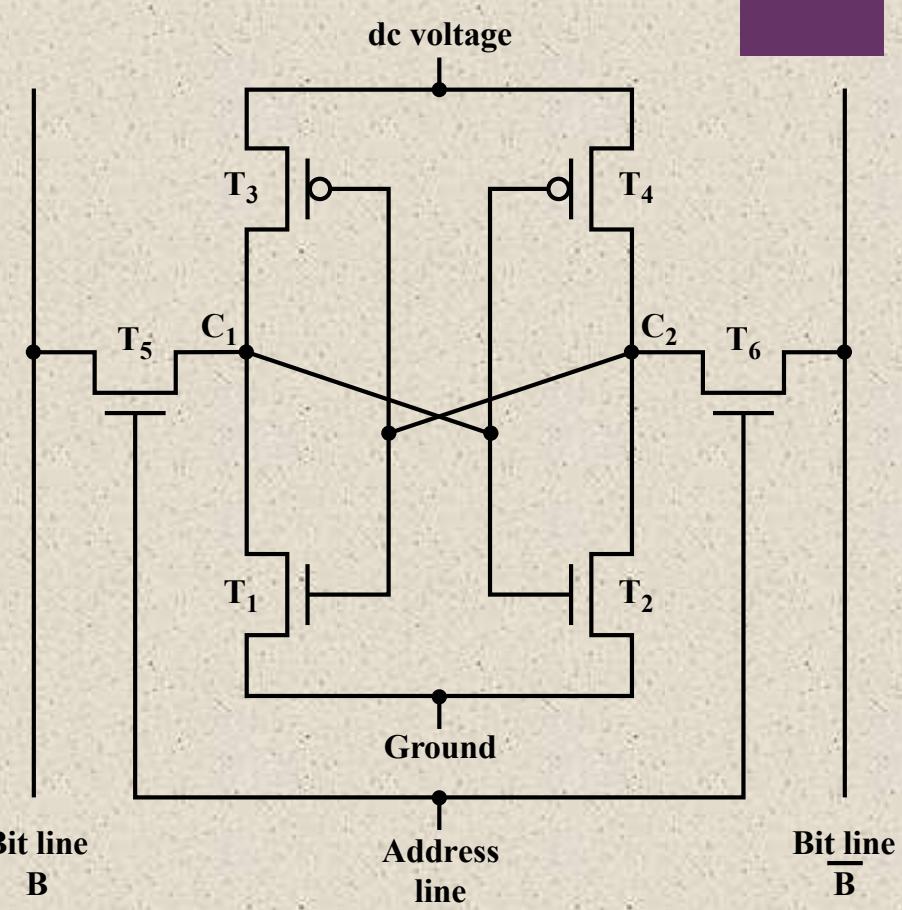


Dynamic RAM (DRAM)

- RAM technology is divided into two technologies:
 - Dynamic RAM (DRAM)
 - Static RAM (SRAM)
- DRAM
 - Made with cells that store data as charge on capacitors
 - 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



(a) Dynamic RAM (DRAM) cell



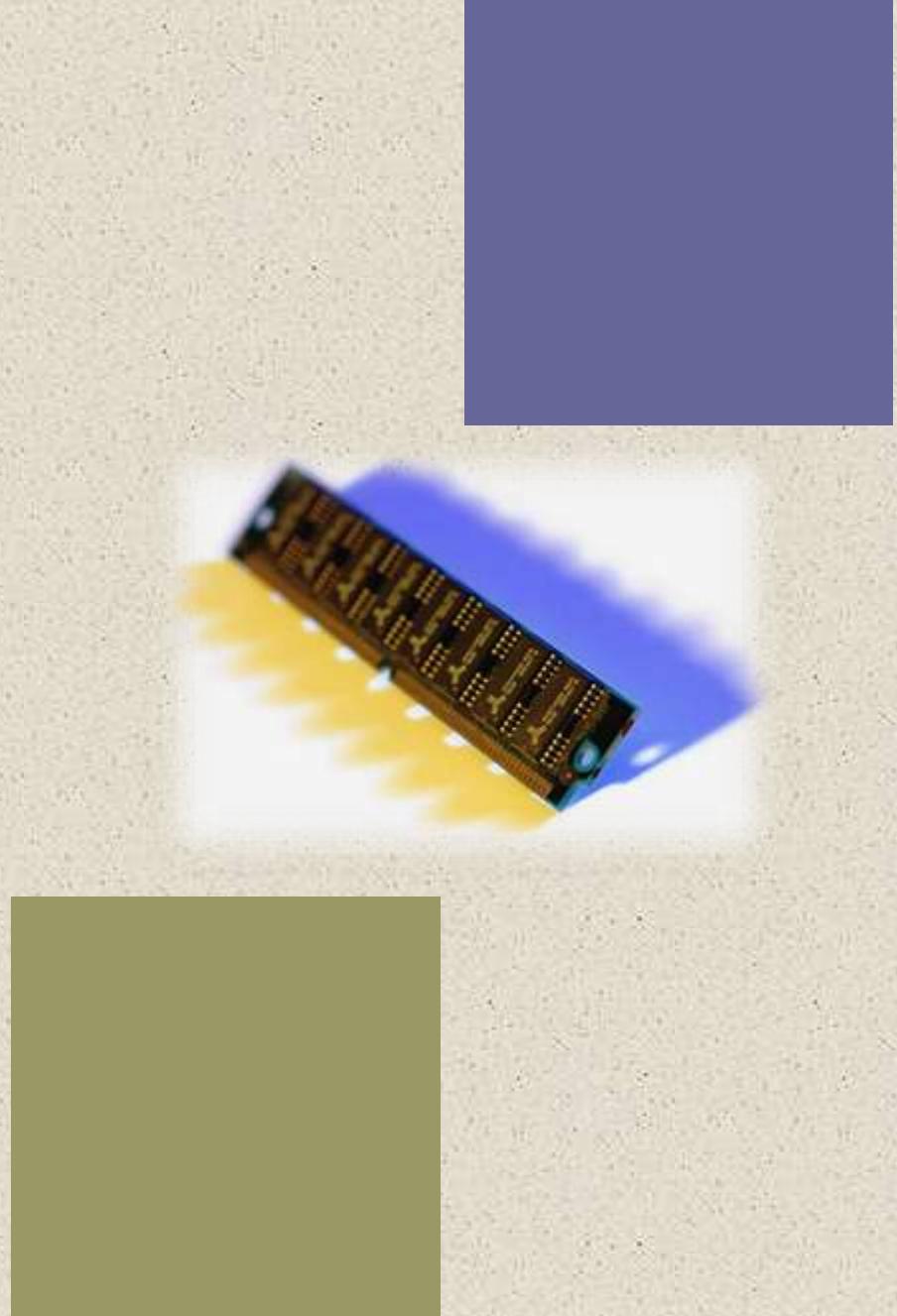
(b) Static RAM (SRAM) cell

Figure 5.2 Typical Memory Cell Structures



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 versus DRAM

- Both volatile
 - Power must be continuously supplied to the memory to preserve the bit values
- Dynamic cell
 - 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)

SRAM

DRAM

Read Only Memory (ROM)

- Contains a permanent pattern of data that cannot be changed or added to
- No power source is required to maintain the bit values in 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 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

Read-Mostly Memory

EPROM

Erasable programmable
read-only memory

Erasure process can be
performed repeatedly

More expensive than PROM
but it has the advantage of
the multiple update
capability

EEPROM

Electrically erasable
programmable read-only
memory

Can be written into at any
time without erasing prior
contents

Combines the advantage of
non-volatility with the
flexibility of being
updatable in place

More expensive than
EPROM

Flash Memory

Intermediate between
EPROM and EEPROM in
both cost and functionality

Uses an electrical erasing
technology, does not
provide byte-level erasure

Microchip is organized so
that a section of memory
cells are erased in a single
action or “flash”

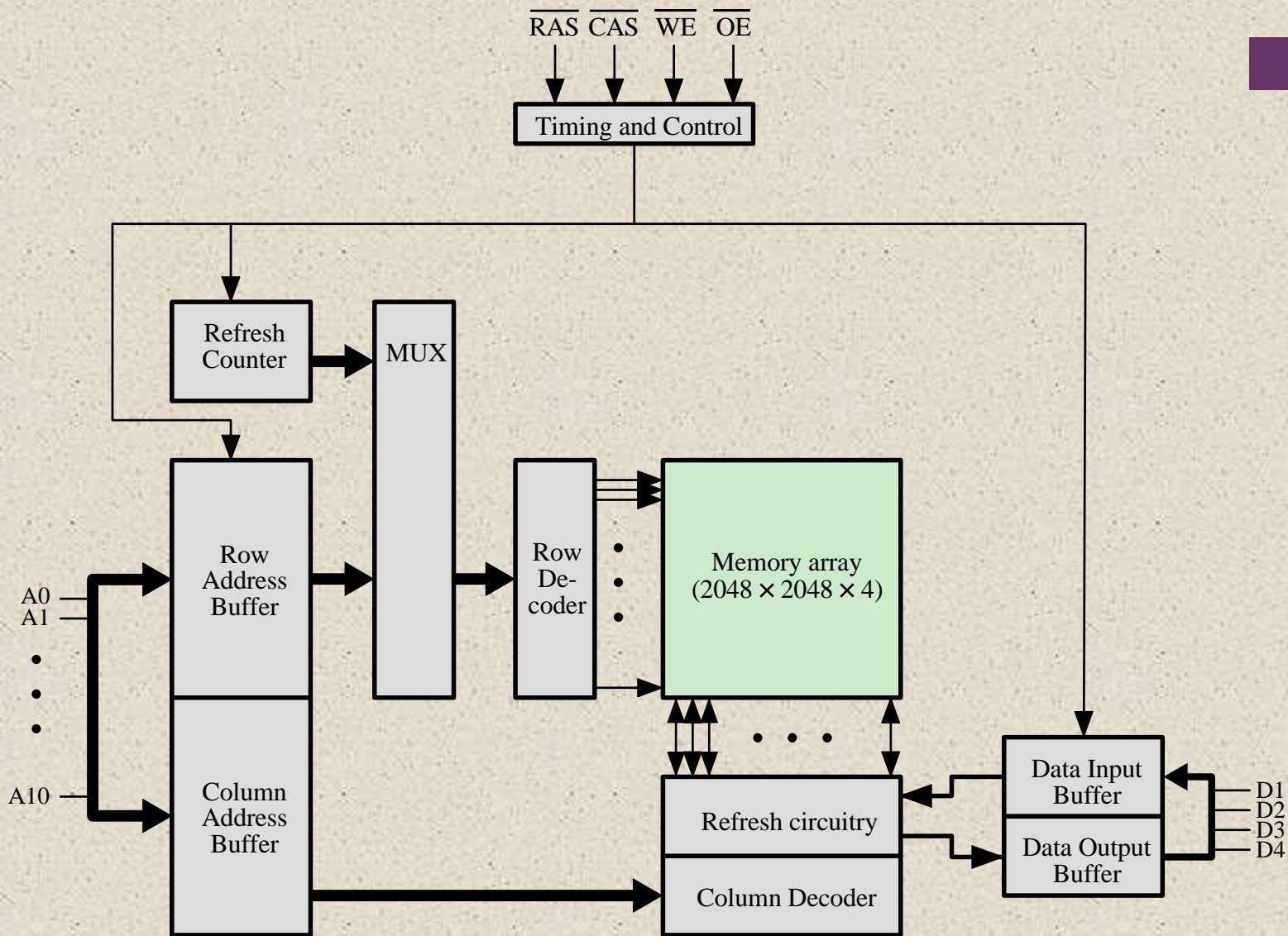
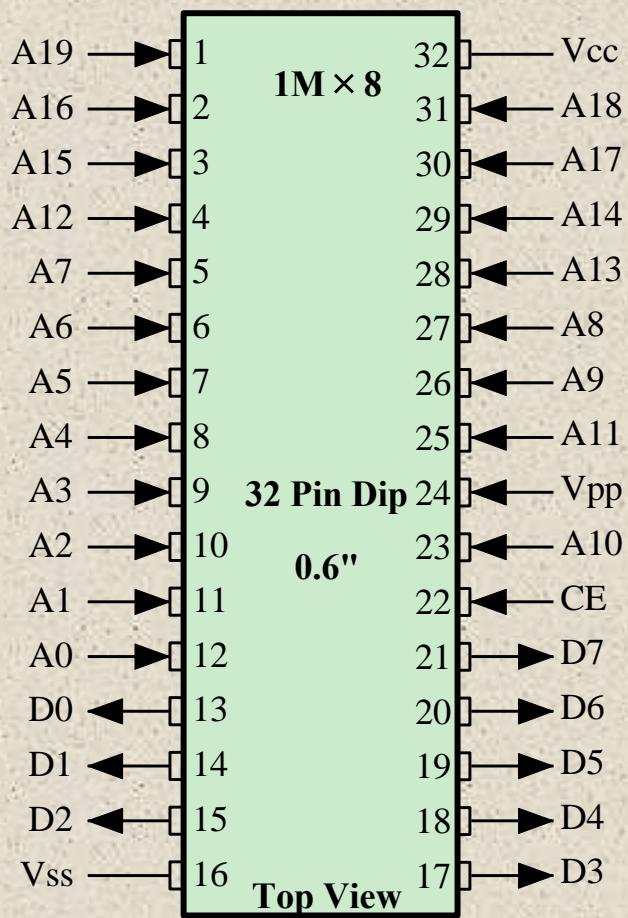
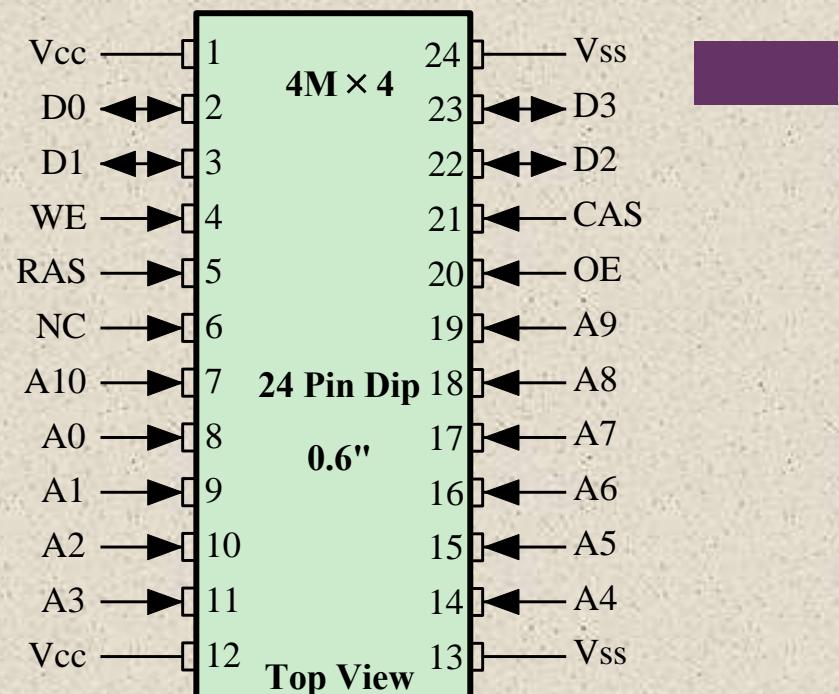


Figure 5.3 Typical 16 Megabit DRAM (4M × 4)



(a) 8 Mbit EPROM



(b) 16 Mbit DRAM

Figure 5.4 Typical Memory Package Pins and Signals

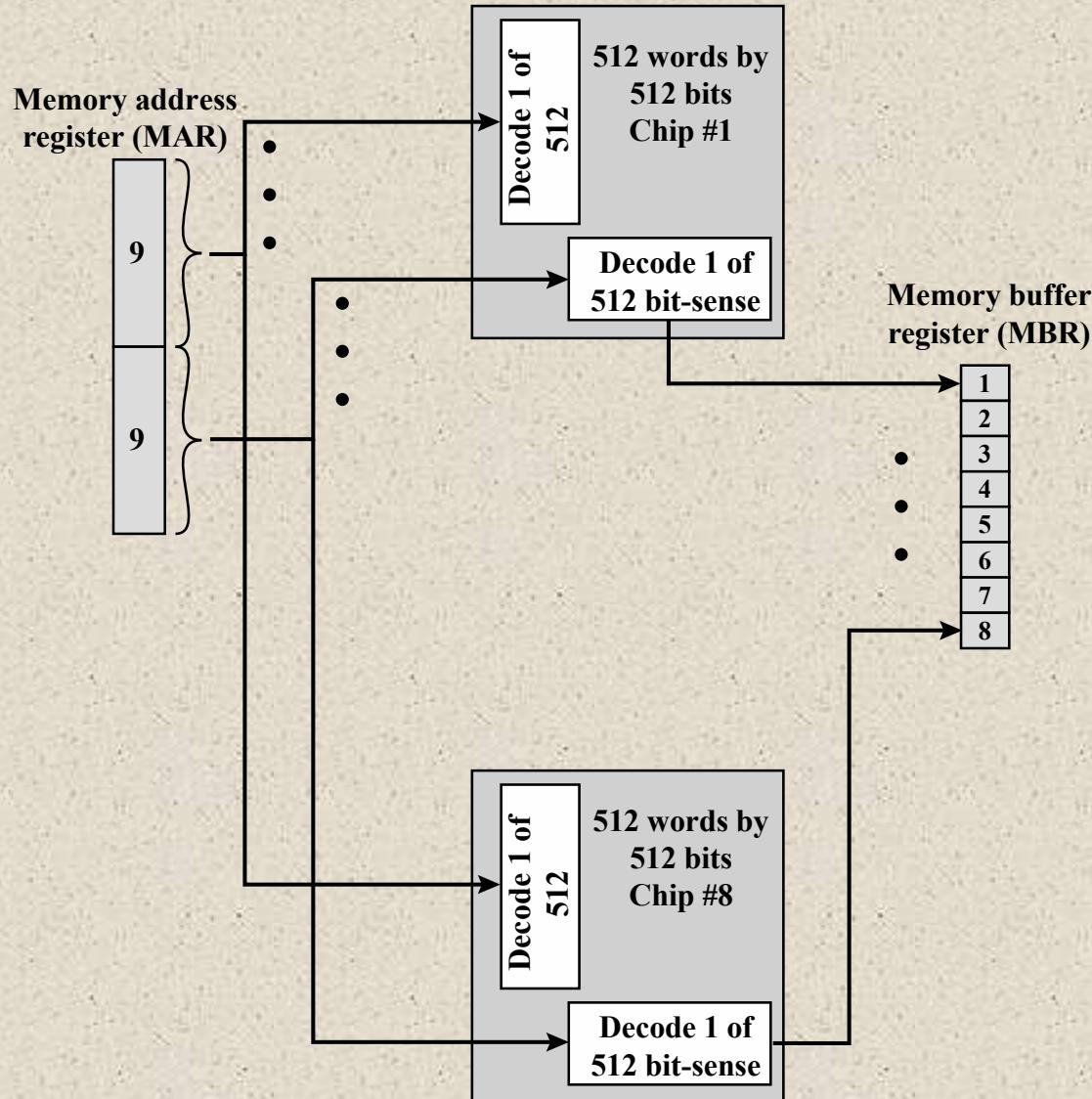


Figure 5.5 256-KByte Memory Organization

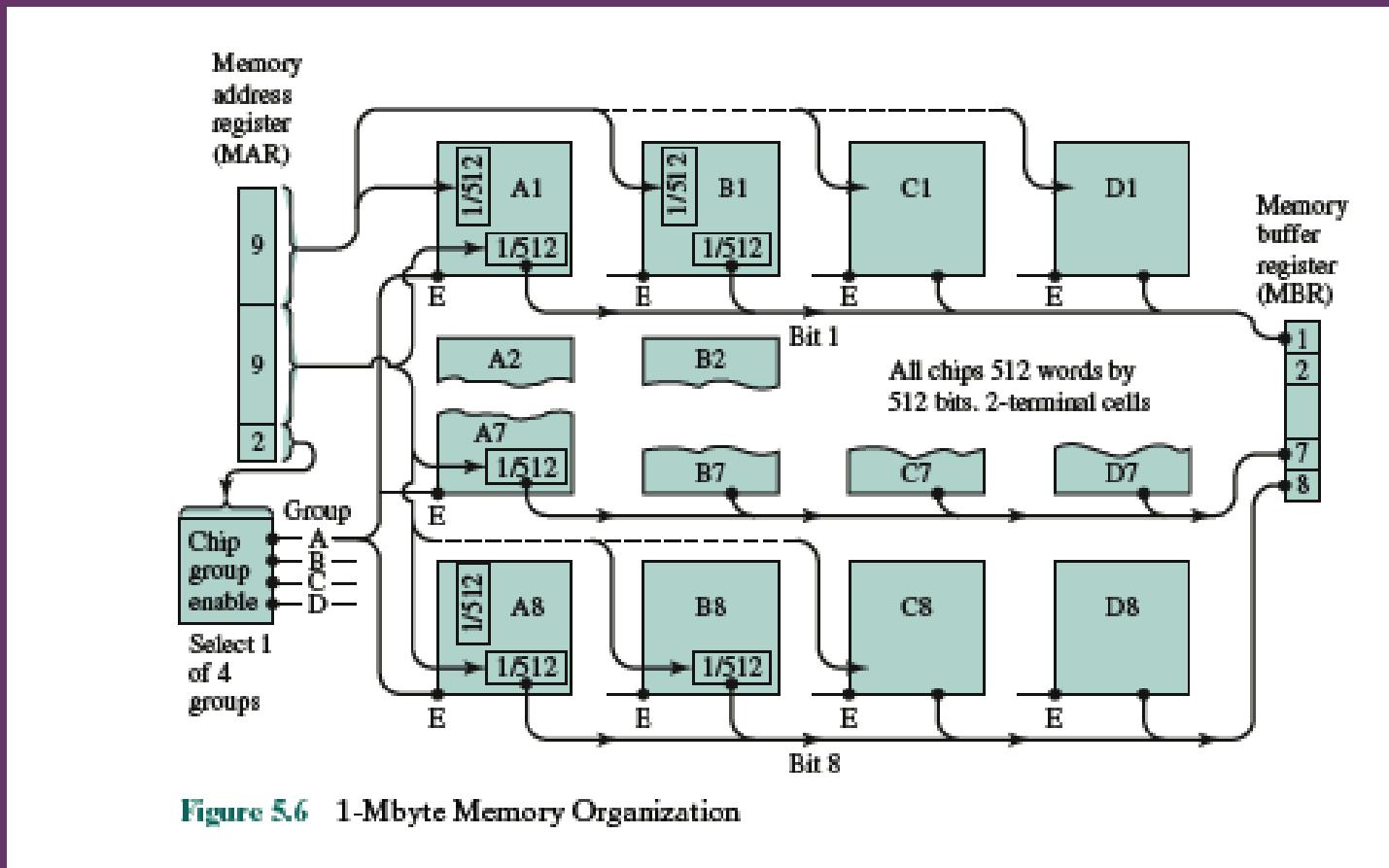


Figure 5.6 1-Mbyte Memory Organization

Interleaved Memory

Composed of a collection of DRAM chips

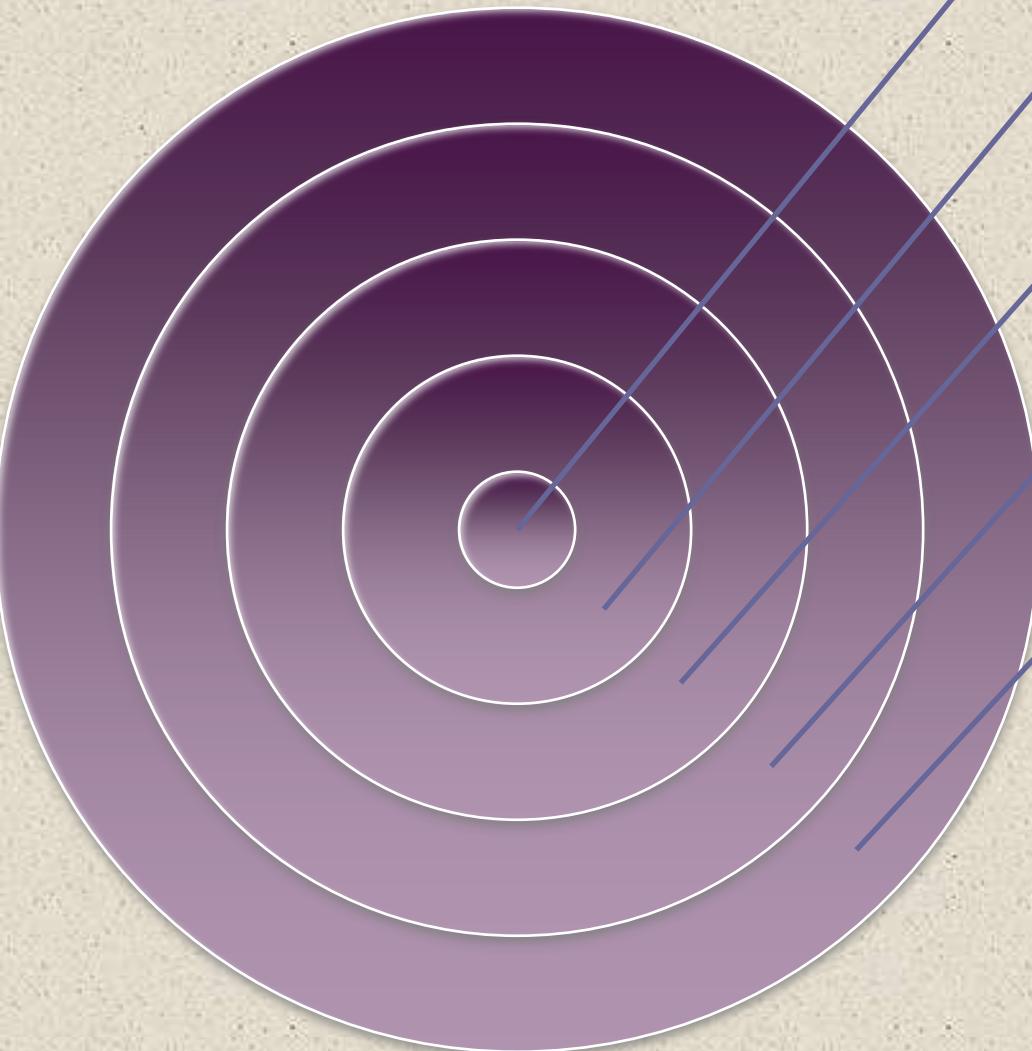
Terdiri dari koleksi Chip DRAM

Grouped together to form a *memory bank*

Each bank is independently able to service a memory read or write request

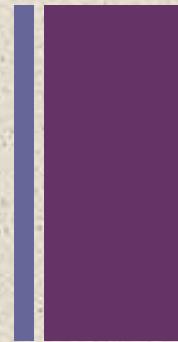
K banks can service K requests simultaneously, increasing memory read or write rates by a factor of K

If consecutive words of memory are stored in different banks, the transfer of a block of memory is speeded up





Error Correction



- Hard Failure
 - Permanent physical defect
 - Memory cell or cells affected cannot reliably store data but become stuck at 0 or 1 or switch erratically between 0 and 1
 - Can be caused by:
 - Harsh environmental abuse
 - Manufacturing defects
 - Wear
- Soft Error
 - Random, non-destructive event that alters the contents of one or more memory cells
 - No permanent damage to memory
 - Can be caused by:
 - Power supply problems
 - Alpha particles

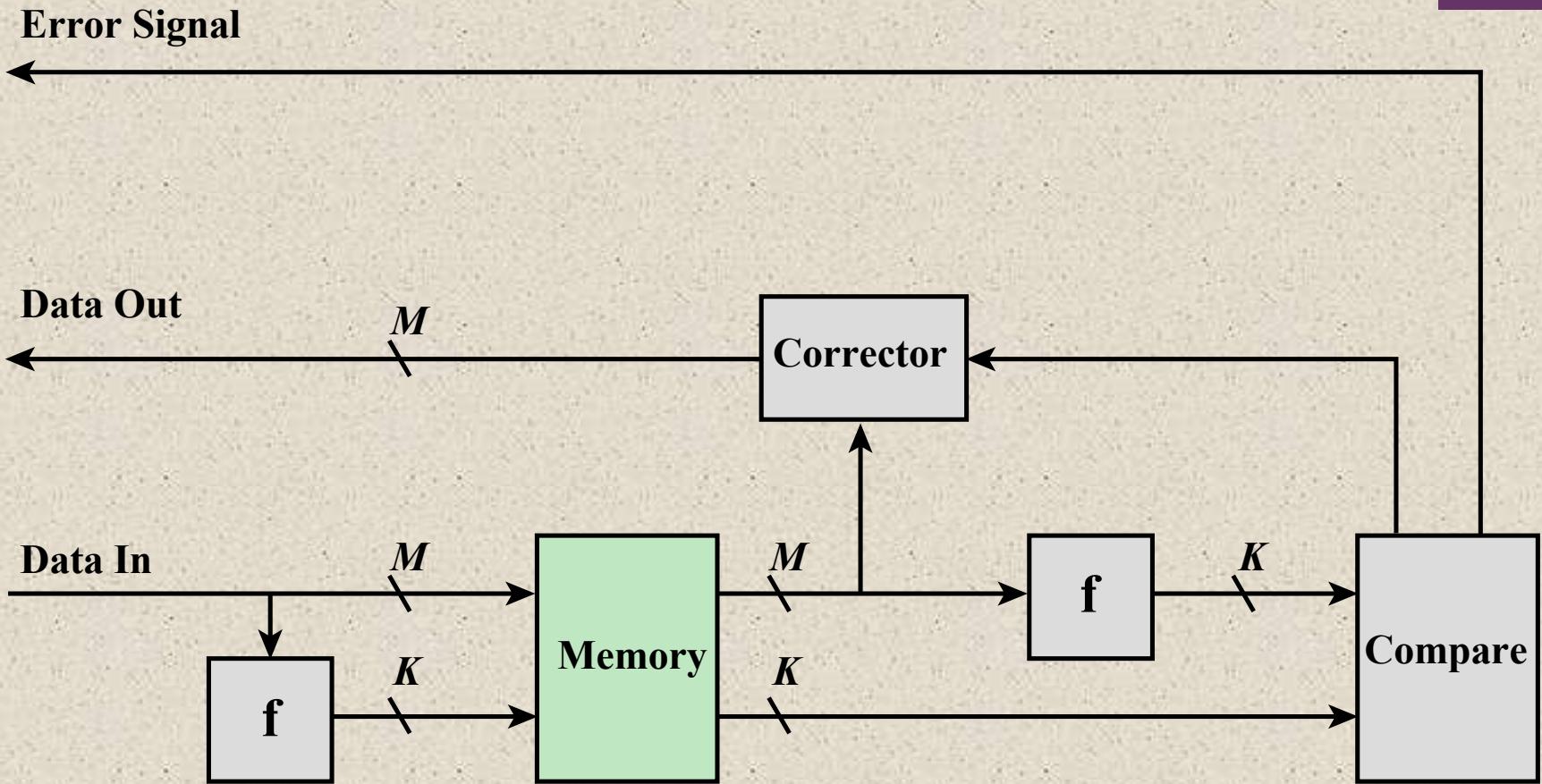


Figure 5.7 Error-Correcting Code Function

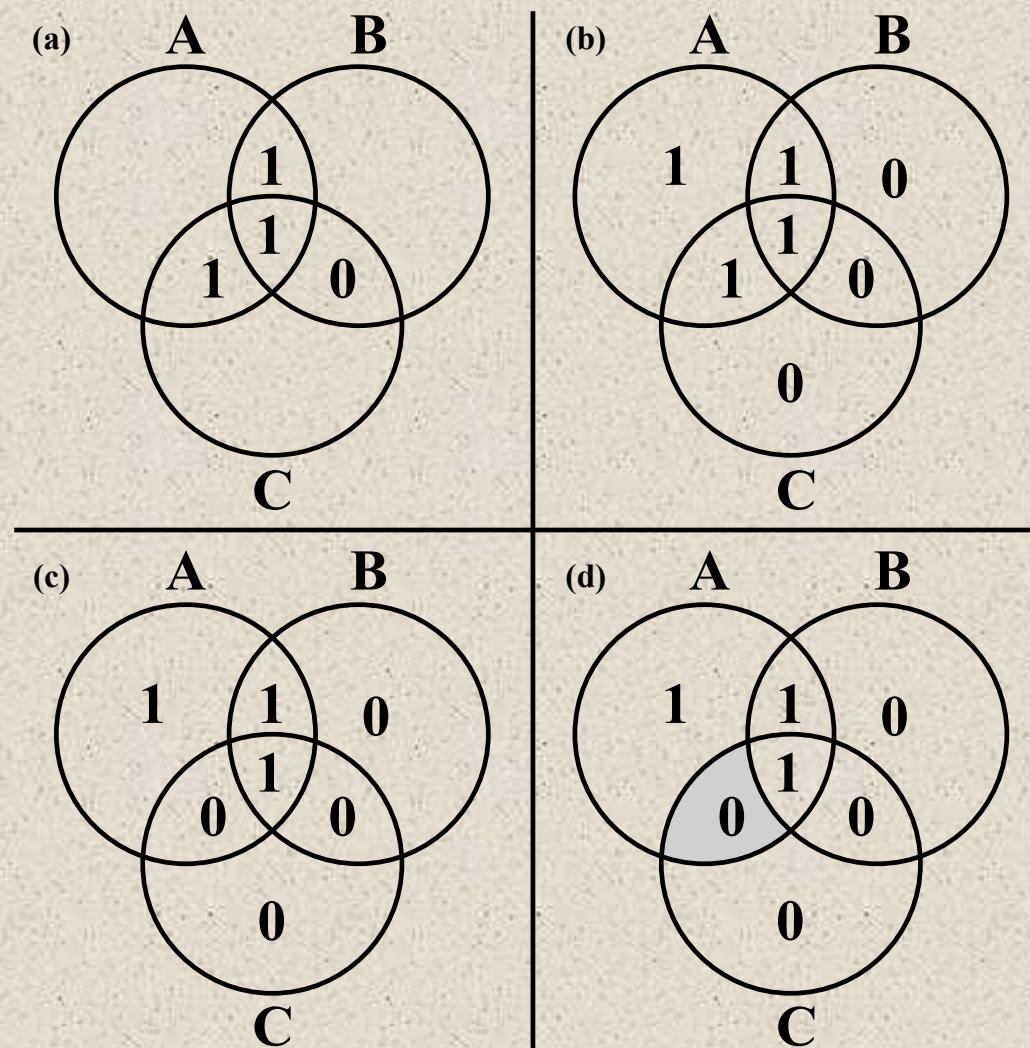
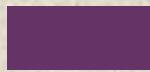
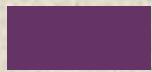
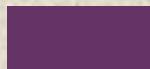


Figure 5.8 Hamming Error-Correcting Code



	Single-Error Correction		Single-Error Correction/ Double-Error Detection	
Data Bits	Check Bits	% Increase	Check Bits	% Increase
8	4	50	5	62.5
16	5	31.25	6	37.5
32	6	18.75	7	21.875
64	7	10.94	8	12.5
128	8	6.25	9	7.03
256	9	3.52	10	3.91

Table 5.2
Increase in Word Length with Error Correction



Bit Position	12	11	10	9	8	7	6	5	4	3	2	1
Position Number	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Data Bit	D8	D7	D6	D5		D4	D3	D2		D1		
Check Bit					C8				C4		C2	C1

Figure 5.9 Layout of Data Bits and Check Bits

Bit position	12	11	10	9	8	7	6	5	4	3	2	1
Position number	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Data bit	D8	D7	D6	D5		D4	D3	D2		D1		
Check bit					C8				C4		C2	C1
Word stored as	0	0	1	1	0	1	0	0	1	1	1	1
Word fetched as	0	0	1	1	0	1	1	0	1	1	1	1
Position Number	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Check Bit					0				0		0	1

Figure 5.10 Check Bit Calculation

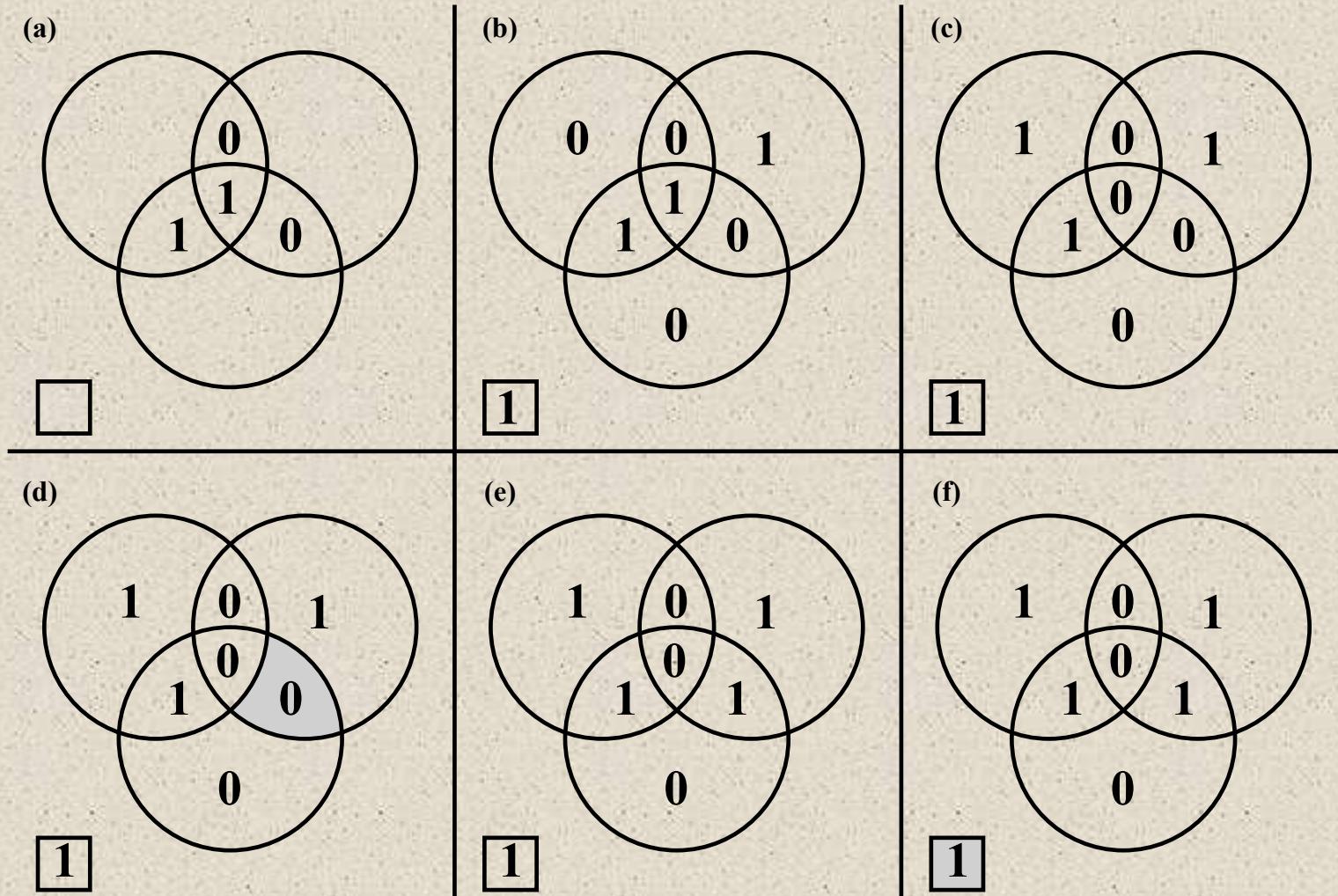


Figure 5.11 Hamming SEC-DED Code

Advanced DRAM Organization

- One of the most critical system bottlenecks when using high-performance processors is the interface to main internal memory
 - The traditional DRAM chip is constrained both by its internal architecture and by its interface to the processor's memory bus
 - A number of enhancements to the basic DRAM architecture have been explored
- +
- The schemes that currently dominate the market are SDRAM and DDR-DRAM

SDRAM

DDR-DRAM

RDRAM

Synchronous DRAM (SDRAM)

One of the most widely used forms of DRAM

Exchanges data with the processor synchronized to an external clock signal and running at the full speed of the processor/memory bus without imposing wait states

With synchronous access the DRAM moves data in and out under control of the system clock

- The processor or other master issues the instruction and address information which is latched by the DRAM
- The DRAM then responds after a set number of clock cycles
- Meanwhile the master can safely do other tasks while the SDRAM is processing

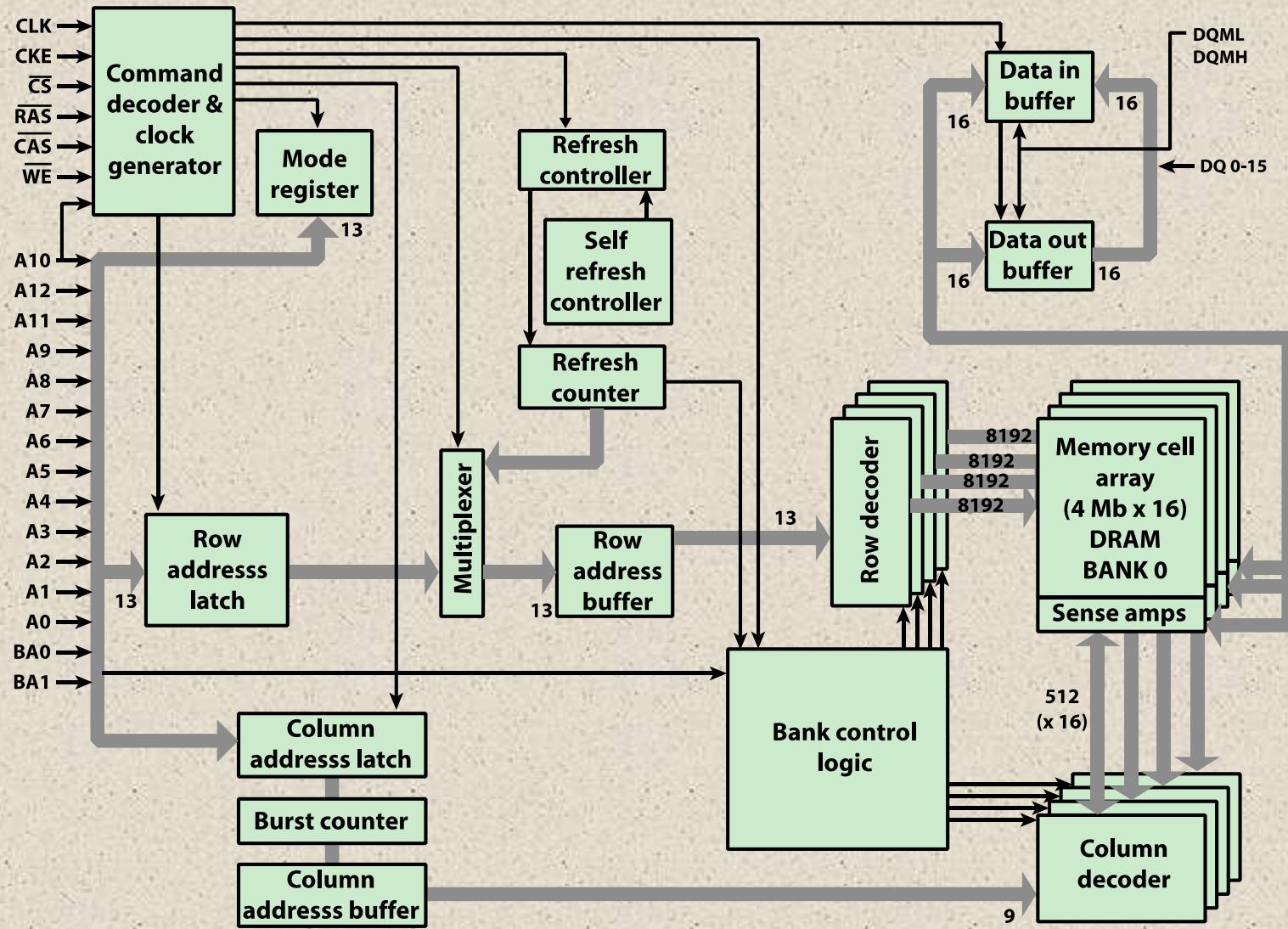


Figure 5.12 256-Mb Synchronous Dynamic RAM (SDRAM)

Table 5.3
SDRAM
Pin
Assignments

A0 to A12	Address inputs
BA0, BA1	Bank address lines
CLK	Clock input
CKE	Clock enable
\overline{CS}	Chip select
\overline{RAS}	Row address strobe
\overline{CAS}	Column address strobe
\overline{WE}	Write enable
DQ0 to DQ15	Data input/output
DQM	Data mask

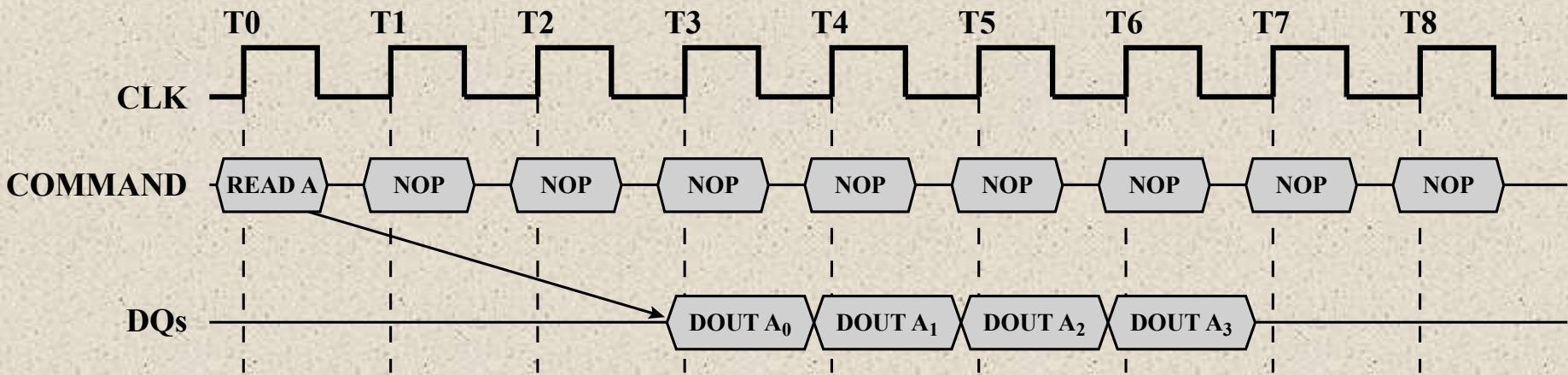
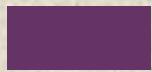


Figure 5.13 SDRAM Read Timing (Burst Length = 4, $\overline{\text{CAS}}$ latency = 2)

Double Data Rate SDRAM (DDR SDRAM)

- Developed by the JEDEC Solid State Technology Association (Electronic Industries Alliance's semiconductor-engineering-standardization body)
- Numerous companies make DDR chips, which are widely used in desktop computers and servers
- DDR achieves higher data rates in three ways:
 - First, the data transfer is synchronized to both the rising and falling edge of the clock, rather than just the rising edge
 - Second, DDR uses higher clock rate on the bus to increase the transfer rate
 - Third, a buffering scheme is used



	DDR1	DDR2	DDR3	DDR4
Prefetch buffer (bits)	2	4	8	8
Voltage level (V)	2.5	1.8	1.5	1.2
Front side bus data rates (Mbps)	200—400	400—1066	800—2133	2133—4266

Table 5.4
DDR Characteristics

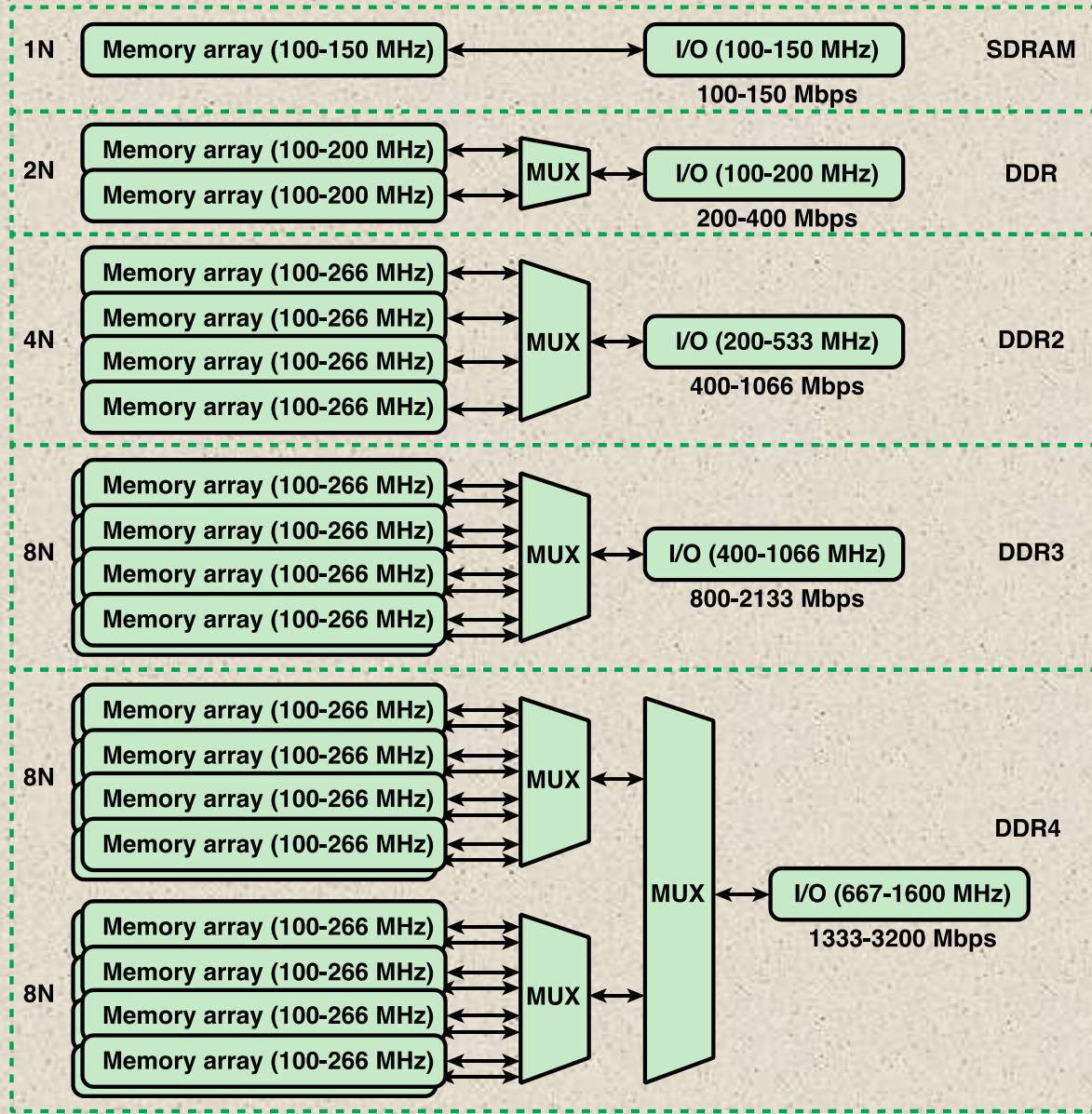
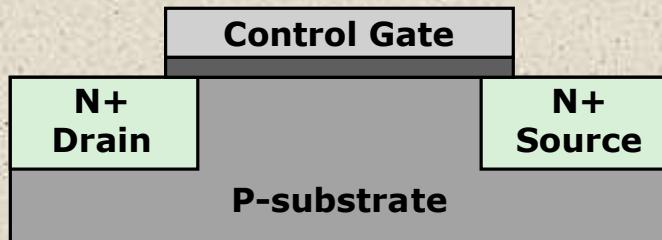


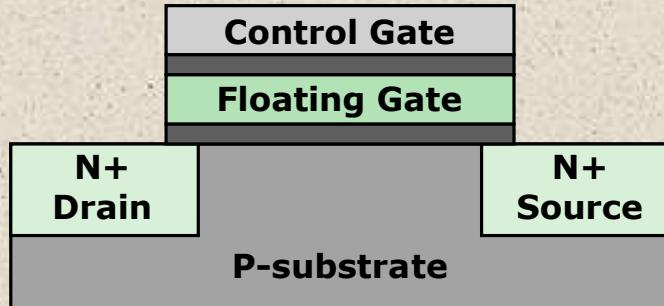
Figure 5.14 DDR Generations

Flash Memory

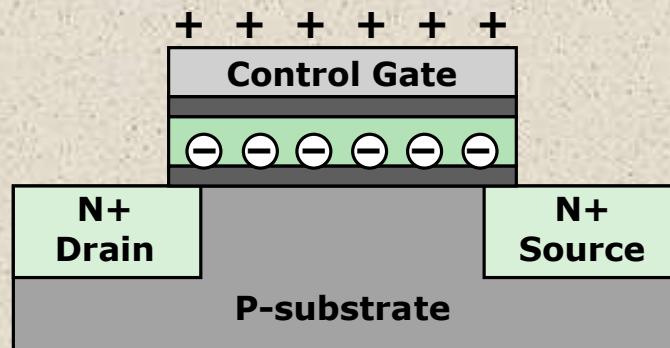
- Used both for internal memory and external memory applications
- First introduced in the mid-1980's
- Is intermediate between EPROM and EEPROM in both cost and functionality
- Uses an electrical erasing technology like EEPROM
- It is possible to erase just blocks of memory rather than an entire chip
- Gets its name because the microchip is organized so that a section of memory cells are erased in a single action
- Does not provide byte-level erasure
- Uses only one transistor per bit so it achieves the high density of EPROM



(a) Transistor structure

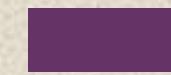


(b) Flash memory cell in one state

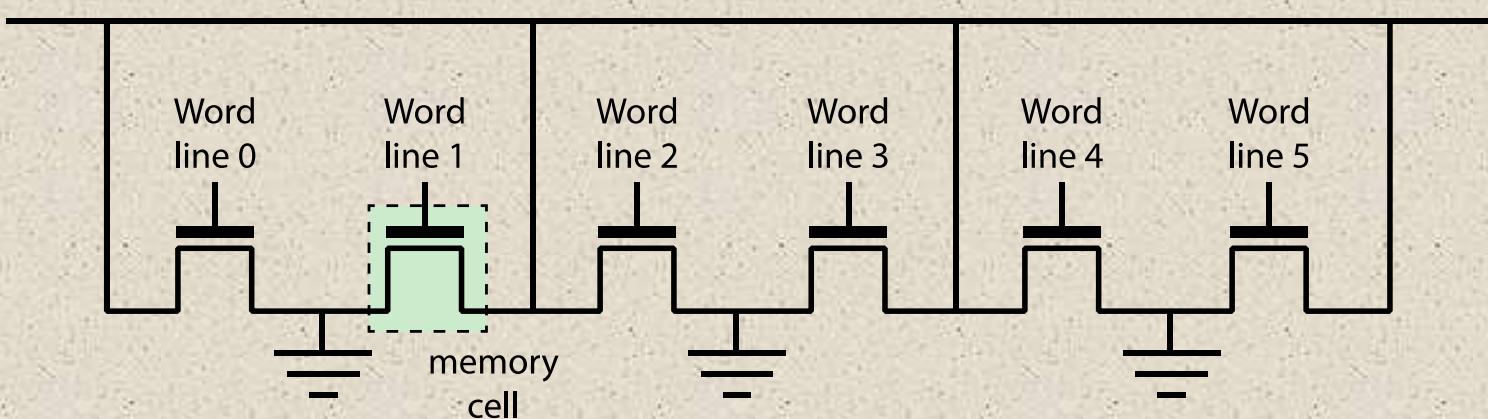


(c) Flash memory cell in zero state

Figure 5.15 Flash Memory Operation

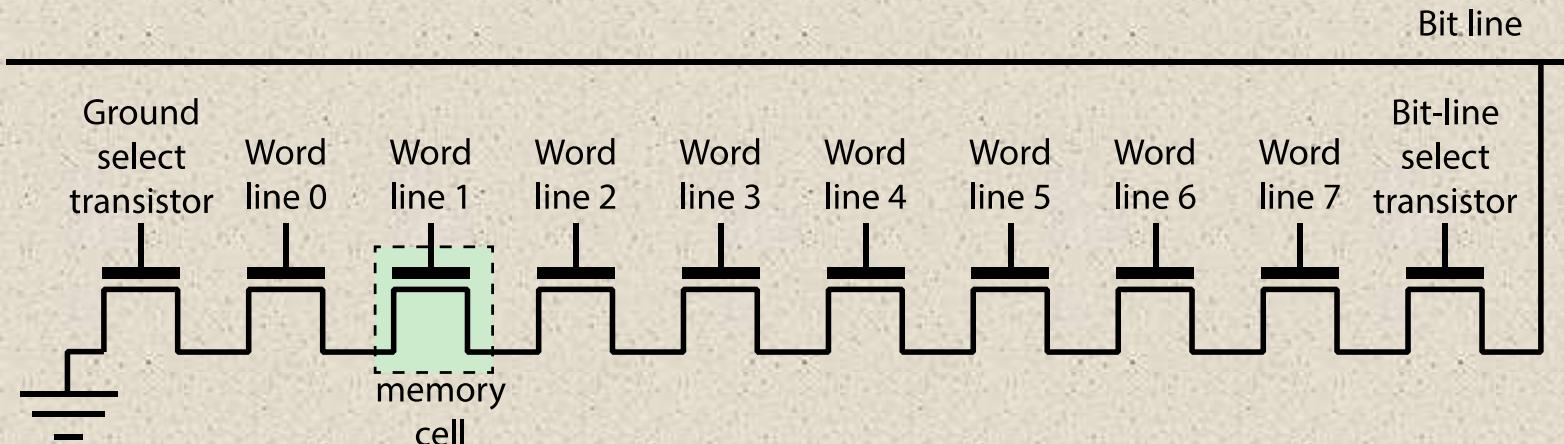


Bit line



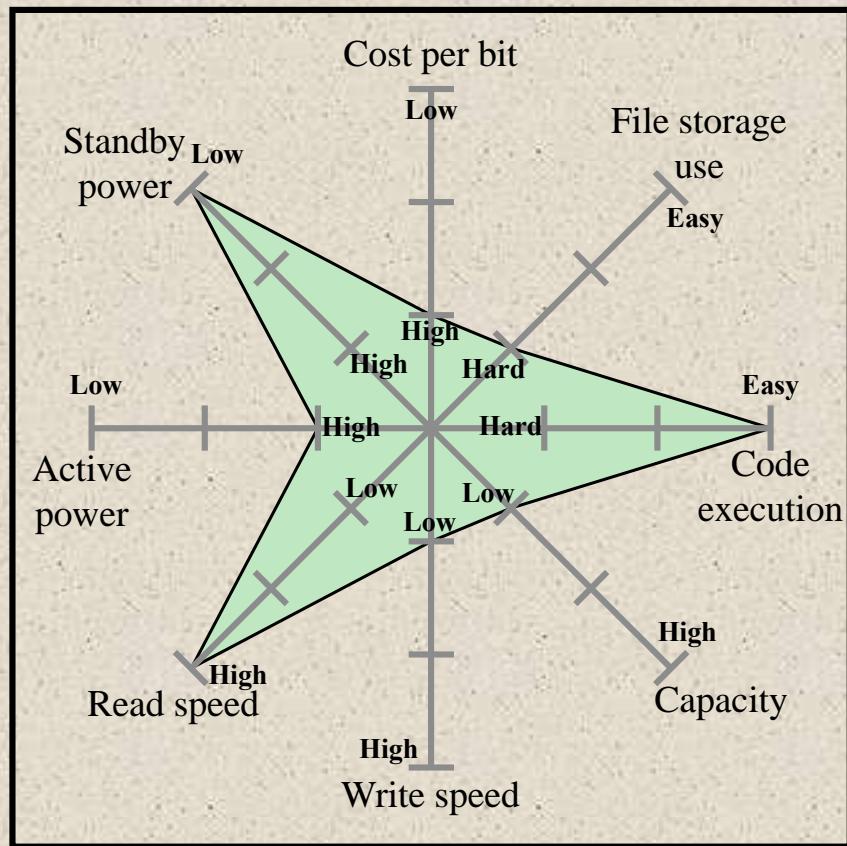
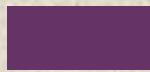
(a) NOR flash structure

Bit line

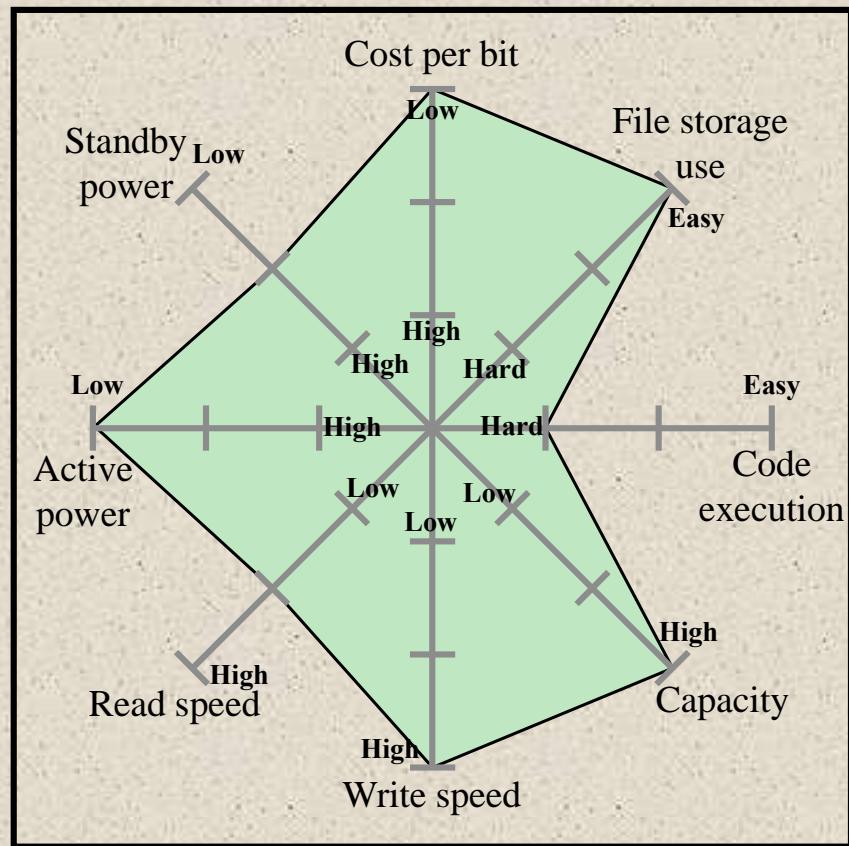


(b) NAND flash structure

Figure 5.16 Flash Memory Structures



(a) NOR



(b) NAND

Figure 5.17 Kiviat Graphs for Flash Memory

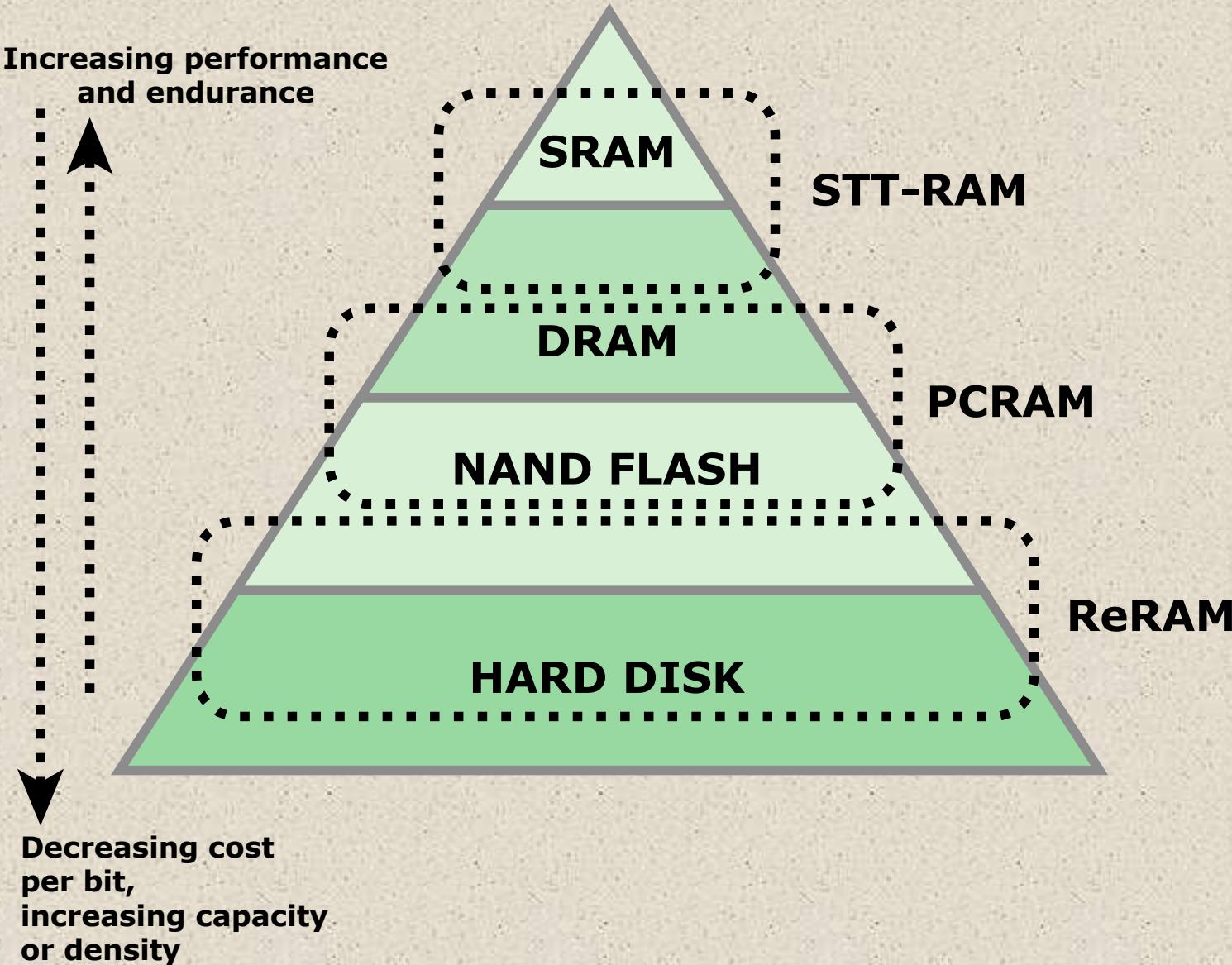
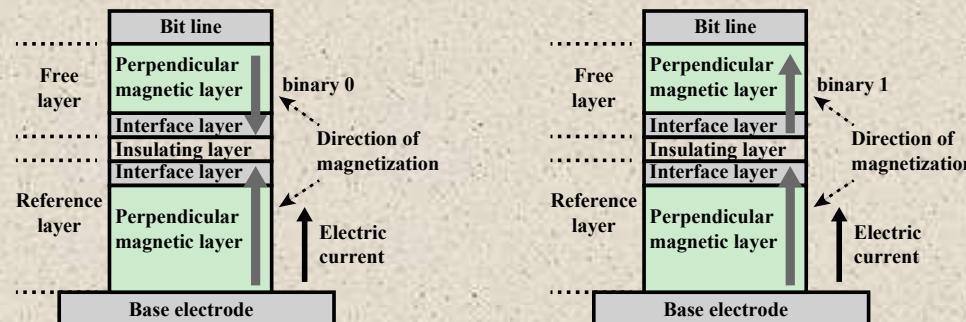
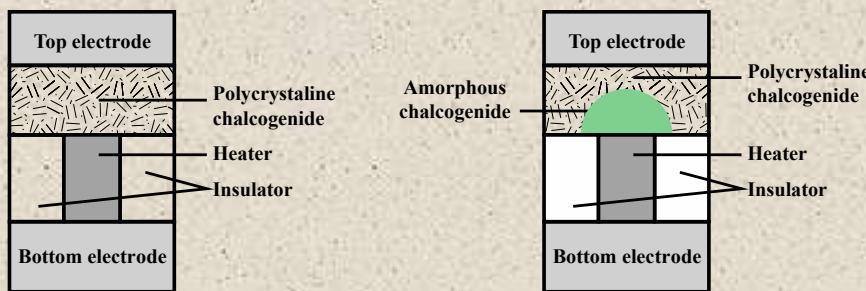


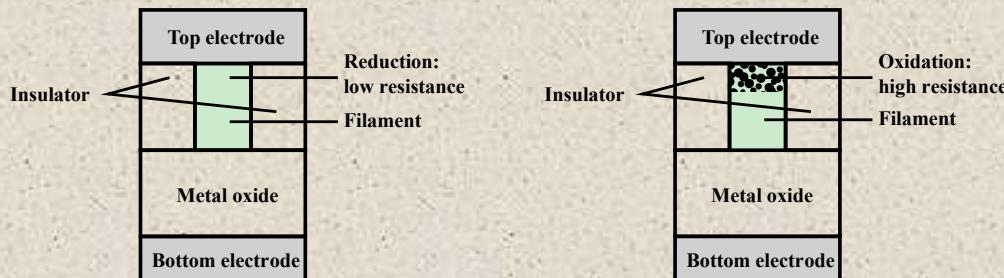
Figure 5.18 Nonvolatile RAM within the Memory Hierarchy



(a) STT-RAM



(b) PCRAM



(c) ReRAM

Figure 5.19 Nonvolatile RAM Technologies

+

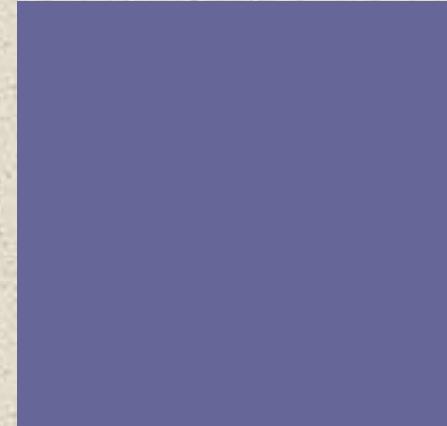
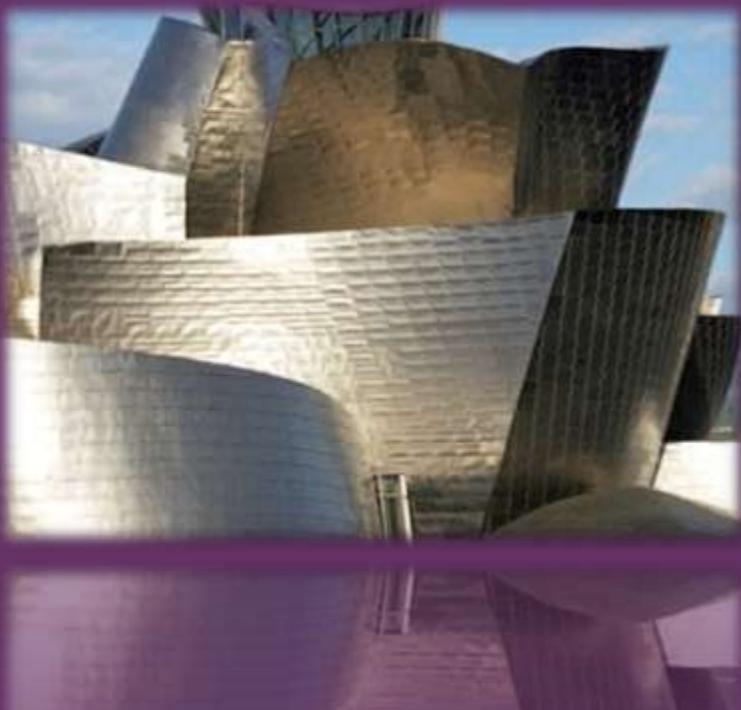
Summary

Chapter 5

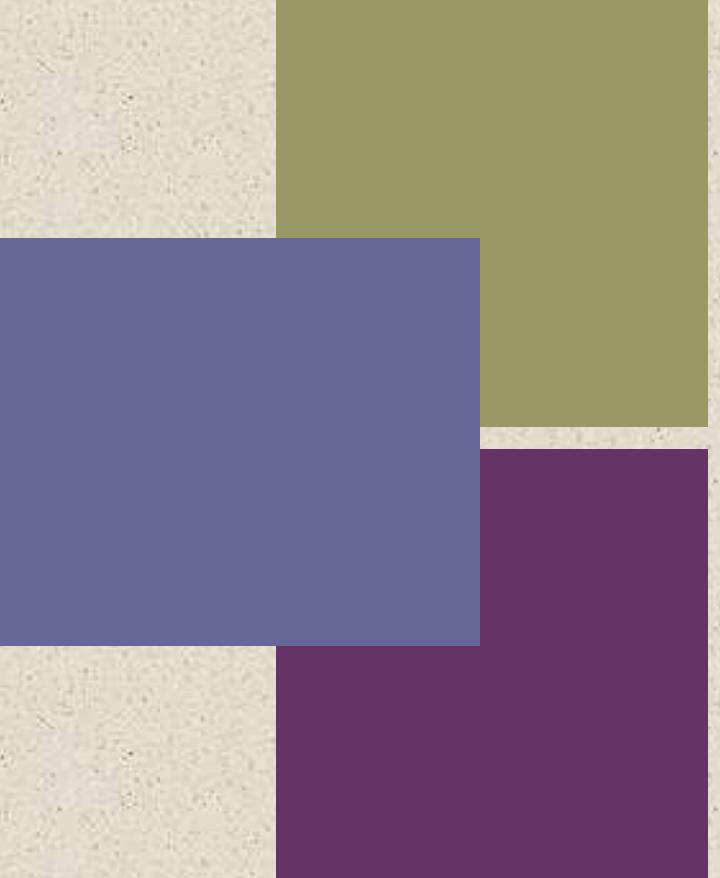
Internal Memory

- Semiconductor main memory
 - Organization
 - DRAM and SRAM
 - Types of ROM
 - Chip logic
 - Chip packaging
 - Module organization
 - Interleaved memory
- Error correction
- DDR DRAM
 - Synchronous DRAM
 - DDR SDRAM
- Flash memory
 - Operation
 - NOR and NAND flash memory
- Newer nonvolatile solid-state memory technologies

+



William Stallings
Computer Organization
and Architecture
10th Edition



+ Chapter 6

External Memory

Magnetic Disk

- A disk is a circular *platter* constructed of nonmagnetic material, called the *substrate*, coated with a magnetizable material
 - Traditionally the substrate has been an aluminium or aluminium alloy material
 - Recently glass substrates have been introduced
- Benefits of the glass substrate:
 - Improvement in the uniformity of the magnetic film surface to increase disk reliability
 - A significant reduction in overall surface defects to help reduce read-write errors
 - Ability to support lower fly heights
 - Better stiffness to reduce disk dynamics
 - Greater ability to withstand shock and damage



Magnetic Read and Write Mechanisms

Data are recorded on and later retrieved from the disk via a conducting coil named the *head*

- In many systems there are two heads, a read head and a write head
- During a read or write operation the head is stationary while the platter rotates beneath it

Electric pulses are sent to the write head and the resulting magnetic patterns are recorded on the surface below, with different patterns for positive and negative currents

An electric current in the wire induces a magnetic field across the gap, which in turn magnetizes a small area of the recording medium

The write mechanism exploits the fact that electricity flowing through a coil produces a magnetic field

The write head itself is made of easily magnetizable material and is in the shape of a rectangular doughnut with a gap along one side and a few turns of conducting wire along the opposite side

Reversing the direction of the current reverses the direction of the magnetization on the recording medium

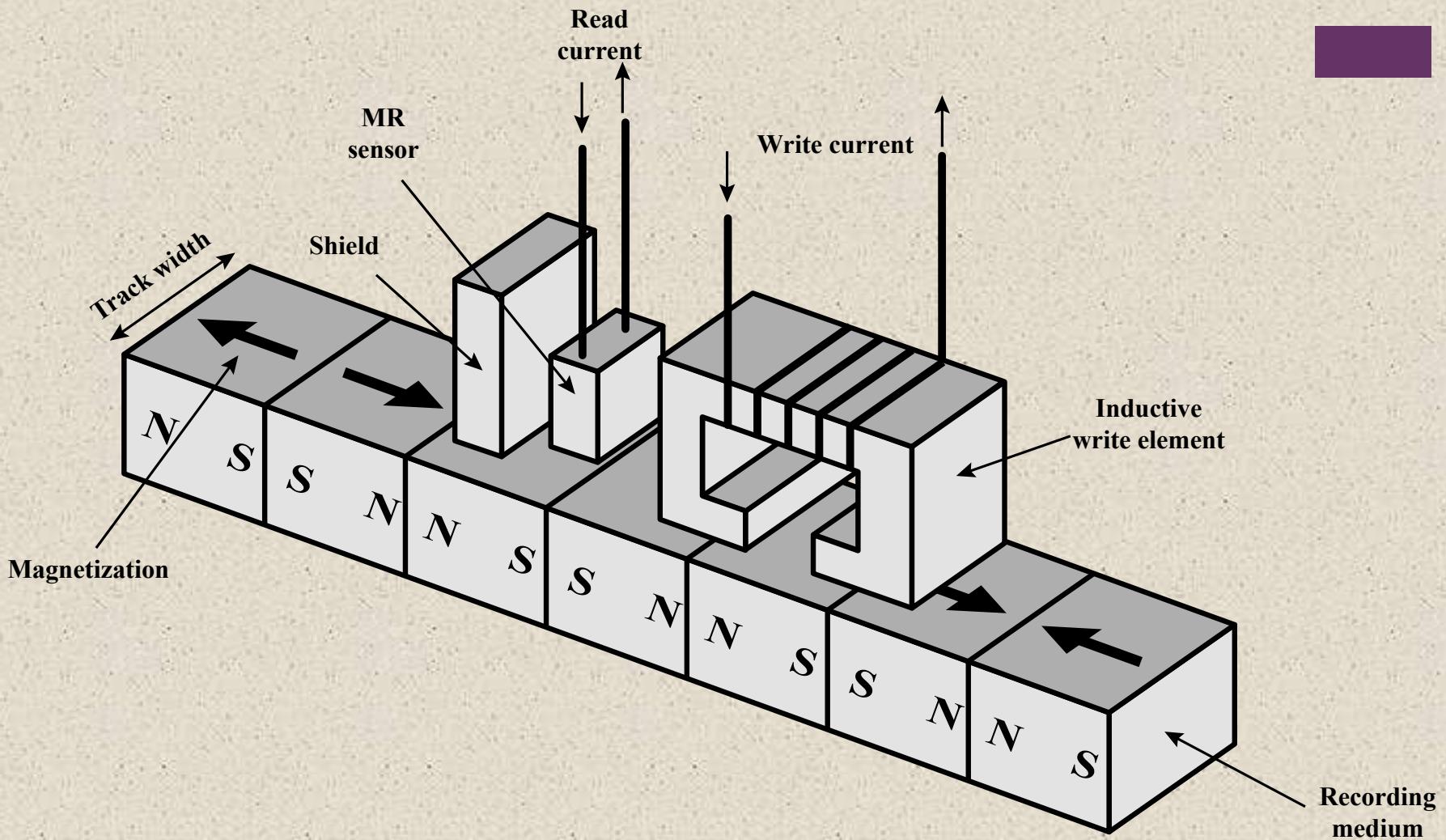


Figure 6.1 Inductive Write/Magnetoresistive Read Head

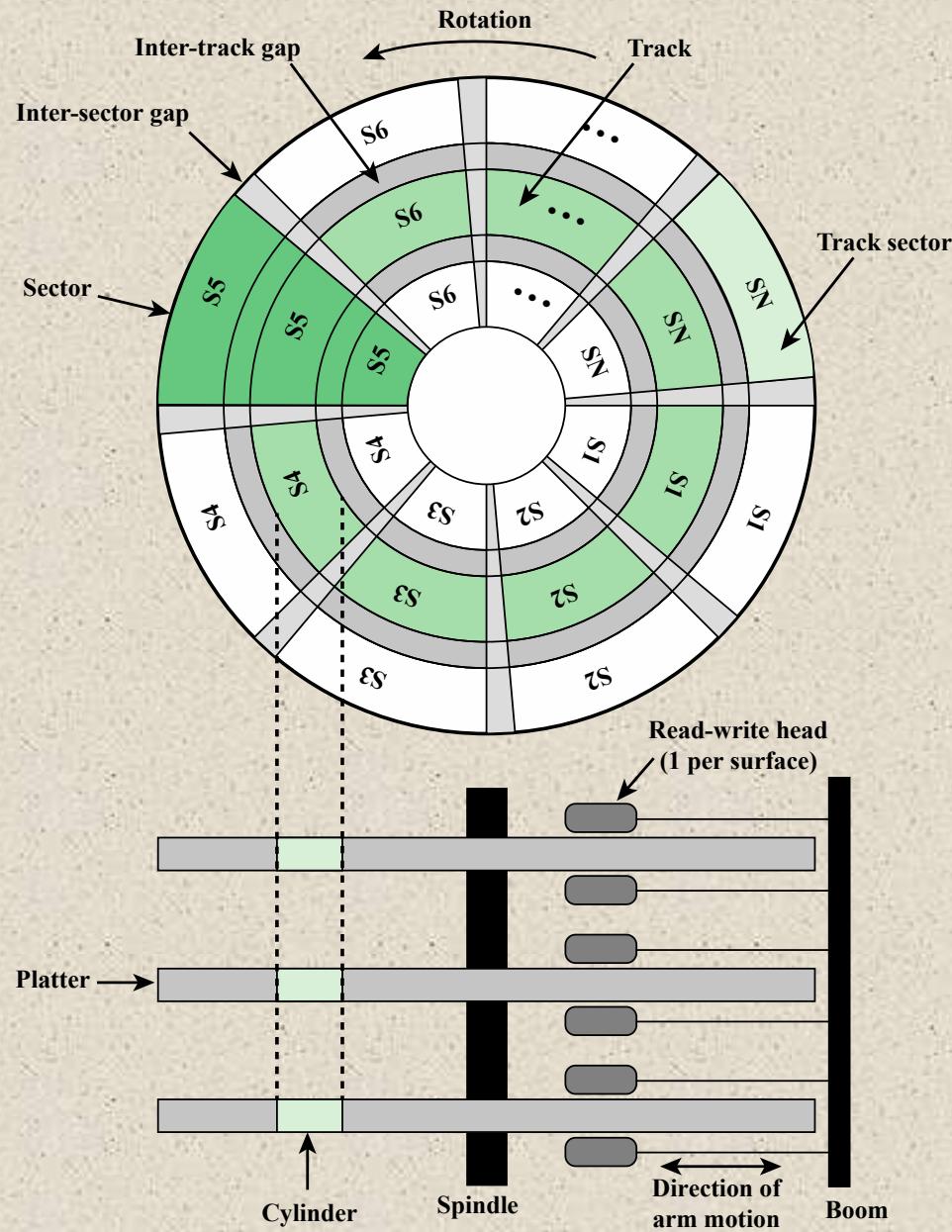
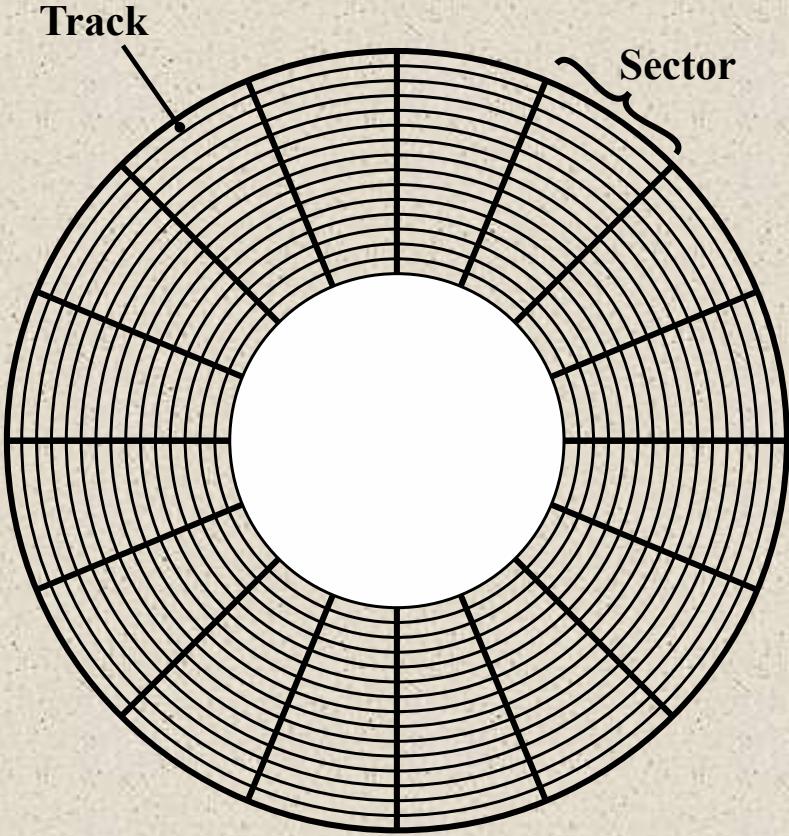
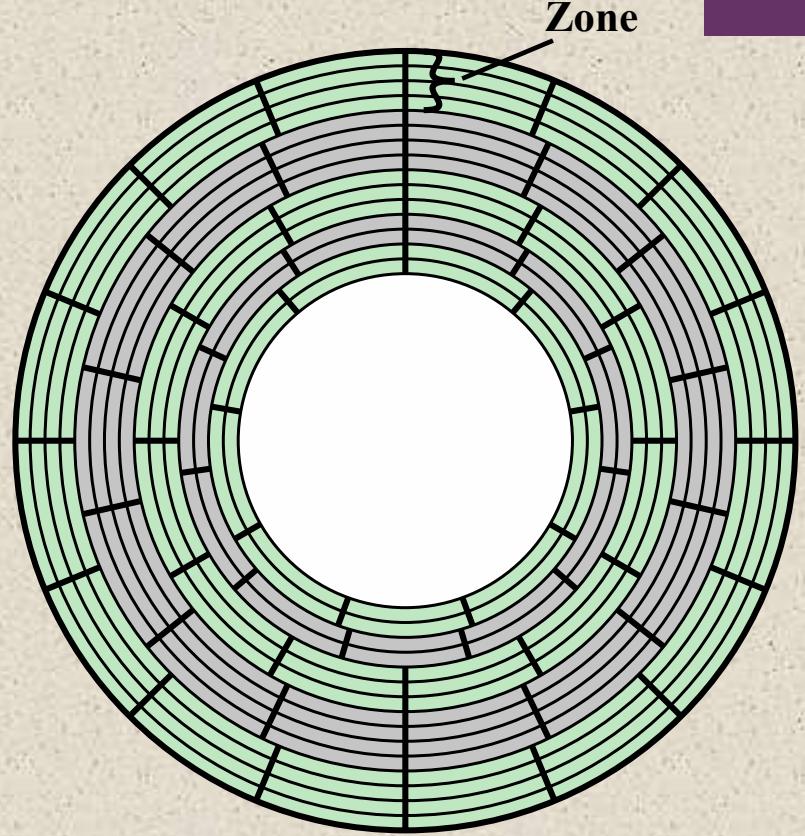


Figure 6.2 Disk Data Layout



(a) Constant angular velocity



(b) Multiple zone recording

Figure 6.3 Comparison of Disk Layout Methods

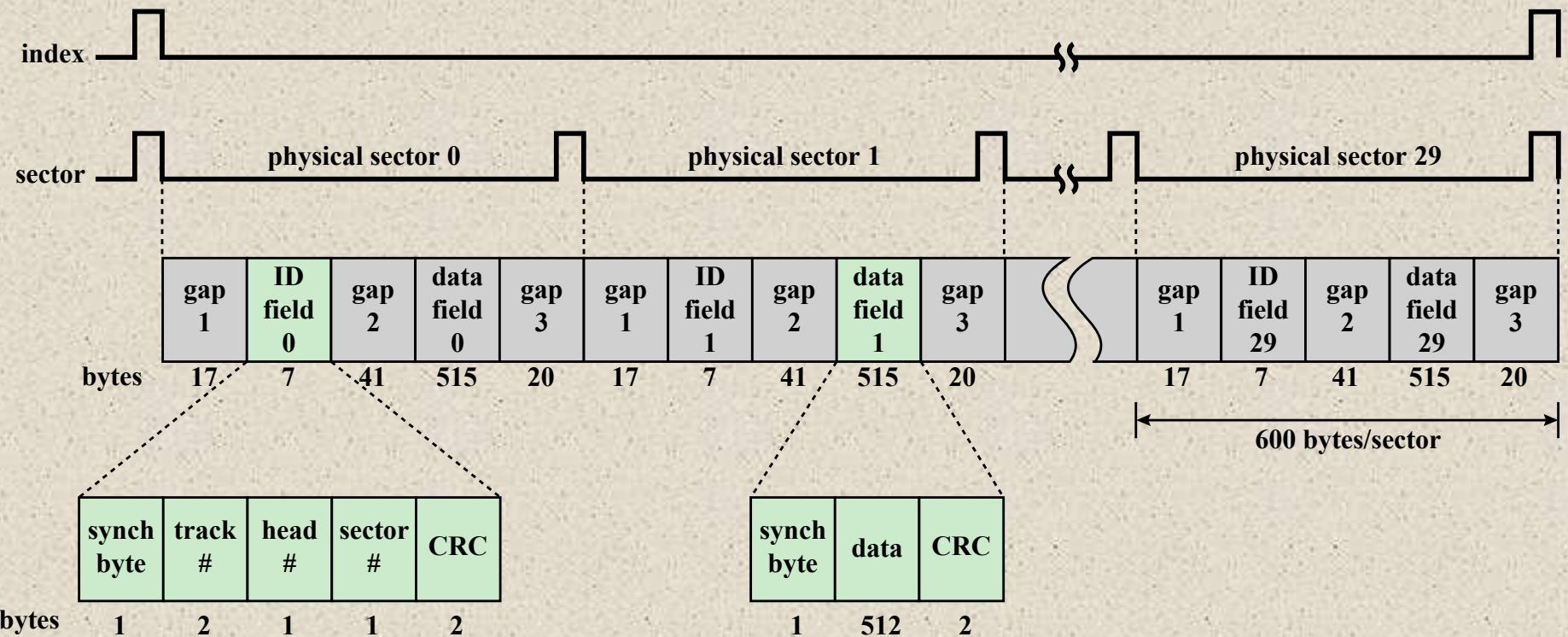


Figure 6.4 Winchester Disk Format (Seagate ST506)

Head Motion

- Fixed head (one per track)
- Movable head (one per surface)

Platters

- Single platter
- Multiple platter

Disk Portability

- Nonremovable disk
- Removable disk

Head Mechanism

- Contact (floppy)
- Fixed gap
- Aerodynamic gap (Winchester)

Sides

- Single sided
- Double sided

Table 6.1
Physical Characteristics of Disk Systems

Characteristics

- Fixed-head disk
 - One read-write head per track
 - Heads are mounted on a fixed ridged arm that extends across all tracks
- Movable-head disk
 - One read-write head
 - Head is mounted on an arm
 - The arm can be extended or retracted
- Non-removable disk
 - Permanently mounted in the disk drive
 - The hard disk in a personal computer is a non-removable disk
- Removable disk
 - Can be removed and replaced with another disk
 - Advantages:
 - Unlimited amounts of data are available with a limited number of disk systems
 - A disk may be moved from one computer system to another
 - Floppy disks and ZIP cartridge disks are examples of removable disks
- Double sided disk
 - Magnetizable coating is applied to both sides of the platter



The head mechanism provides a classification of disks into three types

- The head must generate or sense an electromagnetic field of sufficient magnitude to write and read properly
- The narrower the head, the closer it must be to the platter surface to function
 - A narrower head means narrower tracks and therefore greater data density
- The closer the head is to the disk the greater the risk of error from impurities or imperfections

Disk Classification

Winchester Heads

- Used in sealed drive assemblies that are almost free of contaminants
- Designed to operate closer to the disk's surface than conventional rigid disk heads, thus allowing greater data density
- Is actually an aerodynamic foil that rests lightly on the platter's surface when the disk is motionless
 - The air pressure generated by a spinning disk is enough to make the foil rise above the surface

Table 6.2
Typical Hard Disk Drive Parameters

Characteristics	Seagate Enterprise	Seagate Barracuda XT	Seagate Cheetah NS	Seagate Laptop HDD
Application	Enterprise	Desktop	Network attached storage, application servers	Laptop
Capacity	6 TB	3 TB	600 GB	2 TB
Average seek time	4.16 ms	N/A	3.9 ms read 4.2 ms write	13 ms
Spindle speed	7200 rpm	7200 rpm	10,075 rpm	5400 rpm
Average latency	4.16 ms	4.16 ms	2.98	5.6 ms
Maximum sustained transfer rate	216 MB/s	149 MB/s	97 MB/s	300 MB/s
Bytes per sector	512/4096	512	512	4096
Tracks per cylinder (number of platter surfaces)	8	10	8	4
Cache	128 MB	64 MB	16 MB	8 MB

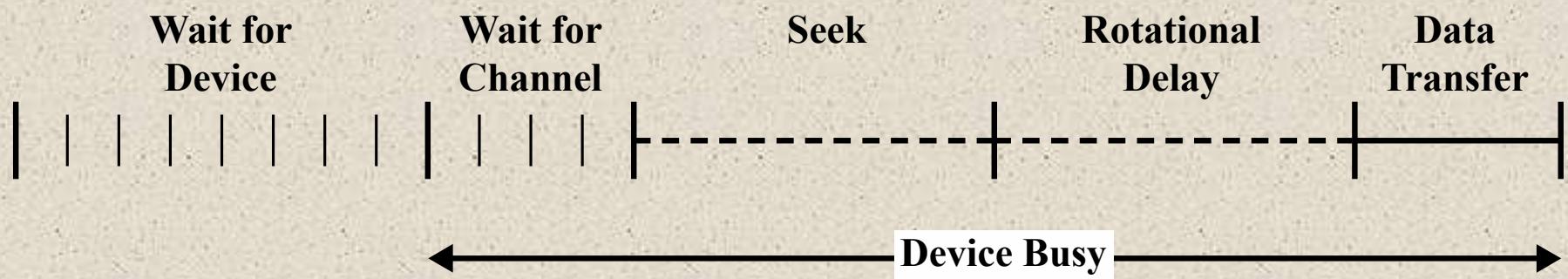
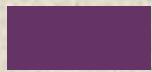
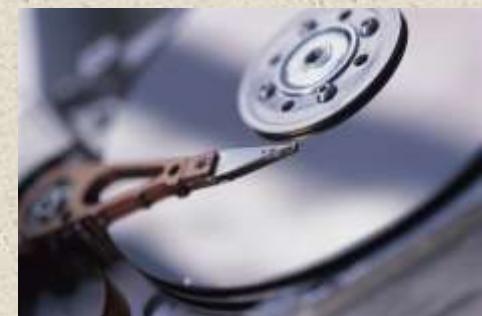


Figure 6.5 Timing of a Disk I/O Transfer

+ Disk Performance Parameters

- When the disk drive is operating the disk is rotating at constant speed
- To read or write the head must be positioned at the desired track and at the beginning of the desired sector on the track
 - Track selection involves moving the head in a movable-head system or electronically selecting one head on a fixed-head-system
 - Once the track is selected, the disk controller waits until the appropriate sector rotates to line up with the head
- Seek time
 - On a movable-head system, the time it takes to position the head at the track
- Rotational delay (*rotational latency*)
 - The time it takes for the beginning of the sector to reach the head
- Access time
 - The sum of the seek time and the rotational delay
 - The time it takes to get into position to read or write
- Transfer time
 - Once the head is in position, the read or write operation is then performed as the sector moves under the head
 - This is the data transfer portion of the operation





RAID

Redundant Array of
Independent Disks

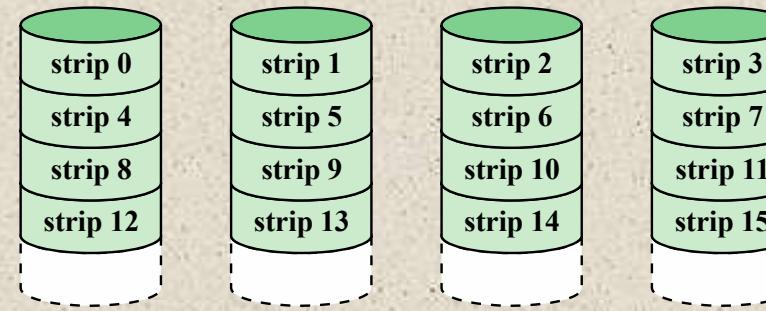
- Consists of 7 levels
- Levels do not imply a hierarchical relationship but designate different design architectures that share three common characteristics:
 - 1) Set of physical disk drives viewed by the operating system as a single logical drive
 - 2) Data are distributed across the physical drives of an array in a scheme known as striping
 - 3) Redundant disk capacity is used to store parity information, which guarantees data recoverability in case of a disk failure

Table 6.3

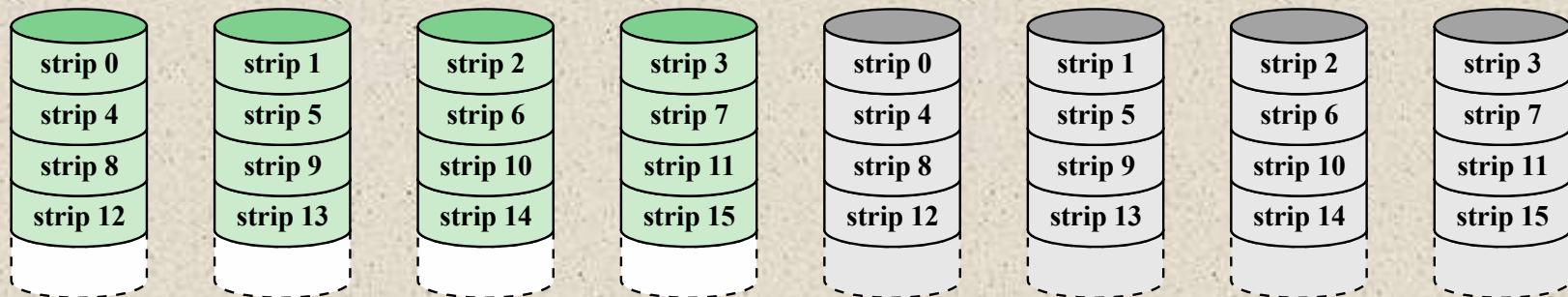
RAID Levels

Category	Level	Description	Disks Required	Data Availability	Large I/O Data Transfer Capacity	Small I/O Request Rate
Striping	0	Nonredundant	N	Lower than single disk	Very high	Very high for both read and write
Mirroring	1	Mirrored	$2N$	Higher than RAID 2, 3, 4, or 5; lower than RAID 6	Higher than single disk for read; similar to single disk for write	Up to twice that of a single disk for read; similar to single disk for write
Parallel access	2	Redundant via Hamming code	$N + m$	Much higher than single disk; comparable to RAID 3, 4, or 5	Highest of all listed alternatives	Approximately twice that of a single disk
	3	Bit-interleaved parity	$N + 1$	Much higher than single disk; comparable to RAID 2, 4, or 5	Highest of all listed alternatives	Approximately twice that of a single disk
Independent access	4	Block-interleaved parity	$N + 1$	Much higher than single disk; comparable to RAID 2, 3, or 5	Similar to RAID 0 for read; significantly lower than single disk for write	Similar to RAID 0 for read; significantly lower than single disk for write
	5	Block-interleaved distributed parity	$N + 1$	Much higher than single disk; comparable to RAID 2, 3, or 4	Similar to RAID 0 for read; lower than single disk for write	Similar to RAID 0 for read; generally lower than single disk for write
	6	Block-interleaved dual distributed parity	$N + 2$	Highest of all listed alternatives	Similar to RAID 0 for read; lower than RAID 5 for write	Similar to RAID 0 for read; significantly lower than RAID 5 for write

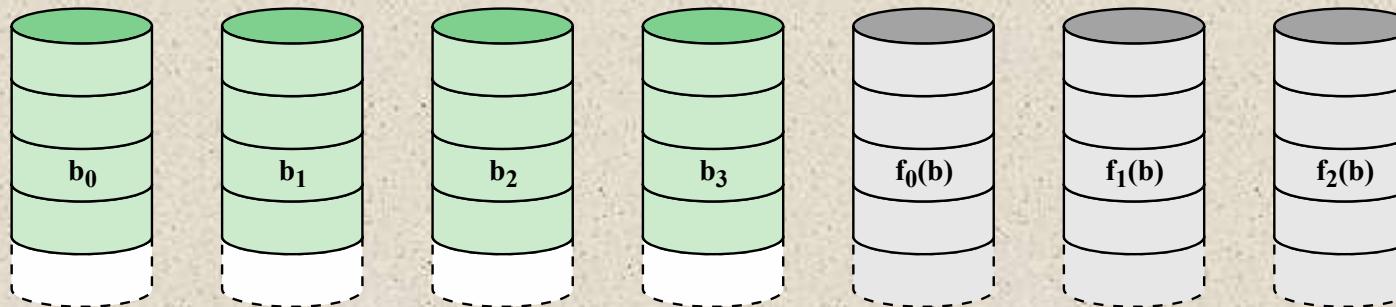
N = number of data disks; m proportional to $\log N$



(a) RAID 0 (non-redundant)

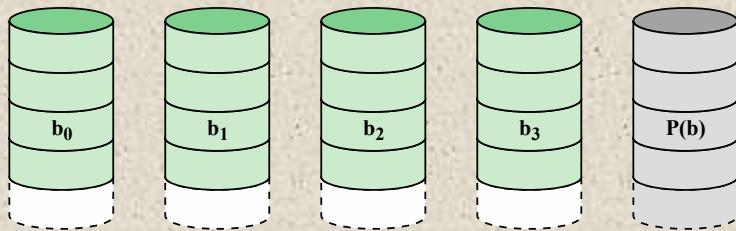


(b) RAID 1 (mirrored)

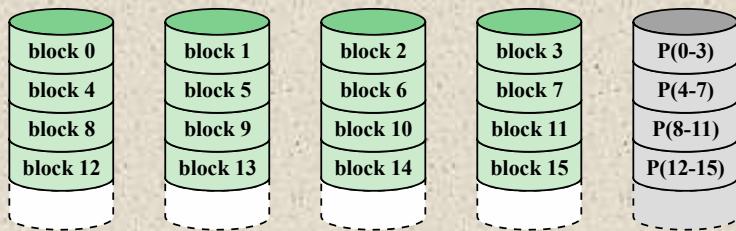


(c) RAID 2 (redundancy through Hamming code)

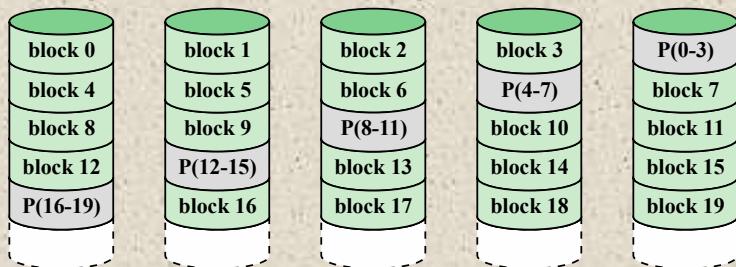
Figure 6.6 RAID Levels (page 1 of 2)



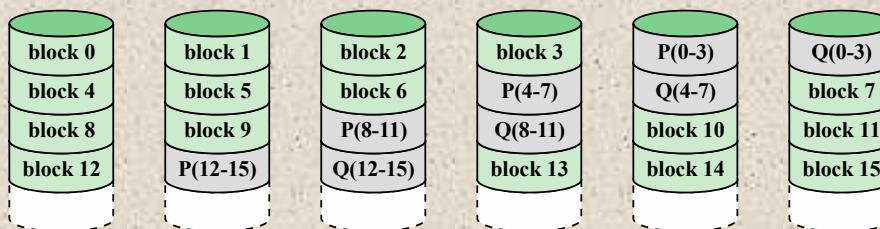
(d) RAID 3 (bit-interleaved parity)



(e) RAID 4 (block-level parity)



(f) RAID 5 (block-level distributed parity)



(g) RAID 6 (dual redundancy)

Figure 6.6 RAID Levels (page 2 of 2)

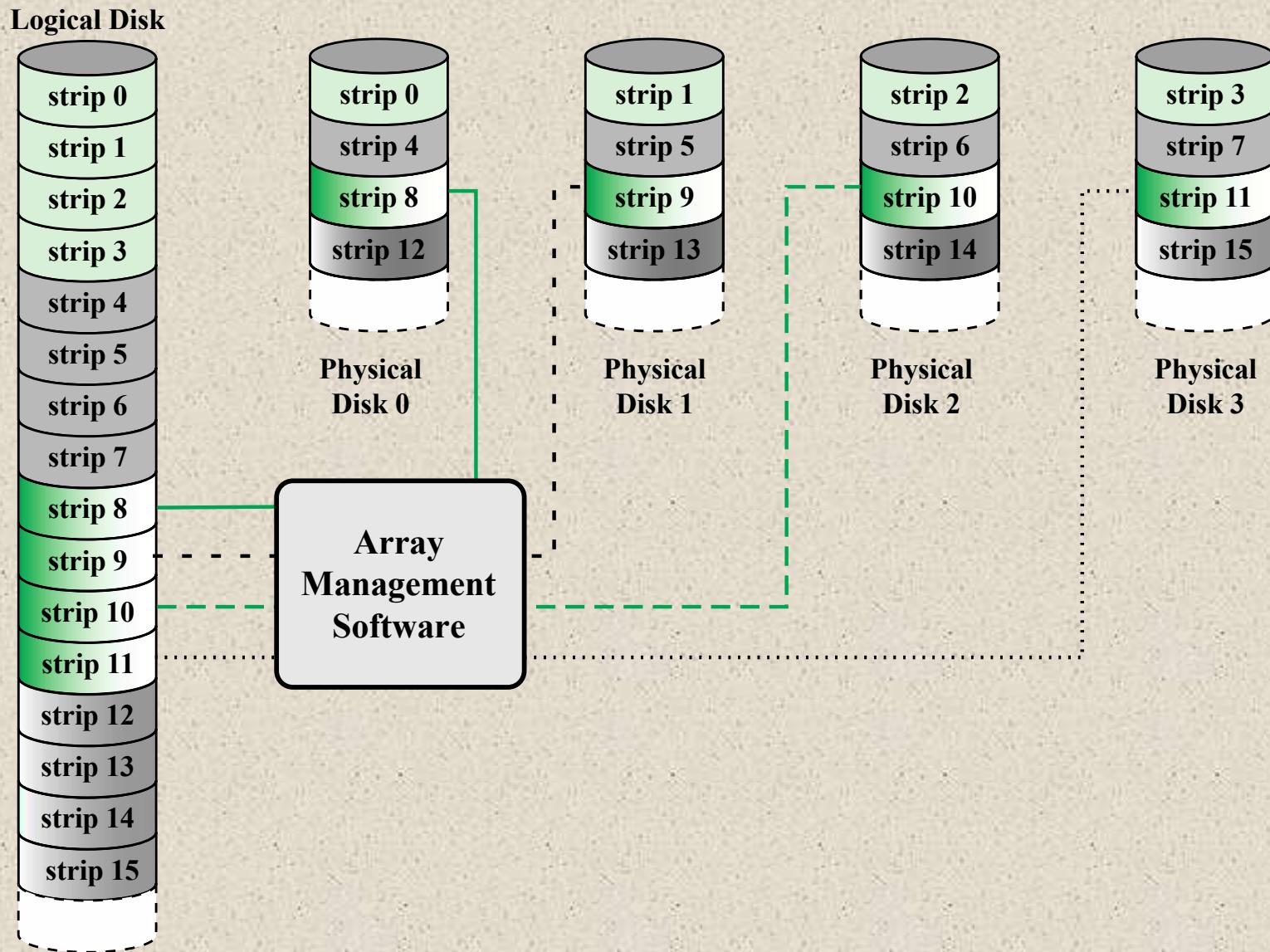


Figure 6.7 Data Mapping for a RAID Level 0 Array

+ RAID Level 0

RAID 0 for High Data Transfer Capacity

- For applications to experience a high transfer rate two requirements must be met:
 1. A high transfer capacity must exist along the entire path between host memory and the individual disk drives
 2. The application must make I/O requests that drive the disk array efficiently

- Addresses the issues of request patterns of the host system and layout of the data
- Impact of redundancy does not interfere with analysis

RAID 0 for High I/O Request Rate

- For an individual I/O request for a small amount of data the I/O time is dominated by the seek time and rotational latency
- A disk array can provide high I/O execution rates by balancing the I/O load across multiple disks
- If the strip size is relatively large multiple waiting I/O requests can be handled in parallel, reducing the queuing time for each request

RAID Level 1

Characteristics

- Differs from RAID levels 2 through 6 in the way in which redundancy is achieved
- Redundancy is achieved by the simple expedient of duplicating all the data
- Data striping is used but each logical strip is mapped to two separate physical disks so that every disk in the array has a mirror disk that contains the same data
- RAID 1 can also be implemented without data striping, although this is less common

Positive Aspects

- A read request can be serviced by either of the two disks that contains the requested data
- There is no “write penalty”
- Recovery from a failure is simple, when a drive fails the data can be accessed from the second drive
- Provides real-time copy of all data
- Can achieve high I/O request rates if the bulk of the requests are reads
- Principal disadvantage is the cost



RAID Level 2

Characteristics

- Makes use of a parallel access technique
- In a parallel access array all member disks participate in the execution of every I/O request
- Spindles of the individual drives are synchronized so that each disk head is in the same position on each disk at any given time
- Data striping is used
 - Strips are very small, often as small as a single byte or word

Performance

- An error-correcting code is calculated across corresponding bits on each data disk and the bits of the code are stored in the corresponding bit positions on multiple parity disks
- Typically a Hamming code is used, which is able to correct single-bit errors and detect double-bit errors
- The number of redundant disks is proportional to the log of the number of data disks
- Would only be an effective choice in an environment in which many disk errors occur

RAID Level 3

Redundancy

- Requires only a single redundant disk, no matter how large the disk array
- Employs parallel access, with data distributed in small strips
- Instead of an error correcting code, a simple parity bit is computed for the set of individual bits in the same position on all of the data disks
- Can achieve very high data transfer rates

Performance

- In the event of a drive failure, the parity drive is accessed and data is reconstructed from the remaining devices
- Once the failed drive is replaced, the missing data can be restored on the new drive and operation resumed
- In the event of a disk failure, all of the data are still available in what is referred to as *reduced mode*
- Return to full operation requires that the failed disk be replaced and the entire contents of the failed disk be regenerated on the new disk
- In a transaction-oriented environment performance suffers

+ RAID Level 4

Characteristics

- Makes use of an independent access technique
 - In an independent access array, each member disk operates independently so that separate I/O requests can be satisfied in parallel
- Data striping is used
 - Strips are relatively large
- To calculate the new parity the array management software must read the old user strip and the old parity strip

Performance

- Involves a write penalty when an I/O write request of small size is performed
- Each time a write occurs the array management software must update not only the user data but also the corresponding parity bits
- Thus each strip write involves two reads and two writes

4

RAID Level 5

Characteristics

- Organized in a similar fashion to RAID 4
- Difference is distribution of the parity strips across all disks
- A typical allocation is a round-robin scheme
- The distribution of parity strips across all drives avoids the potential I/O bottleneck found in RAID 4

RAID Level 6

Characteristics

- Two different parity calculations are carried out and stored in separate blocks on different disks
- Advantage is that it provides extremely high data availability
- Three disks would have to fail within the mean time to repair (MTTR) interval to cause data to be lost
- Incurs a substantial write penalty because each write affects two parity blocks

Table 6.4
RAID
Comparison
(page 1 of 2)

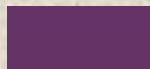
Level	Advantages	Disadvantages	Applications
0	I/O performance is greatly improved by spreading the I/O load across many channels and drives No parity calculation overhead is involved Very simple design Easy to implement	The failure of just one drive will result in all data in an array being lost	Video production and Editing Image editing Pre-press applications Any application requiring high bandwidth
1	100% redundancy of data means no rebuild is necessary in case of a disk failure, just a copy to the replacement disk Under certain circumstances, RAID 1 can sustain multiple simultaneous drive failures Simplest RAID storage subsystem design	Highest disk overhead of all RAID types (100%) - inefficient	Accounting Payroll Financial Any application requiring very high availability
2	Extremely high data transfer rates possible The higher the data transfer rate required, the better the ratio of data disks to ECC disks Relatively simple controller design compared to RAID levels 3,4 & 5	Very high ratio of ECC disks to data disks with smaller word sizes - inefficient Entry level cost very high - requires very high transfer rate requirement to justify	No commercial implementations exist / not commercially viable

Table 6.4
RAID
Comparison
(page 2 of 2)

3	Very high read data transfer rate Very high write data transfer rate Disk failure has an insignificant impact on throughput Low ratio of ECC (parity) disks to data disks means high efficiency	Transaction rate equal to that of a single disk drive at best (if spindles are synchronized) Controller design is fairly complex	Video production and live streaming Image editing Video editing Prepress applications Any application requiring high throughput
4	Very high Read data transaction rate Low ratio of ECC (parity) disks to data disks means high efficiency	Quite complex controller design Worst write transaction rate and Write aggregate transfer rate Difficult and inefficient data rebuild in the event of disk failure	No commercial implementations exist / not commercially viable
5	Highest Read data transaction rate Low ratio of ECC (parity) disks to data disks means high efficiency Good aggregate transfer rate	Most complex controller design Difficult to rebuild in the event of a disk failure (as compared to RAID level 1)	File and application servers Database servers Web, e-mail, and news servers Intranet servers Most versatile RAID level
6	Provides for an extremely high data fault tolerance and can sustain multiple simultaneous drive failures	More complex controller design Controller overhead to compute parity addresses is extremely high	Perfect solution for mission critical applications

SSD Compared to HDD

- SSDs have the following advantages over HDDs:
- High-performance input/output operations per second (IOPS)
- Durability
- Longer lifespan
- Lower power consumption
- Quieter and cooler running capabilities
- Lower access times and latency rates



	NAND Flash Drives	Seagate Laptop Internal HDD
File copy/write speed	200—550 Mbps	50—120 Mbps
Power draw/battery life	Less power draw, averages 2–3 watts, resulting in 30+ minute battery boost	More power draw, averages 6–7 watts and therefore uses more battery
Storage capacity	Typically not larger than 512 GB for notebook size drives; 1 TB max for desktops	Typically around 500 GB and 2 TB maximum for notebook size drives; 4 TB max for desktops
Cost	Approx. \$0.50 per GB for a 1-TB drive	Approx \$0.15 per GB for a 4-TB drive

Table 6.5
Comparison of Solid State Drives and Disk Drives

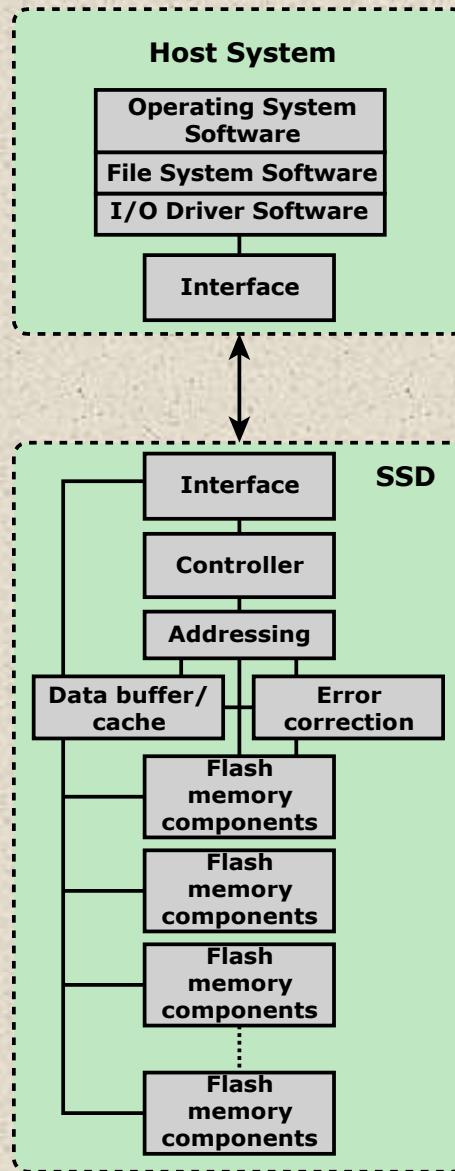


Figure 6.8 Solid State Drive Architecture

+ Practical Issues

There are two practical issues peculiar to SSDs that are not faced by HDDs:

- SSD performance has a tendency to slow down as the device is used
 - The entire block must be read from the flash memory and placed in a RAM buffer
 - Before the block can be written back to flash memory, the entire block of flash memory must be erased
 - The entire block from the buffer is now written back to the flash memory
- Flash memory becomes unusable after a certain number of writes
 - Techniques for prolonging life:
 - Front-ending the flash with a cache to delay and group write operations
 - Using wear-leveling algorithms that evenly distribute writes across block of cells
 - Bad-block management techniques
 - Most flash devices estimate their own remaining lifetimes so systems can anticipate failure and take preemptive action

CD

Compact Disk. A nonerasable disk that stores digitized audio information. The standard system uses 12-cm disks and can record more than 60 minutes of uninterrupted playing time.

CD-ROM

Compact Disk Read-Only Memory. A nonerasable disk used for storing computer data. The standard system uses 12-cm disks and can hold more than 650 Mbytes.

CD-R

CD Recordable. Similar to a CD-ROM. The user can write to the disk only once.

CD-RW

CD Rewritable. Similar to a CD-ROM. The user can erase and rewrite to the disk multiple times.

DVD

Digital Versatile Disk. A technology for producing digitized, compressed representation of video information, as well as large volumes of other digital data. Both 8 and 12 cm diameters are used, with a double-sided capacity of up to 17 Gbytes. The basic DVD is read-only (DVD-ROM).

DVD-R

DVD Recordable. Similar to a DVD-ROM. The user can write to the disk only once. Only one-sided disks can be used.

DVD-RW

DVD Rewritable. Similar to a DVD-ROM. The user can erase and rewrite to the disk multiple times. Only one-sided disks can be used.

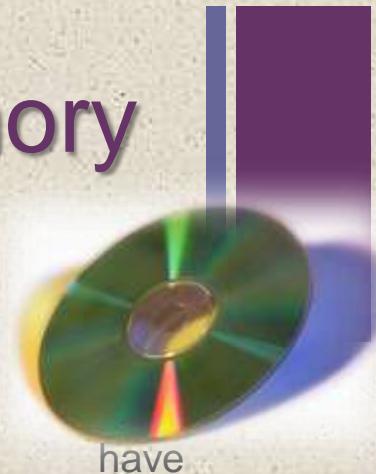
Blu-Ray DVD

High definition video disk. Provides considerably greater data storage density than DVD, using a 405-nm (blue-violet) laser. A single layer on a single side can store 25 Gbytes.

Table 6. 6

Optical Disk Products

Compact Disk Read-Only Memory (CD-ROM)



- Audio CD and the CD-ROM share a similar technology
 - The main difference is that CD-ROM players are more rugged and have error correction devices to ensure that data are properly transferred
- Production:
 - The disk is formed from a resin such as polycarbonate
 - Digitally recorded information is imprinted as a series of microscopic pits on the surface of the polycarbonate
 - This is done with a finely focused, high intensity laser to create a master disk
 - The master is used, in turn, to make a die to stamp out copies onto polycarbonate
 - The pitted surface is then coated with a highly reflective surface, usually aluminum or gold
 - This shiny surface is protected against dust and scratches by a top coat of clear acrylic
 - Finally a label can be silkscreened onto the acrylic

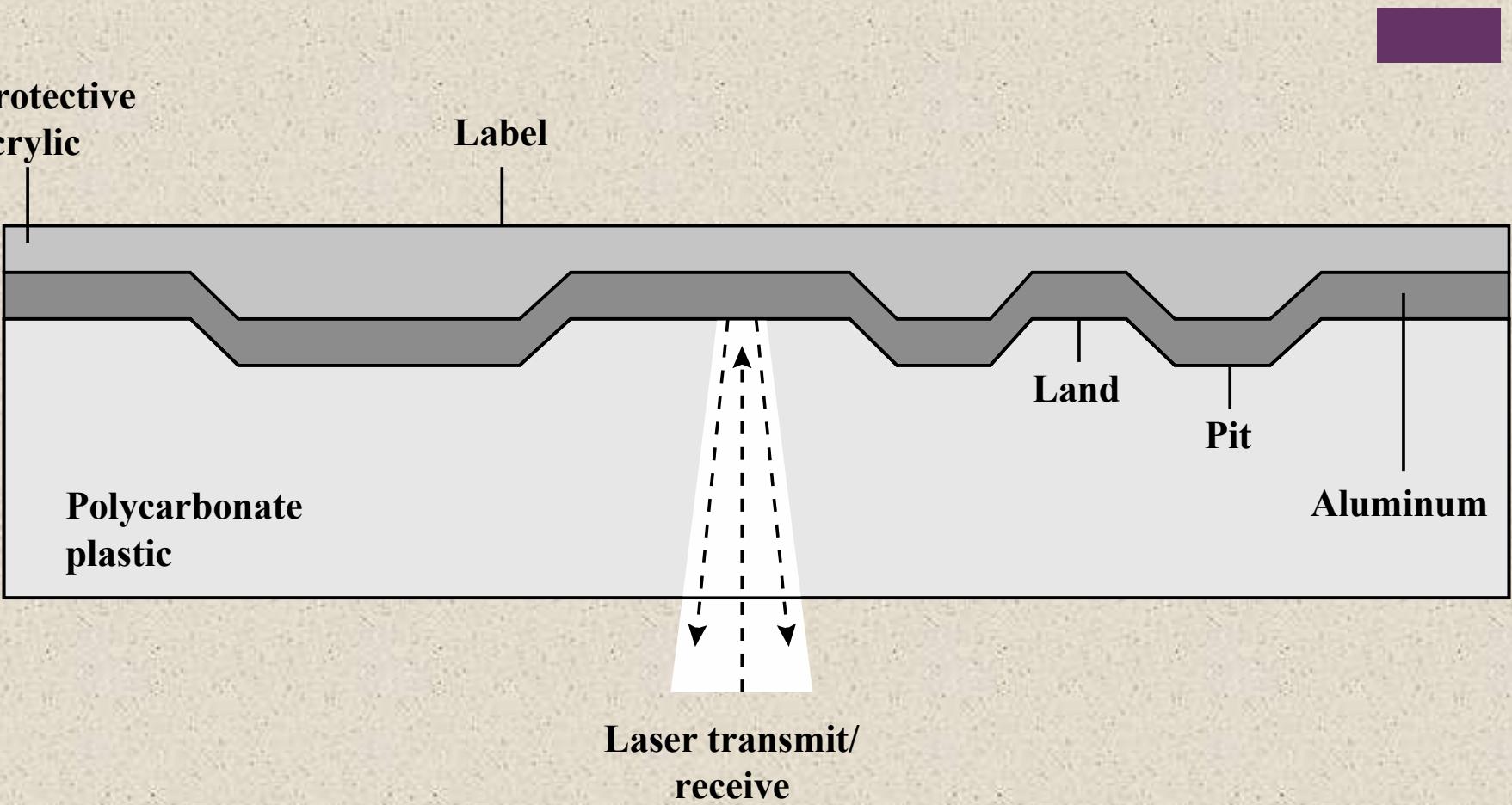


Figure 6.9 CD Operation

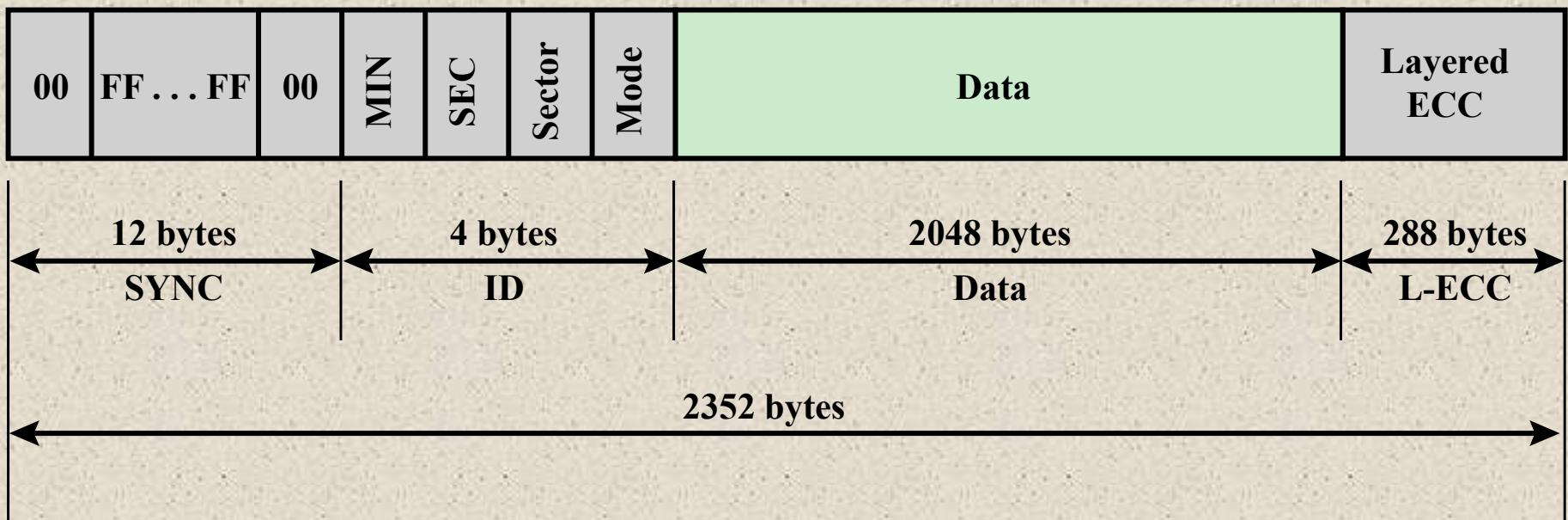
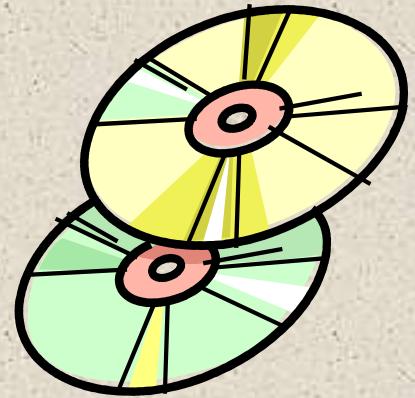


Figure 6.10 CD-ROM Block Format



- CD-ROM is appropriate for the distribution of large amounts of data to a large number of users
- Because the expense of the initial writing process it is not appropriate for individualized applications
- The CD-ROM has two advantages:
 - The optical disk together with the information stored on it can be mass replicated inexpensively
 - The optical disk is removable, allowing the disk itself to be used for archival storage
- The CD-ROM disadvantages:
 - It is read-only and cannot be updated
 - It has an access time much longer than that of a magnetic disk drive

CD-ROM

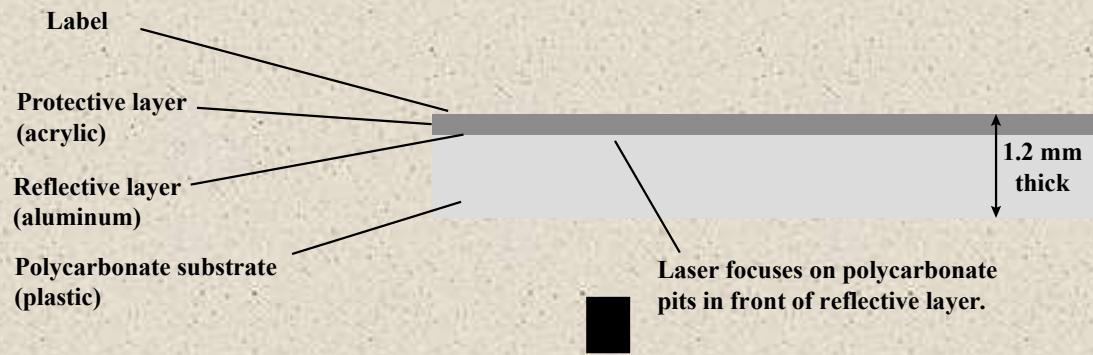


CD Recordable (CD-R)

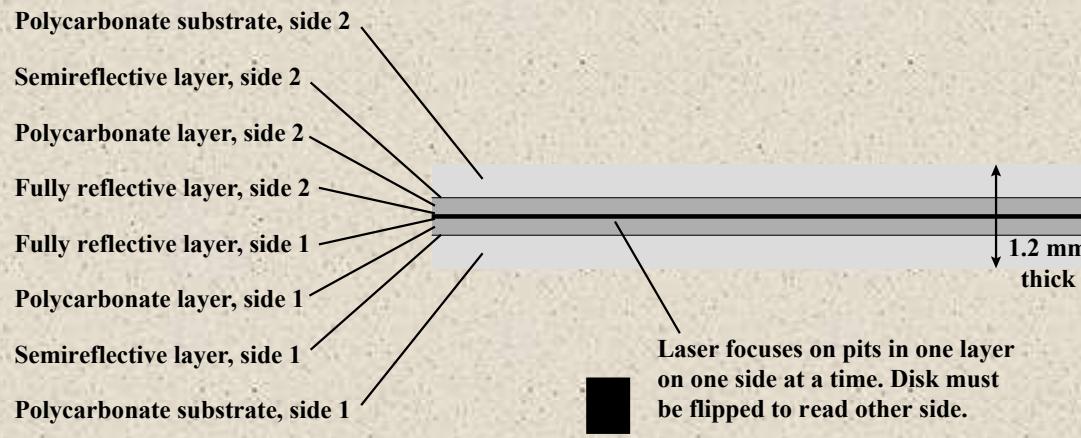
- Write-once read-many
- Accommodates applications in which only one or a small number of copies of a set of data is needed
- Disk is prepared in such a way that it can be subsequently written once with a laser beam of modest-intensity
- Medium includes a dye layer which is used to change reflectivity and is activated by a high-intensity laser
- Provides a permanent record of large volumes of user data

CD Rewritable (CD-RW)

- Can be repeatedly written and overwritten
- Phase change disk uses a material that has two significantly different reflectivities in two different phase states
- Amorphous state
 - Molecules exhibit a random orientation that reflects light poorly
- Crystalline state
 - Has a smooth surface that reflects light well
- A beam of laser light can change the material from one phase to the other
- Disadvantage is that the material eventually and permanently loses its desirable properties
- Advantage is that it can be rewritten



(a) CD-ROM - Capacity 682 MB



(b) DVD-ROM, double-sided, dual-layer - Capacity 17 GB

Figure 6.11 CD-ROM and DVD-ROM

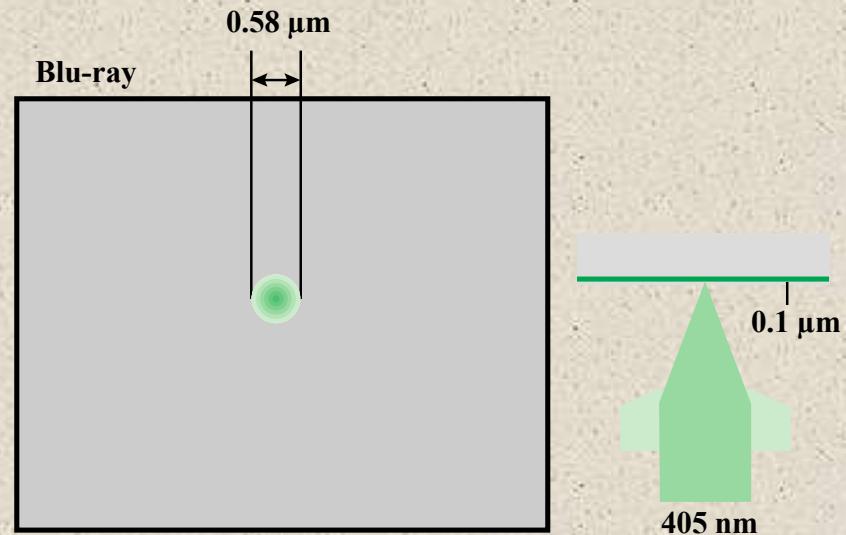
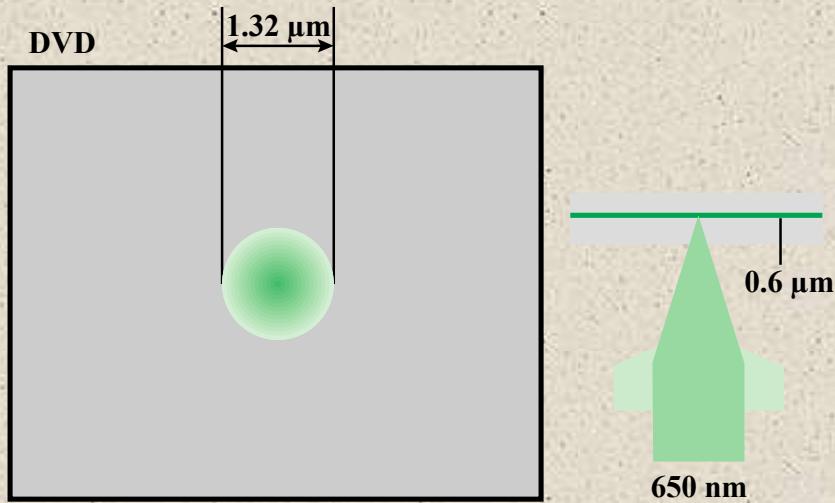
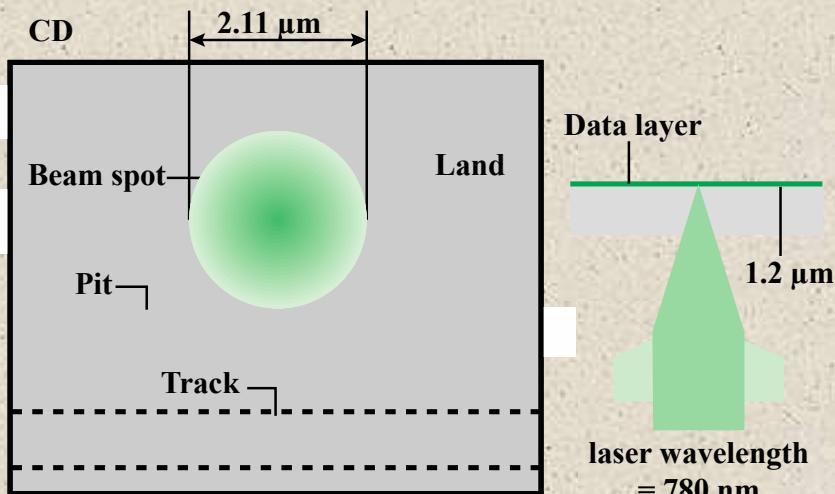
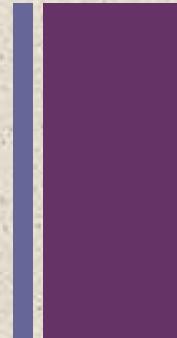


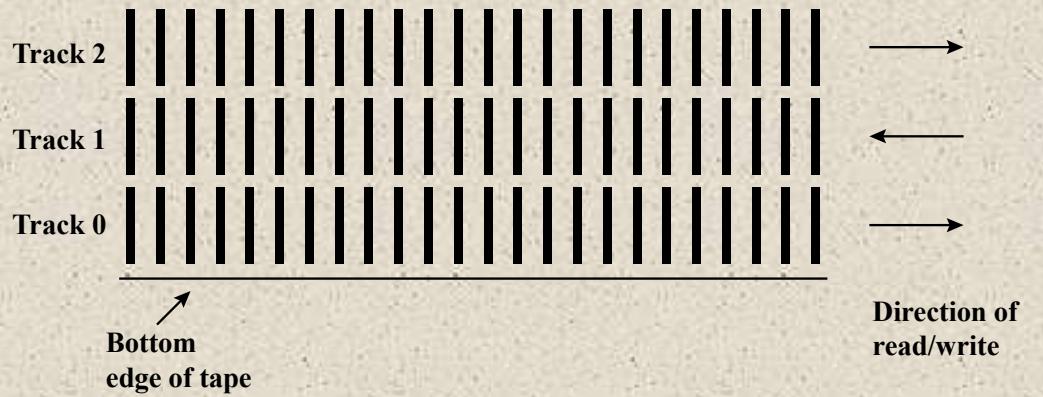
Figure 6.12 Optical Memory Characteristics

+ Magnetic Tape

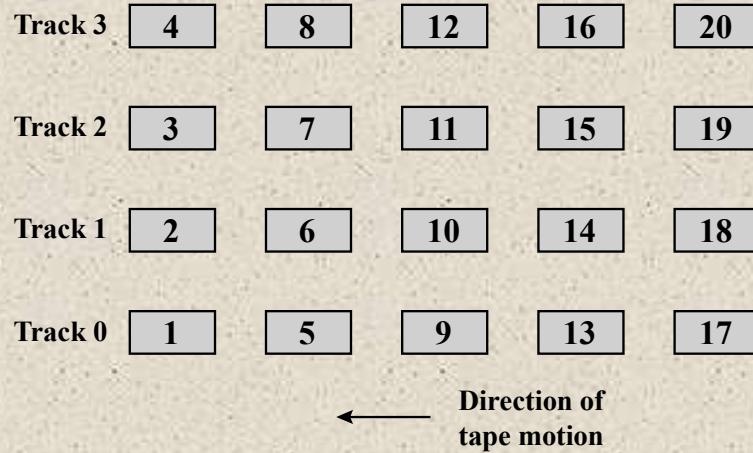


- Tape systems use the same reading and recording techniques as disk systems
- Medium is flexible polyester tape coated with magnetizable material
- Coating may consist of particles of pure metal in special binders or vapor-plated metal films
- Data on the tape are structured as a number of parallel tracks running lengthwise
- Serial recording
 - Data are laid out as a sequence of bits along each track
- Data are read and written in contiguous blocks called *physical records*
- Blocks on the tape are separated by gaps referred to as *inter-record gaps*





(a) Serpentine reading and writing



(b) Block layout for system that reads/writes four tracks simultaneously

Figure 6.13 Typical Magnetic Tape Features

Table 6.7

LTO Tape Drives

	LTO-1	LTO-2	LTO-3	LTO-4	LTO-5	LTO-6	LTO-7	LTO-8
Release date	2000	2003	2005	2007	2010	TBA	TBA	TBA
Compressed capacity	200 GB	400 GB	800 GB	1600 GB	3.2 TB	8 TB	16 TB	32 TB
Compressed transfer rate (MB/s)	40 MB/s	80 MB/s	160 MB/s	240 MB/s	280 MB/s	525 MB/s	788 MB/s	1.18 GB/s
Linear density (bits/mm)	4880	7398	9638	13250	15142			
Tape tracks	384	512	704	896	1280			
Tape length	609 m	609 m	680 m	820 m	846 m			
Tape width (cm)	1.27	1.27	1.27	1.27	1.27			
Write elements	8	8	16	16	16			
WORM?	No	No	Yes	Yes	Yes	Yes	Yes	Yes
Encryption Capable?	No	No	No	Yes	Yes	Yes	Yes	Yes
Partitioning?	No	No	No	No	Yes	Yes	Yes	Yes

+ Summary

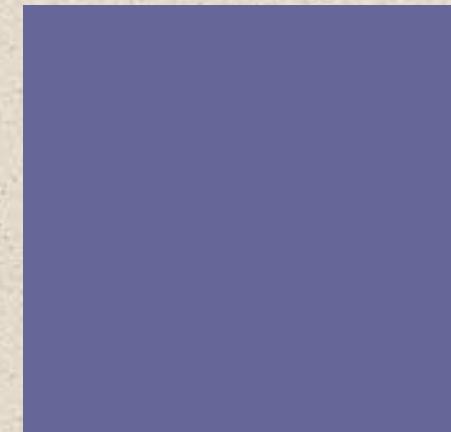
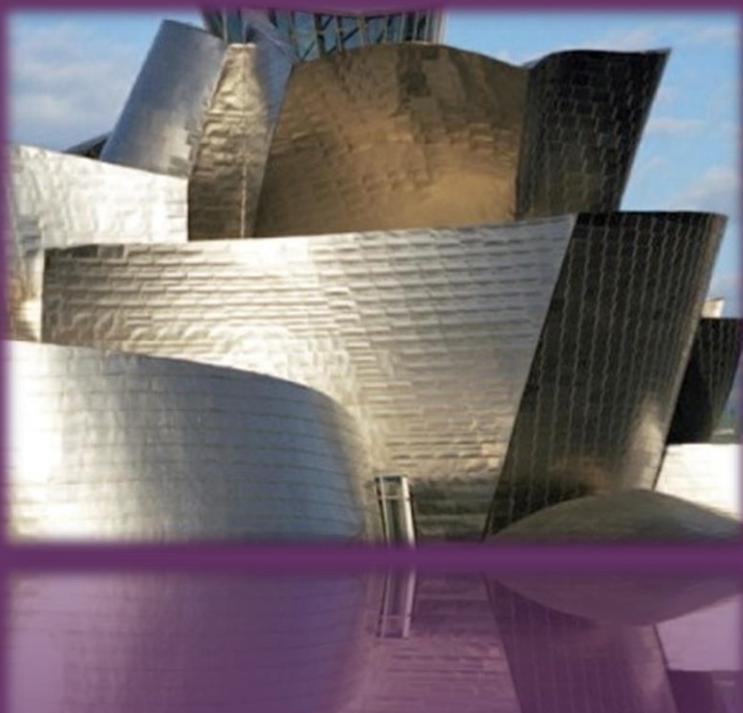
Chapter 6

- Magnetic disk
 - Magnetic read and write mechanisms
 - Data organization and formatting
 - Physical characteristics
 - Disk performance parameters
- Solid state drives
 - SSD compared to HDD
 - SSD organization
 - Practical issues
- Magnetic tape

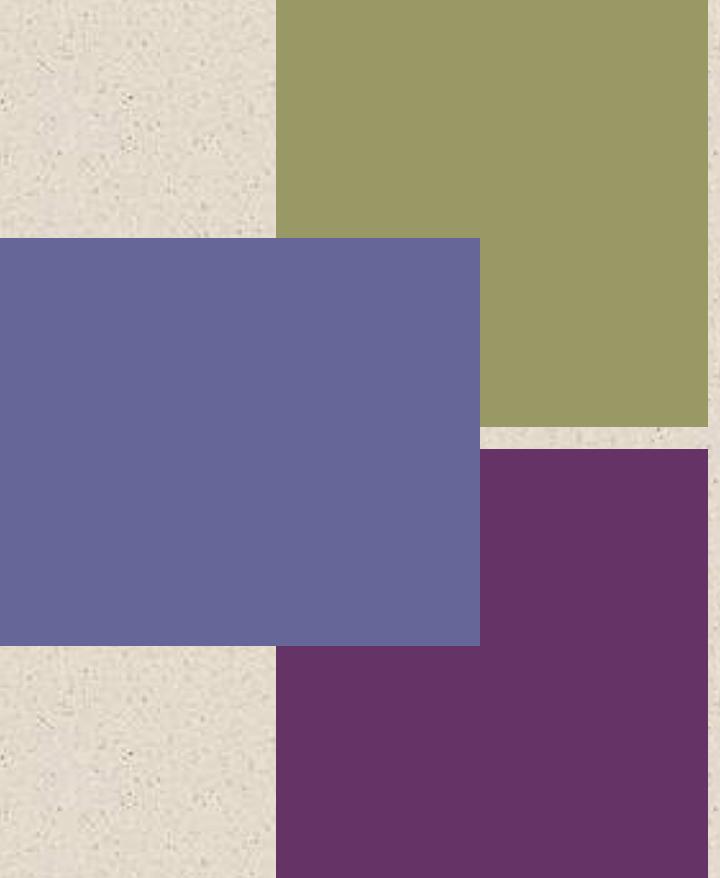
External Memory

- RAID
 - RAID level 0
 - RAID level 1
 - RAID level 2
 - RAID level 3
 - RAID level 4
 - RAID level 5
 - RAID level 6
- Optical memory
 - Compact disk
 - Digital versatile disk
 - High-definition optical disks

+



William Stallings
Computer Organization
and Architecture
10th Edition



+ Chapter 7

Input/Output

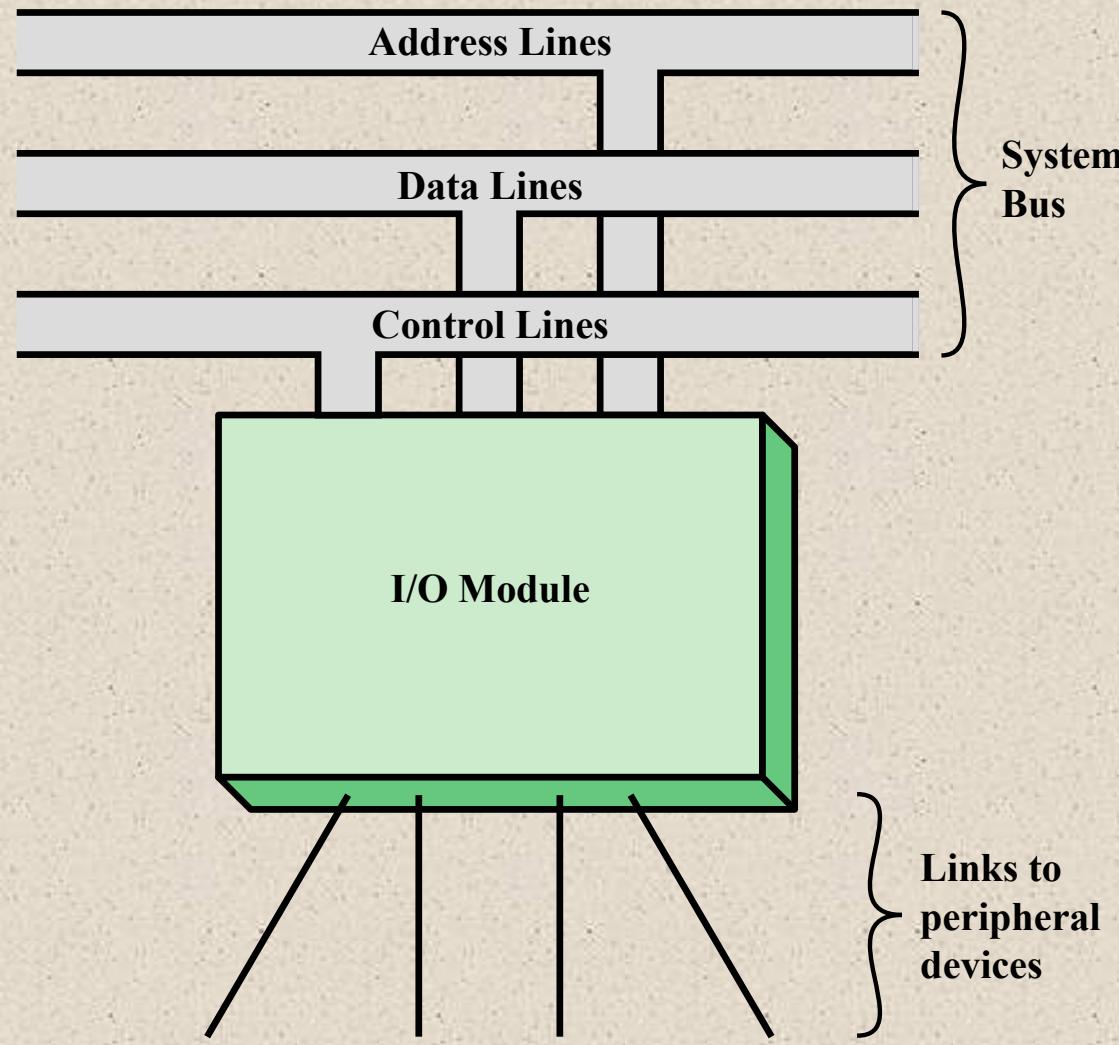


Figure 7.1 Generic Model of an I/O Module



External Devices

- Provide a means of exchanging data between the external environment and the computer
- Attach to the computer by a link to an I/O module
 - The link is used to exchange control, status, and data between the I/O module and the external device
- *Peripheral device*
 - An external device connected to an I/O module

Three categories:

- Human readable
 - Suitable for communicating with the computer user
 - Video display terminals (VDTs), printers
- Machine readable
 - Suitable for communicating with equipment
 - Magnetic disk and tape systems, sensors and actuators
- Communication
 - Suitable for communicating with remote devices such as a terminal, a machine readable device, or another computer



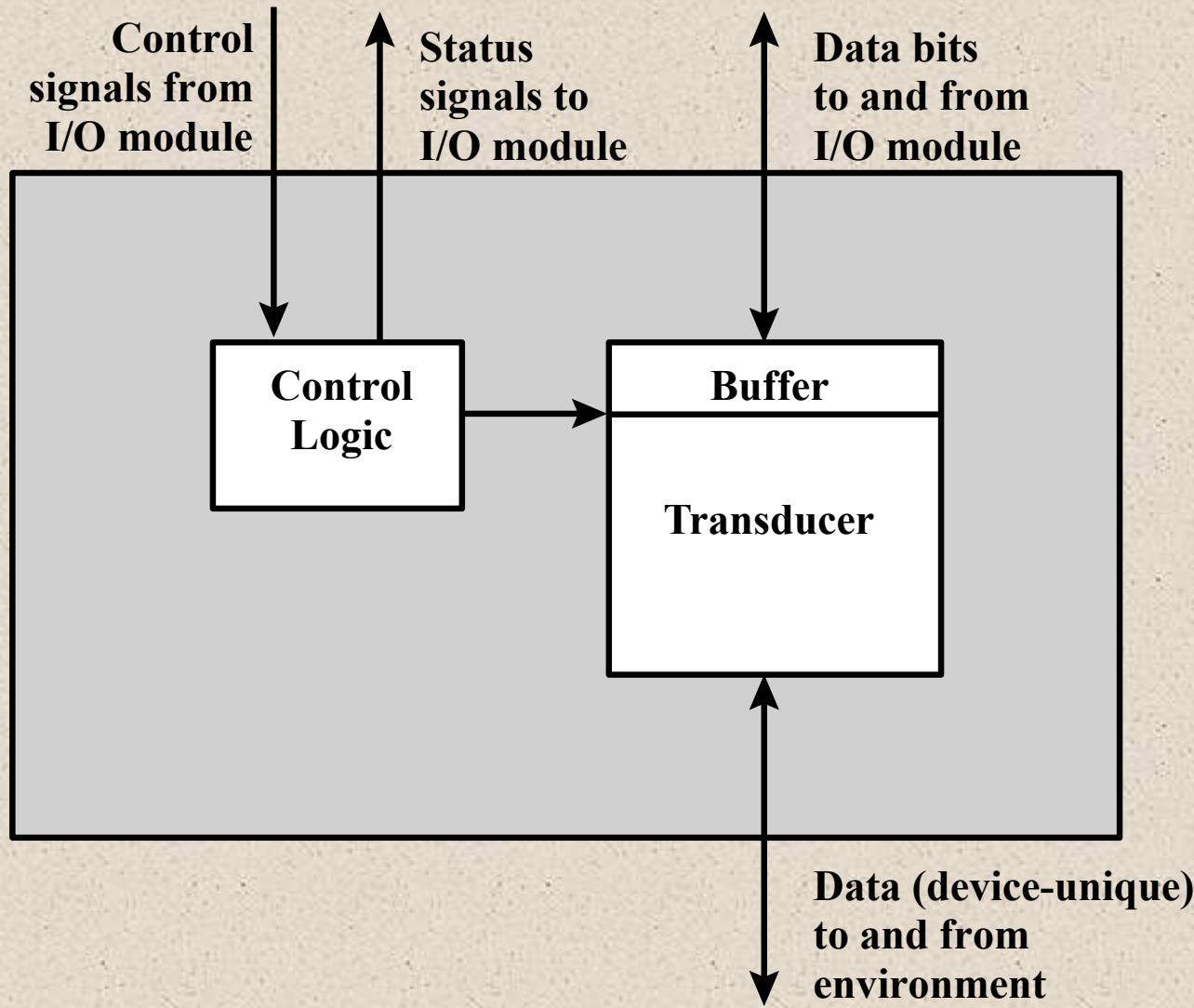


Figure 7.2 Block Diagram of an External Device

+

Keyboard/Monitor

International Reference Alphabet (IRA)

- Basic unit of exchange is the character
 - Associated with each character is a code
 - Each character in this code is represented by a unique 7-bit binary code
 - 128 different characters can be represented
- Characters are of two types:
 - Printable
 - Alphabetic, numeric, and special characters that can be printed on paper or displayed on a screen
 - Control
 - Have to do with controlling the printing or displaying of characters
 - Example is carriage return
 - Other control characters are concerned with communications procedures

Most common means of computer/user interaction

User provides input through the keyboard

The monitor displays data provided by the computer

Keyboard Codes

- When the user depresses a key it generates an electronic signal that is interpreted by the transducer in the keyboard and translated into the bit pattern of the corresponding IRA code
- This bit pattern is transmitted to the I/O module in the computer
- On output, IRA code characters are transmitted to an external device from the I/O module
- The transducer interprets the code and sends the required electronic signals to the output device either to display the indicated character or perform the requested control function

The major functions for an I/O module fall into the following categories:

Control and timing

- Coordinates the flow of traffic between internal resources and external devices

Processor communication

- Involves command decoding, data, status reporting, address recognition

Device communication

- Involves commands, status information, and data

Data buffering

- Performs the needed buffering operation to balance device and memory speeds

Error detection

- Detects and reports transmission errors

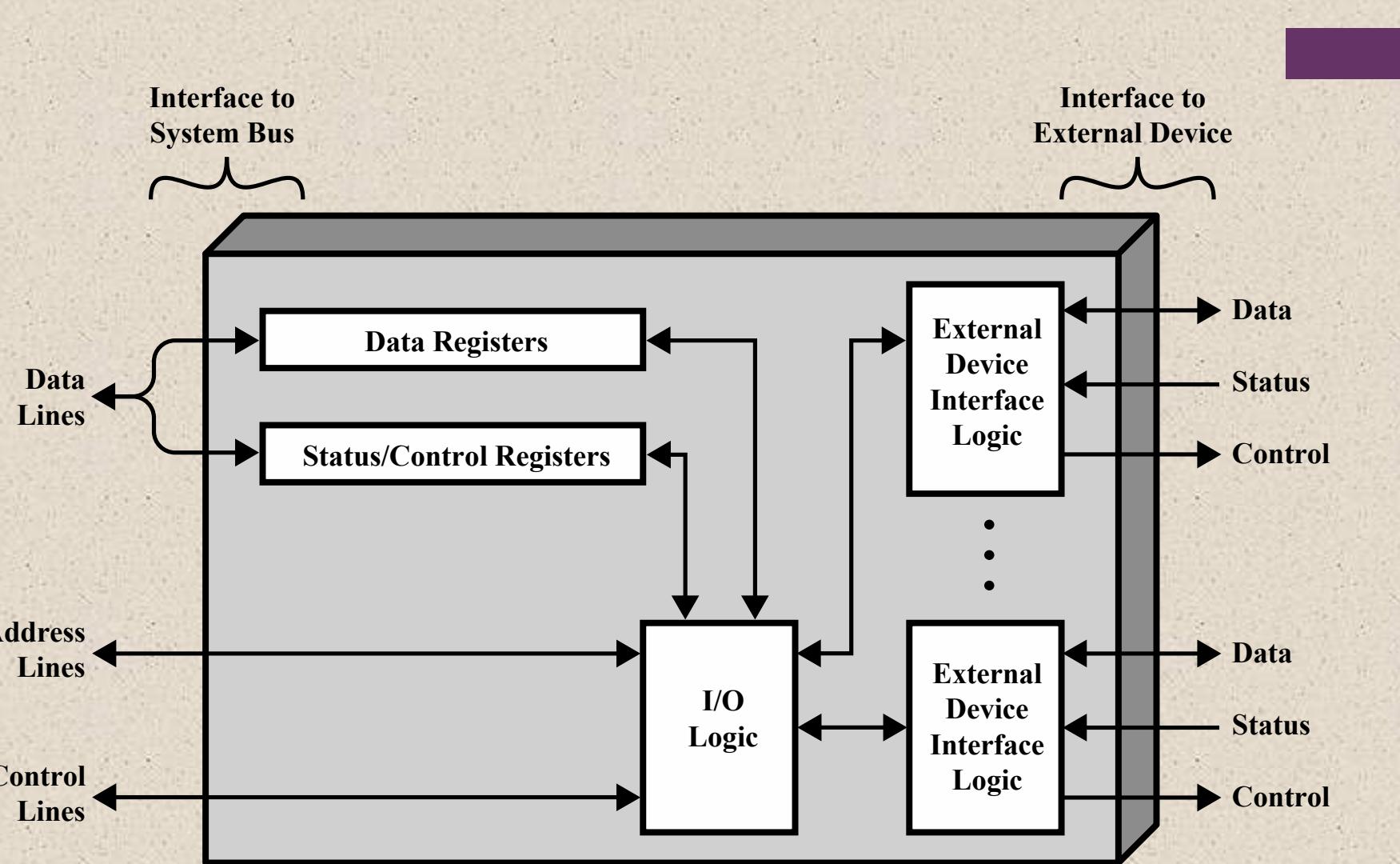


Figure 7.3 Block Diagram of an I/O Module

+ Programmed I/O

Three techniques are possible for I/O operations:

- Programmed I/O
 - Data are exchanged between the processor and the I/O module
 - Processor executes a program that gives it direct control of the I/O operation
 - When the processor issues a command it must wait until the I/O operation is complete
 - If the processor is faster than the I/O module this is wasteful of processor time
- Interrupt-driven I/O
 - Processor issues an I/O command, continues to execute other instructions, and is interrupted by the I/O module when the latter has completed its work
- Direct memory access (DMA)
 - The I/O module and main memory exchange data directly without processor involvement

Table 7.1

I/O Techniques

	No Interrupts	Use of Interrupts
I/O-to-memory transfer through processor	Programmed I/O	Interrupt-driven I/O
Direct I/O-to-memory transfer		Direct memory access (DMA)

I/O Commands

- There are four types of I/O commands that an I/O module may receive when it is addressed by a processor:

1) Control

- used to activate a peripheral and tell it what to do

2) Test

- used to test various status conditions associated with an I/O module and its peripherals

3) Read

- causes the I/O module to obtain an item of data from the peripheral and place it in an internal buffer

4) Write

- causes the I/O module to take an item of data from the data bus and subsequently transmit that data item to the peripheral

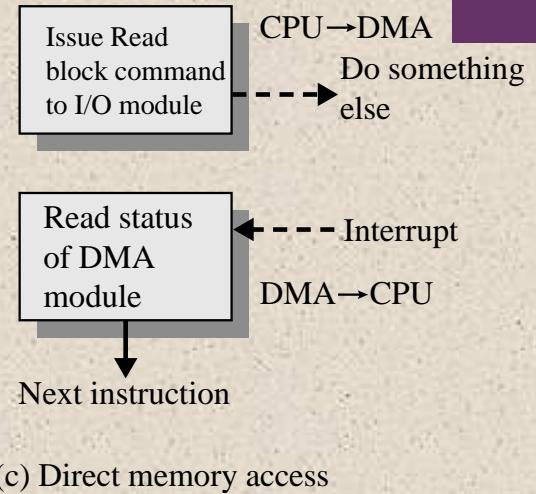
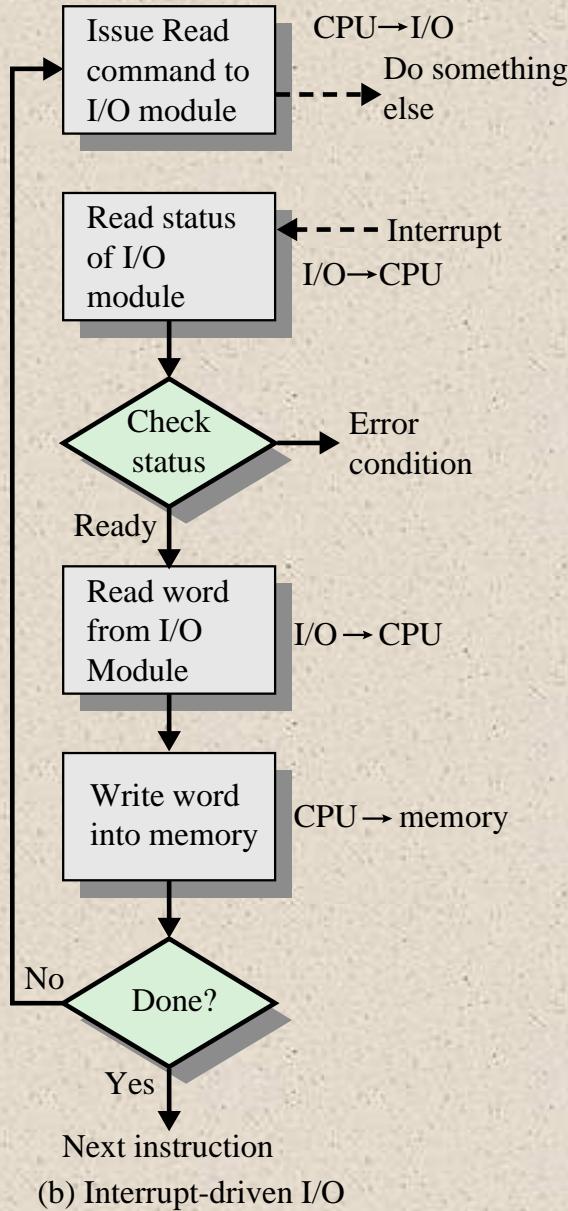
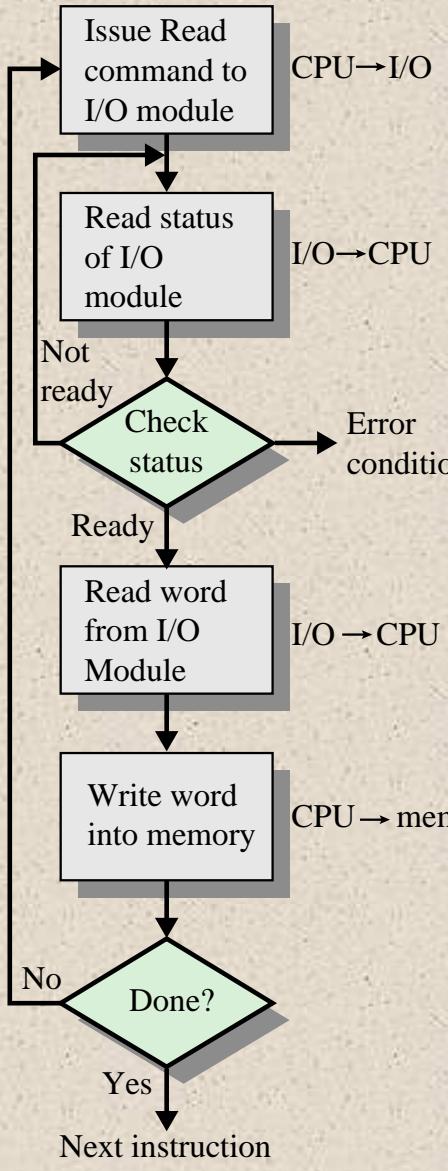


Figure 7.4 Three Techniques for Input of a Block of Data

I/O Instructions

With programmed I/O there is a close correspondence between the I/O-related instructions that the processor fetches from memory and the I/O commands that the processor issues to an I/O module to execute the instructions

The form of the instruction depends on the way in which external devices are addressed

Each I/O device connected through I/O modules is given a unique identifier or address

When the processor issues an I/O command, the command contains the address of the desired device

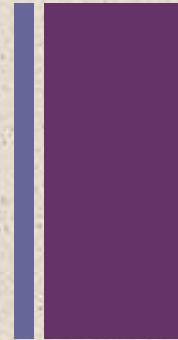
Thus each I/O module must interpret the address lines to determine if the command is for itself

Memory-mapped I/O

There is a single address space for memory locations and I/O devices

A single read line and a single write line are needed on the bus

I/O Mapping Summary

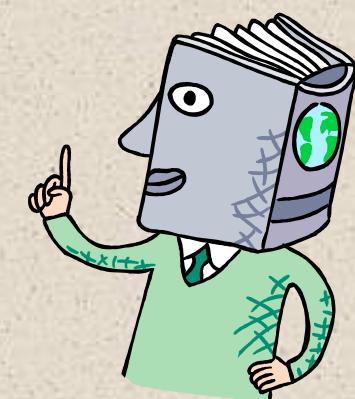


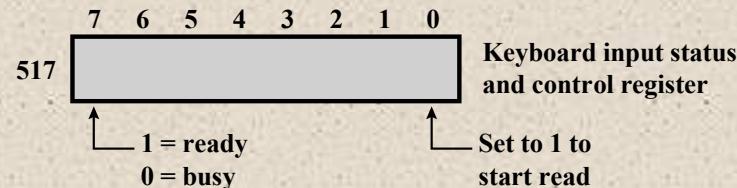
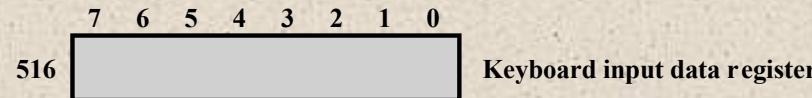
■ Memory mapped I/O

- Devices and memory share an address space
- I/O looks just like memory read/write
- No special commands for I/O
 - Large selection of memory access commands available

■ Isolated I/O

- Separate address spaces
- Need I/O or memory select lines
- Special commands for I/O
 - Limited set





ADDRESS	INSTRUCTION	OPERAND	COMMENT
200	Load AC	"1"	Load accumulator
	Store AC	517	Initiate keyboard read
202	Load AC	517	Get status byte
	Branch if Sign = 0	202	Loop until ready
	Load AC	516	Load data byte

(a) Memory-mapped I/O

ADDRESS	INSTRUCTION	OPERAND	COMMENT
200	Load I/O	5	Initiate keyboard read
201	Test I/O	5	Check for completion
	Branch Not Ready	201	Loop until complete
	In	5	Load data byte

(b) Isolated I/O

Figure 7.5 Memory-Mapped and Isolated I/O

Interrupt-Driven I/O

The problem with programmed I/O is that the processor has to wait a long time for the I/O module to be ready for either reception or transmission of data

An alternative is for the processor to issue an I/O command to a module and then go on to do some other useful work

The I/O module will then interrupt the processor to request service when it is ready to exchange data with the processor

The processor executes the data transfer and resumes its former processing

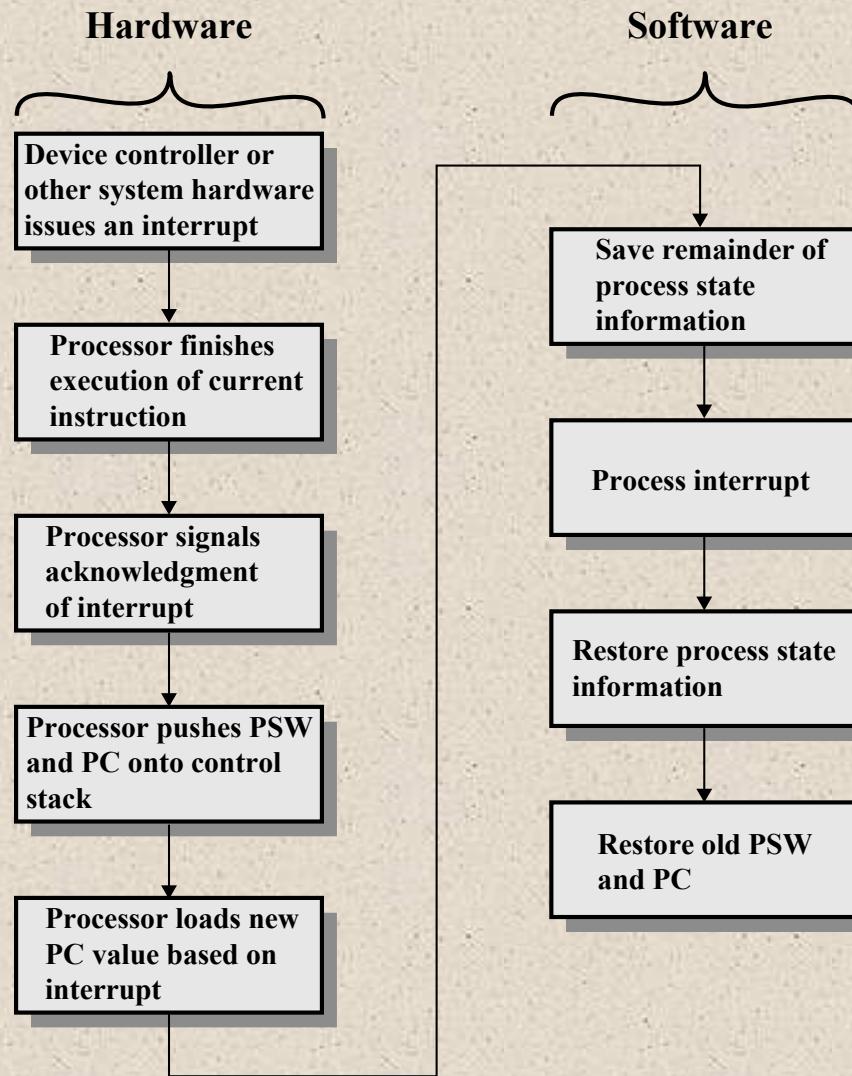


Figure 7.6 Simple Interrupt Processing

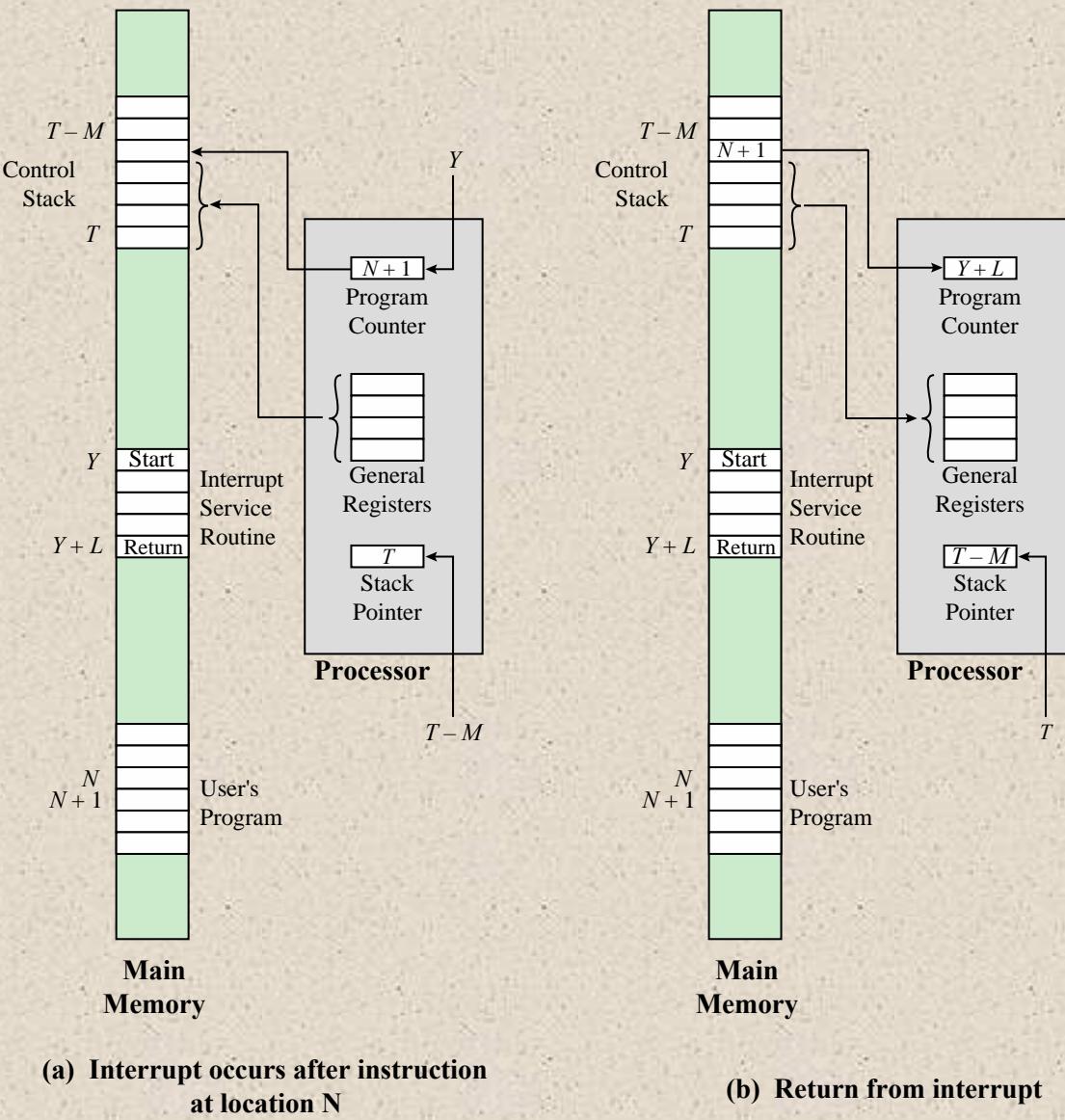


Figure 7.7 Changes in Memory and Registers for an Interrupt

Design Issues

Two design issues arise in implementing interrupt I/O:

- Because there will be multiple I/O modules how does the processor determine which device issued the interrupt?
- If multiple interrupts have occurred how does the processor decide which one to process?

+ Device Identification

Four general categories of techniques are in common use:

- **Multiple interrupt lines**
 - Between the processor and the I/O modules
 - Most straightforward approach to the problem
 - Consequently even if multiple lines are used, it is likely that each line will have multiple I/O modules attached to it
- **Software poll**
 - When processor detects an interrupt it branches to an interrupt-service routine whose job is to poll each I/O module to determine which module caused the interrupt
 - Time consuming
- **Daisy chain (hardware poll, vectored)**
 - The interrupt acknowledge line is daisy chained through the modules
 - Vector – address of the I/O module or some other unique identifier
 - Vectored interrupt – processor uses the vector as a pointer to the appropriate device-service routine, avoiding the need to execute a general interrupt-service routine first
- **Bus arbitration (vectored)**
 - An I/O module must first gain control of the bus before it can raise the interrupt request line
 - When the processor detects the interrupt it responds on the interrupt acknowledge line
 - Then the requesting module places its vector on the data lines

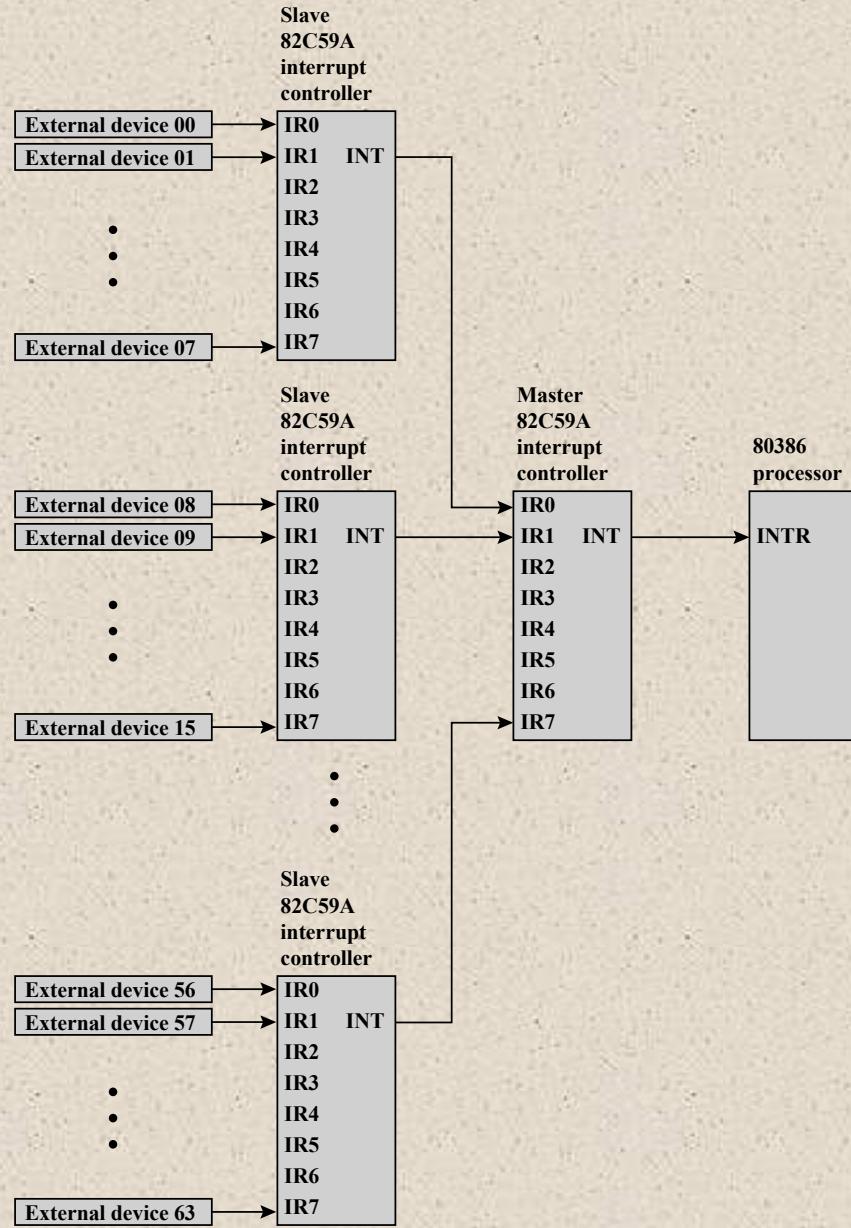
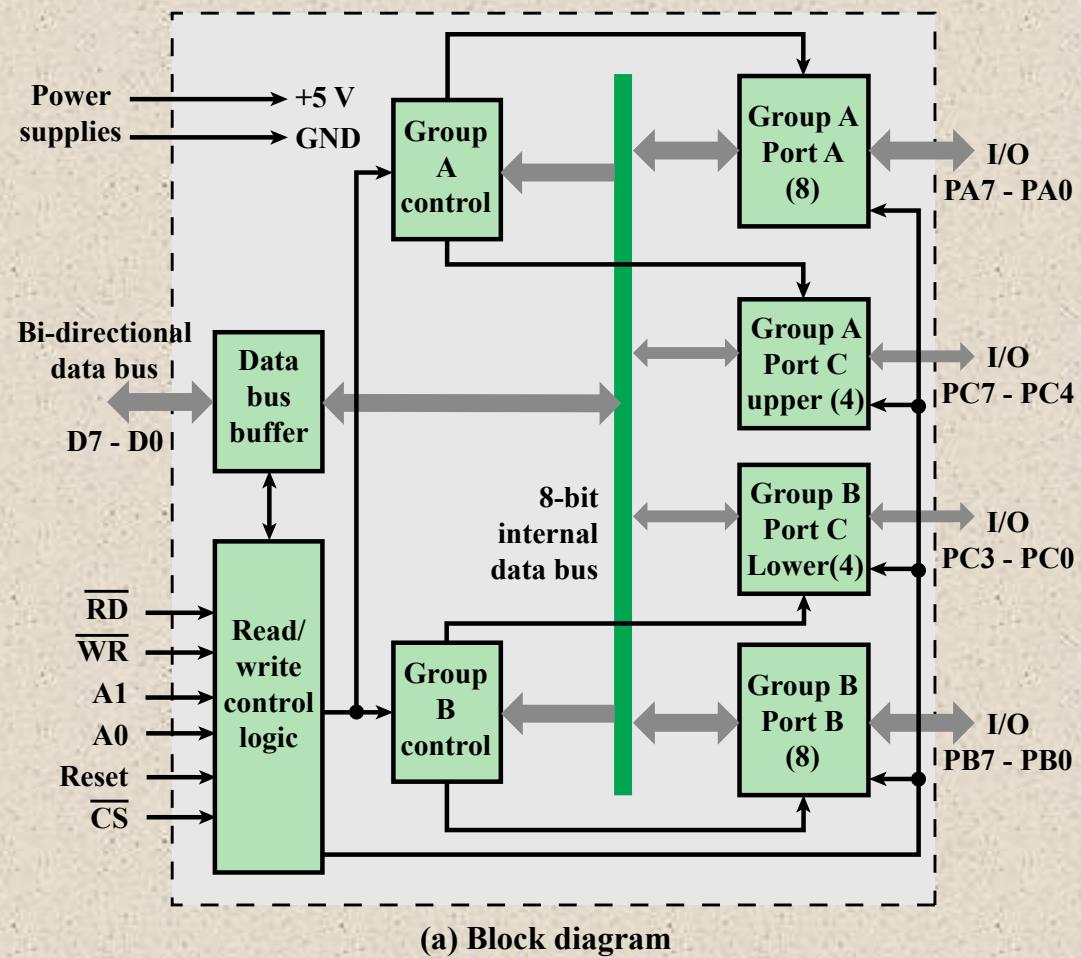


Figure 7.8 Use of the 82C59A Interrupt Controller

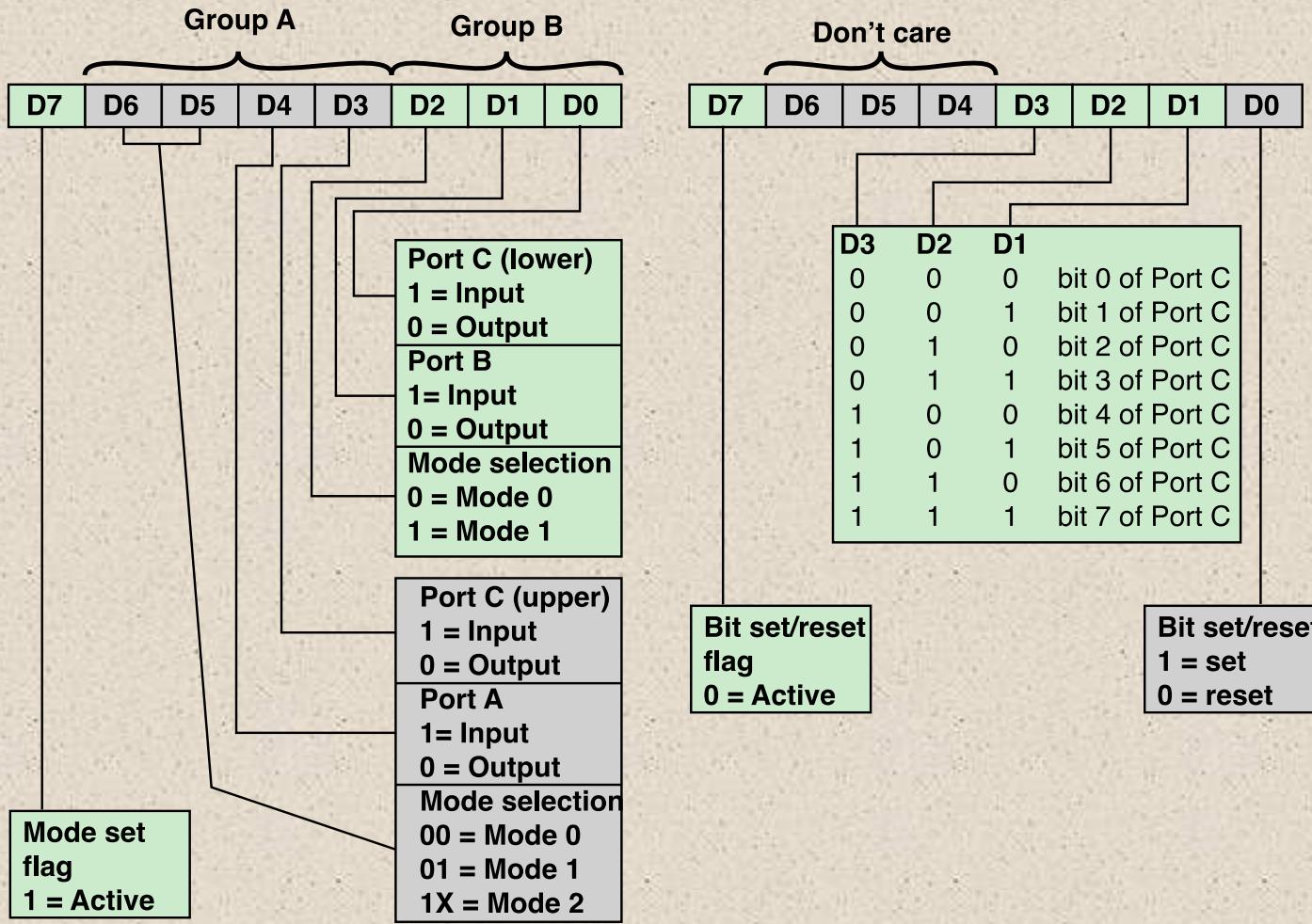


(a) Block diagram

PA3	1	40	PA4
PA2	2	39	PA5
PA1	3	38	PA6
PA0	4	37	PA7
RD	5	36	WR
CS	6	35	Reset
GND	7	34	D0
A1	8	33	D1
A0	9	8255A	32
PC7	10	31	D3
PC6	11	30	D4
PC5	12	29	D5
PC4	13	28	D6
PC3	14	27	D7
PC2	15	26	V
PC1	16	25	PB7
PC0	17	24	PB6
PB0	18	23	PB5
PB1	19	22	PB4
PB2	20	21	PB3

(b) Pin layout

Figure 7.9 The Intel 8255A Programmable Peripheral Interface



(a) Mode definition of the 8255 control register to configure the 8255

(b) Bit definitions of the 8255 control register to modify single bits of port C

Figure 7.10 The Intel 8255A Control Word

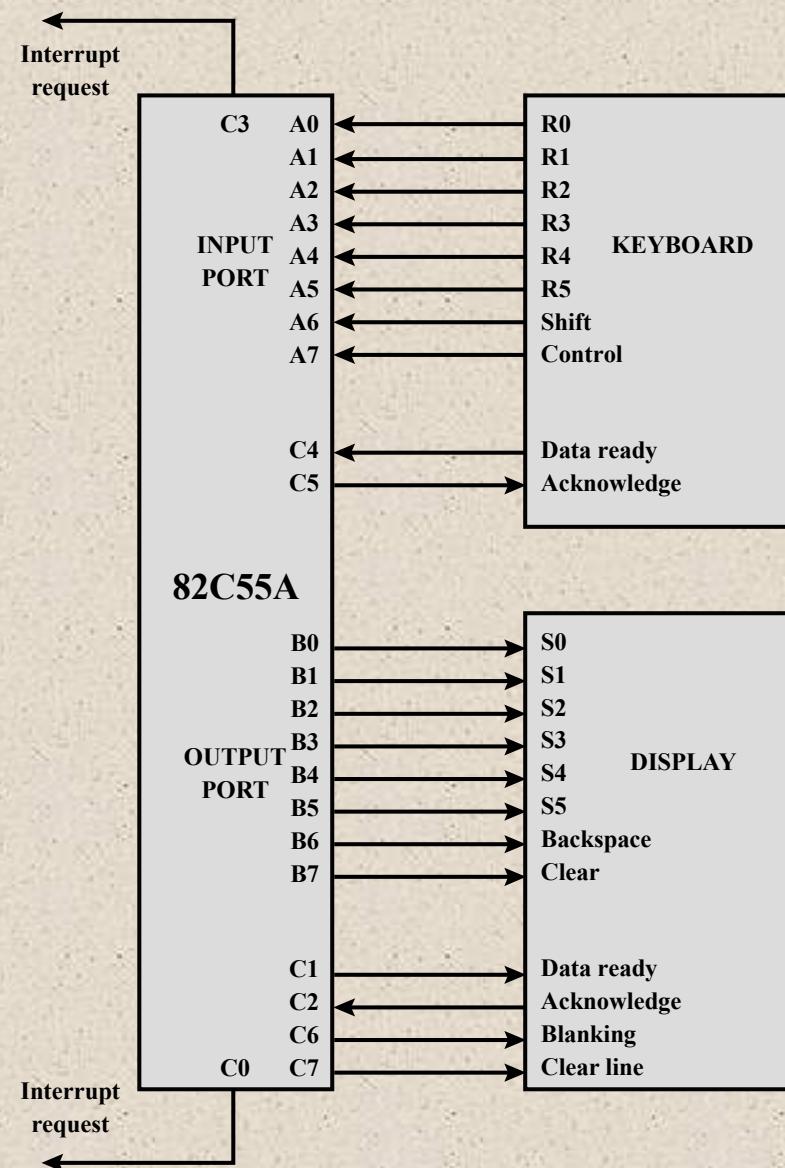


Figure 7.11 Keyboard/Display Interface to 82C55A

Drawbacks of Programmed and Interrupt-Driven I/O

- Both forms of I/O suffer from two inherent drawbacks:
 - 1) The I/O transfer rate is limited by the speed with which the processor can test and service a device
 - 2) The processor is tied up in managing an I/O transfer; a number of instructions must be executed for each I/O transfer
- When large volumes of data are to be moved a more efficient technique is *direct memory access* (DMA)

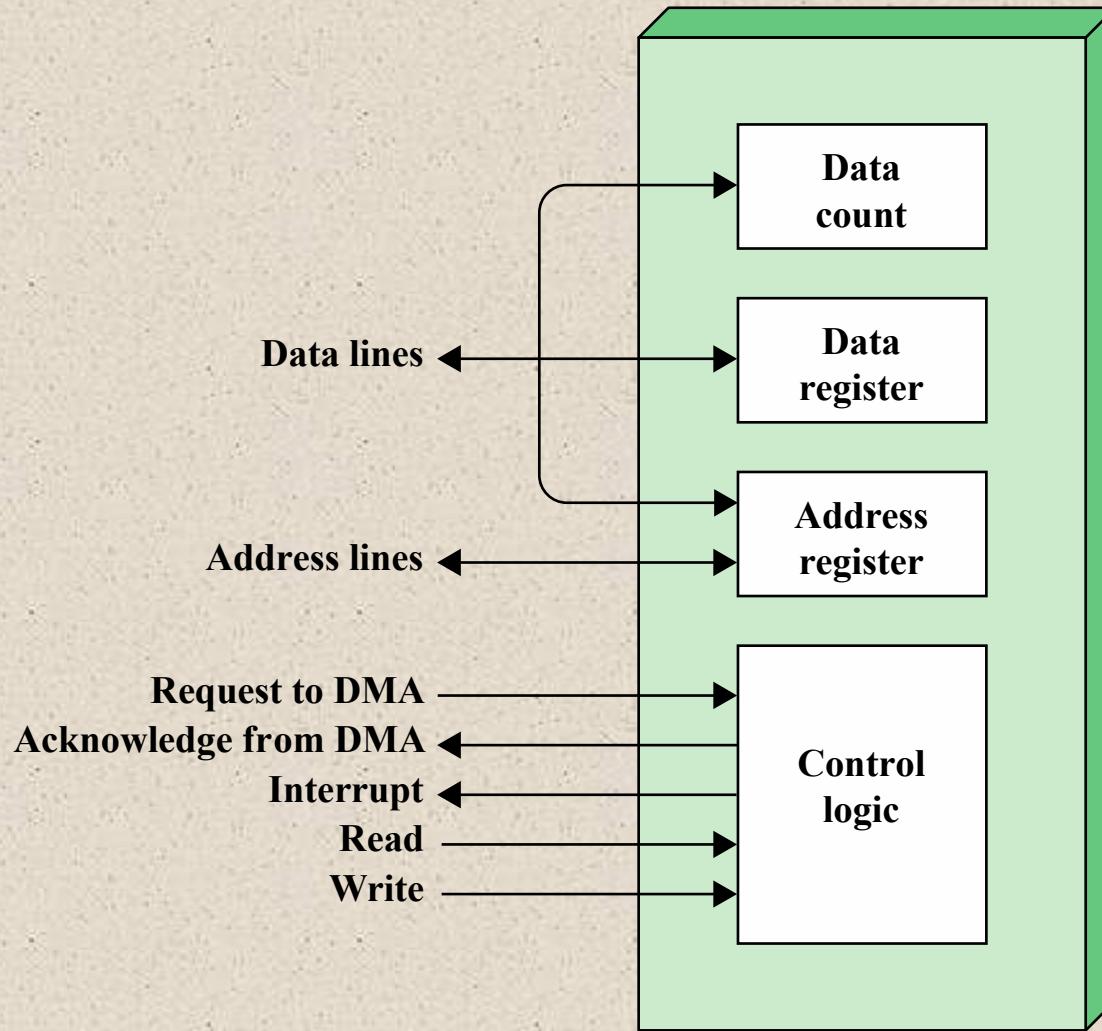


Figure 7.12 Typical DMA Block Diagram

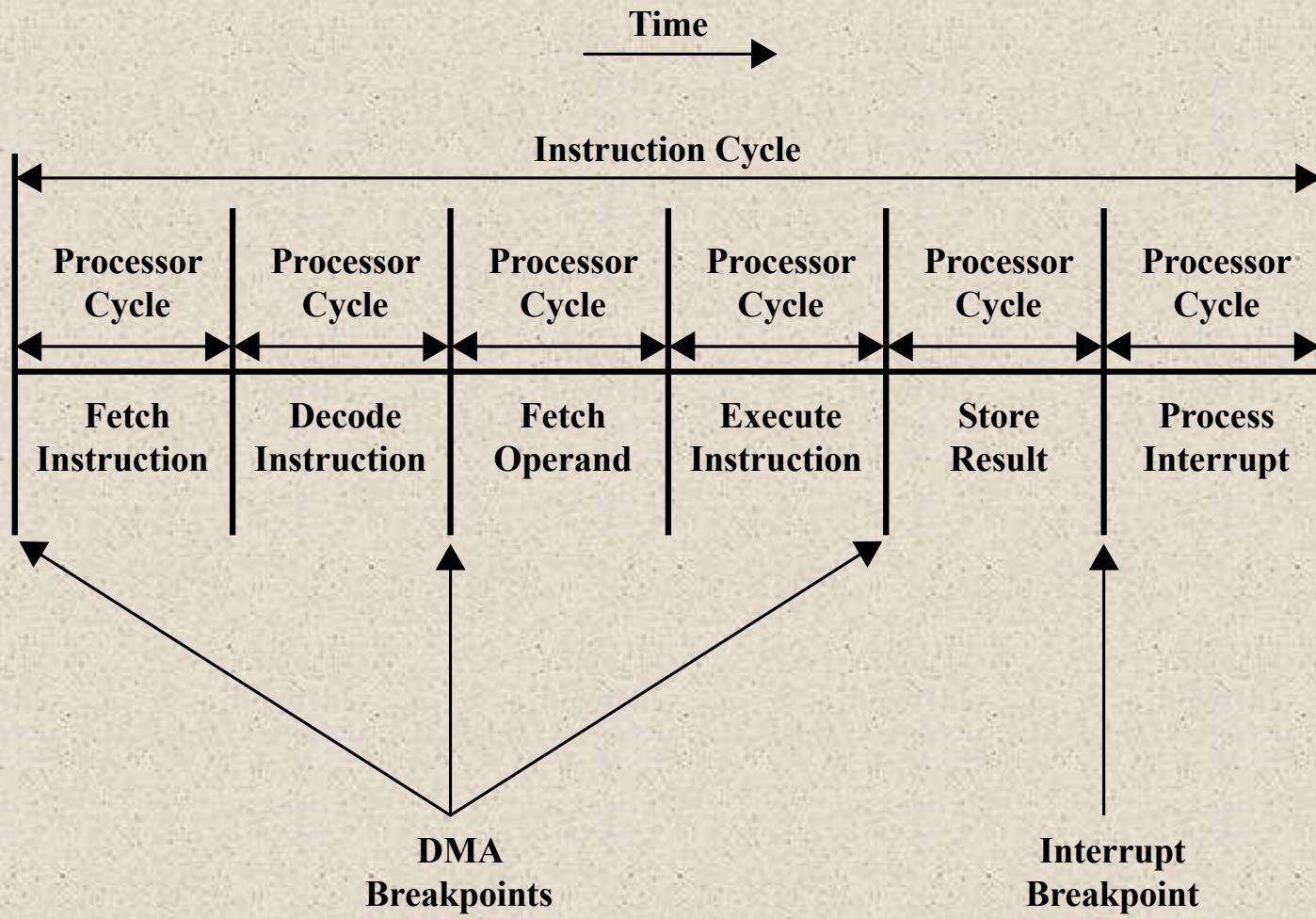
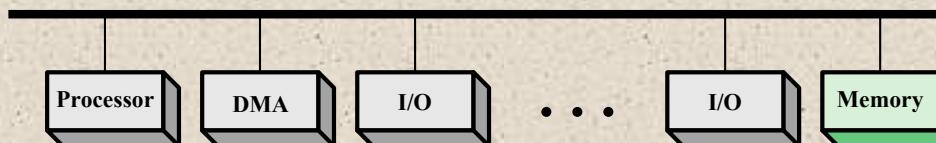
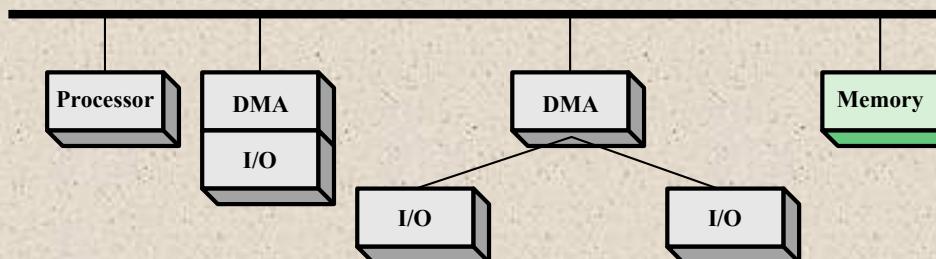


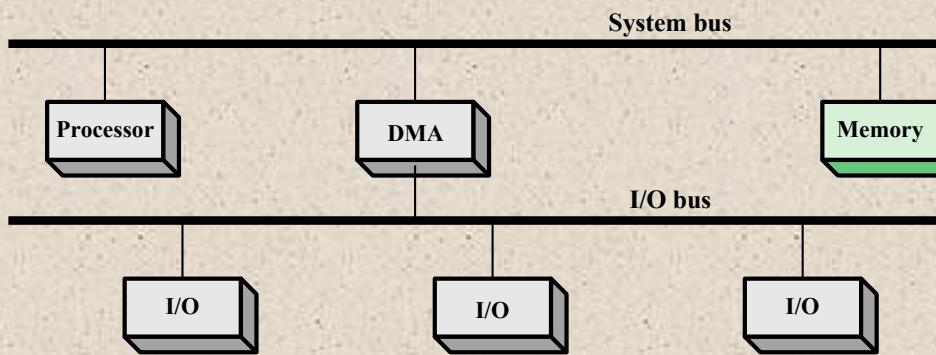
Figure 7.13 DMA and Interrupt Breakpoints During an Instruction Cycle



(a) Single-bus, detached DMA

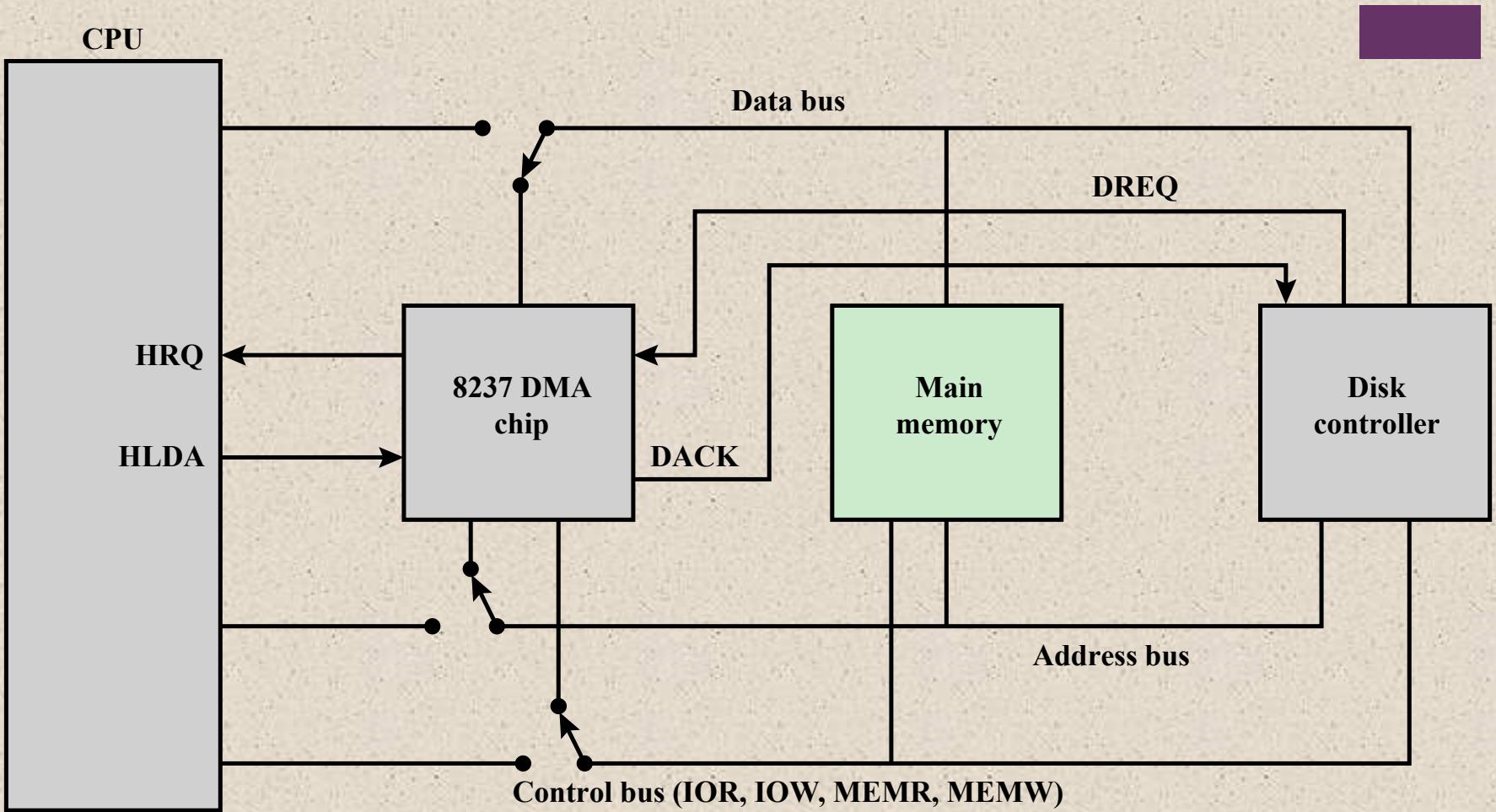


(b) Single-bus, Integrated DMA-I/O



(c) I/O bus

Figure 7.14 Alternative DMA Configurations



DACK = DMA acknowledge

DREQ = DMA request

HLDA = HOLD acknowledge

HRQ = HOLD request

Figure 7.15 8237 DMA Usage of System Bus

Fly-By DMA Controller

Data does not pass through and is not stored in DMA chip

- DMA only between I/O port and memory
- Not between two I/O ports or two memory locations

Can do memory to memory via register

8237 contains four DMA channels

- Programmed independently
- Any one active
- Numbered 0, 1, 2, and 3



Table 7.2
**Intel
8237A
Registers**

Bit	Command	Status	Mode	Single Mask	All Mask
D0	Memory-to-memory E/D	Channel 0 has reached TC	Channel select	Select channel mask bit	Clear/set channel 0 mask bit
D1	Channel 0 address hold E/D	Channel 1 has reached TC	Verify/write/read transfer	Clear/set mask bit	Clear/set channel 1 mask bit
D2	Controller E/D	Channel 2 has reached TC			Clear/set channel 2 mask bit
D3	Normal/compressed timing	Channel 3 has reached TC			Clear/set channel 3 mask bit
D4	Fixed/rotating priority	Channel 0 request	Auto-initialization E/D	Not used	Not used
D5	Late/extended write selection	Channel 0 request	Address increment/decrement select		
D6	DREQ sense active high/low	Channel 0 request			
D7	DACK sense active high/low	Channel 0 request	Demand/single/block/cascade mode select		

E/D = enable/disable

TC = terminal count



Direct Cache Access (DCA)

- DMA is not able to scale to meet the increased demand due to dramatic increases in data rates for network I/O
- Demand is coming primarily from the widespread deployment of 10-Gbps and 100-Gbps Ethernet switches to handle massive amounts of data transfer to and from database servers and other high-performance systems
- Another source of traffic comes from Wi-Fi in the gigabit range
- Network Wi-Fi devices that handle 3.2 Gbps and 6.76 Gbps are becoming widely available and producing demand on enterprise systems

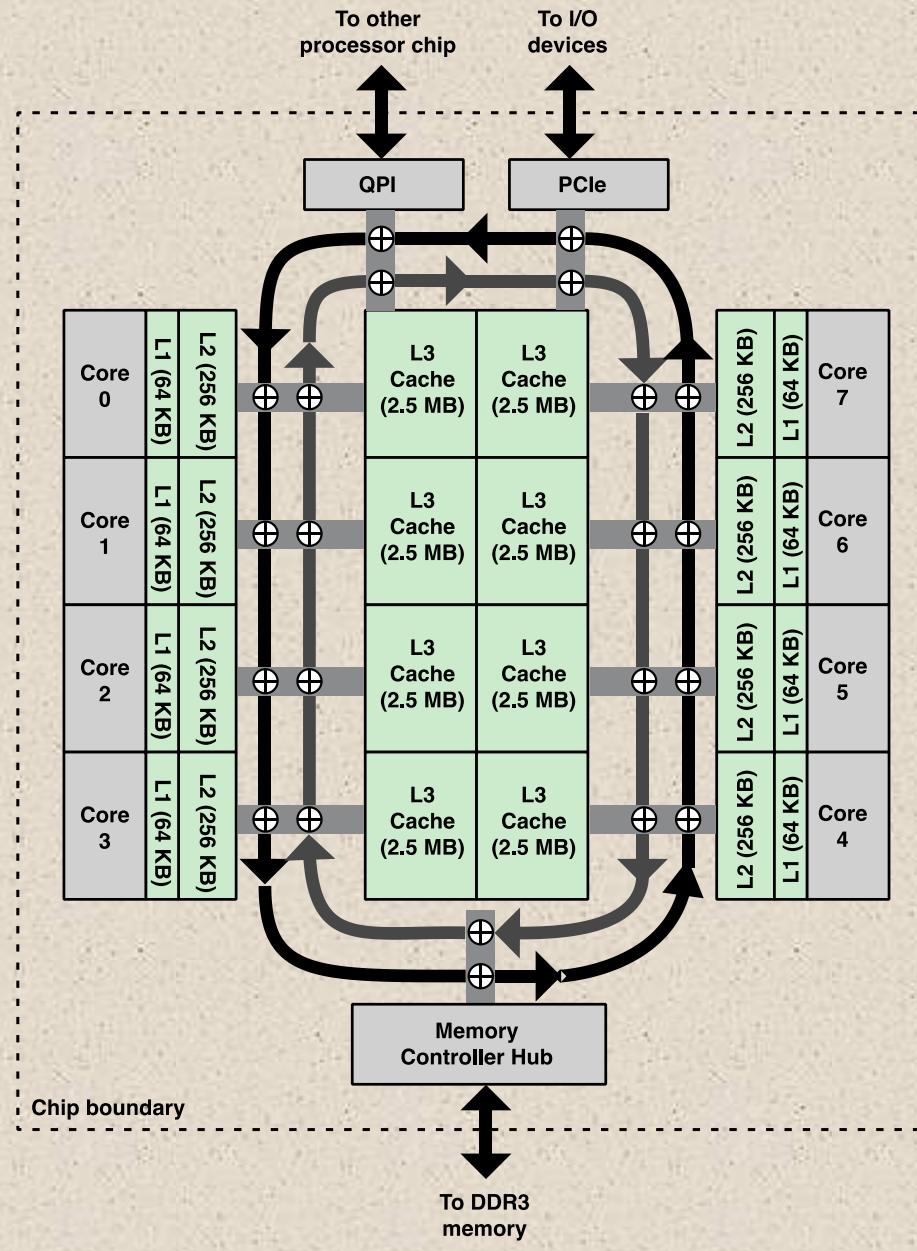
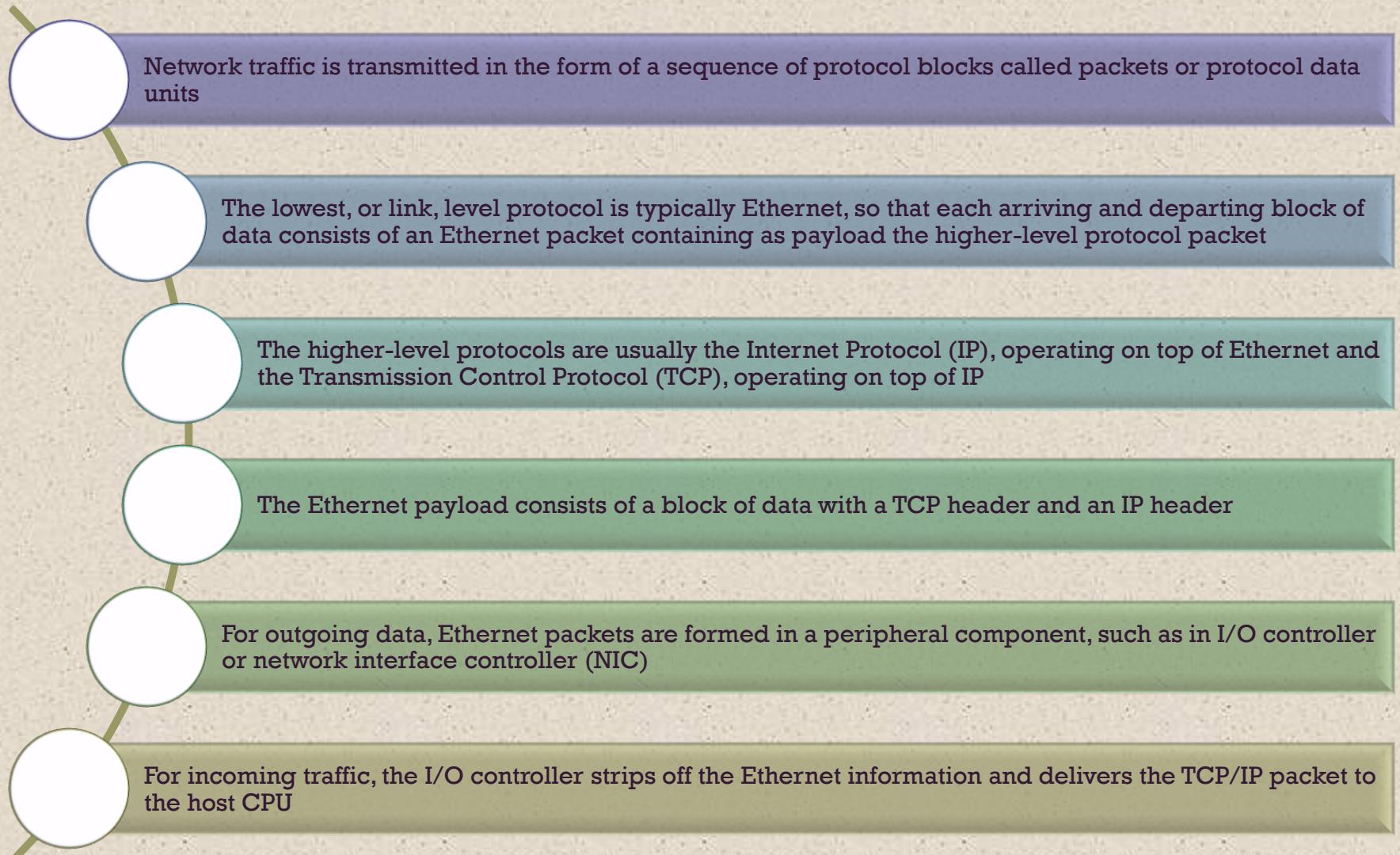


Figure 7.16 Xeon E5-2600/4600 Chip Architecture

Cache-Related Performance Issues



Cache-Related Performance Issues

For both outgoing and incoming traffic the core, main memory, and cache are all involved

In a DMA scheme, when an application wishes to transmit data, it places that data in an application-assigned buffer in main memory

- The core transfers this to a system buffer in main memory and creates the necessary TCP and IP headers, which are also buffered in system memory
- The packet is then picked up via DMA for transfer via the NIC
- This activity engages not only main memory but also the cache
- Similar transfers between system and application buffers are required for incoming traffic



Packet Traffic Steps:

Incoming

- Packet arrives
- DMA
- NIC interrupts host
- Retrieve descriptors and headers
- Cache miss occurs
- Header is processed
- Payload transferred

Outgoing

- Packet transfer requested
- Packet created
- Output operation invoked
- DMA transfer
- NIC signals completion
- Driver frees buffer



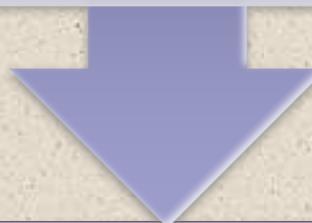
Direct Cache Access Strategies

Simplest strategy was implemented as a prototype on a number of Intel Xeon processors between 2006 and 2010

This form of DCA applies only to incoming network traffic

The DCA function in the memory controller sends a prefetch hint to the core as soon as the data is available in system memory

This enables the core to prefetch the data packet from the system buffer



Much more substantial gains can be realized by avoiding the system buffer in main memory altogether

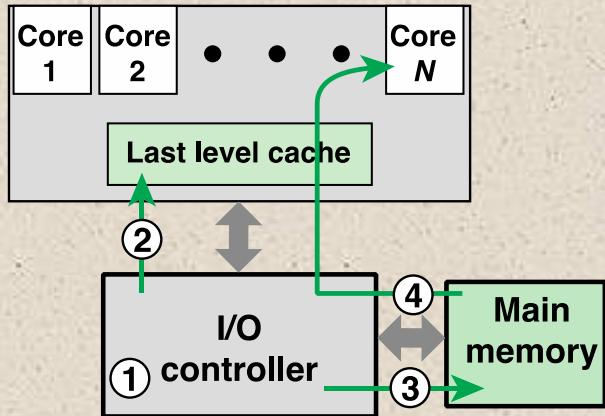
The packet and packet descriptor information are accessed only once in the system buffer by the core

For incoming packets, the core reads the data from the buffer and transfers the packet payload to an application buffer

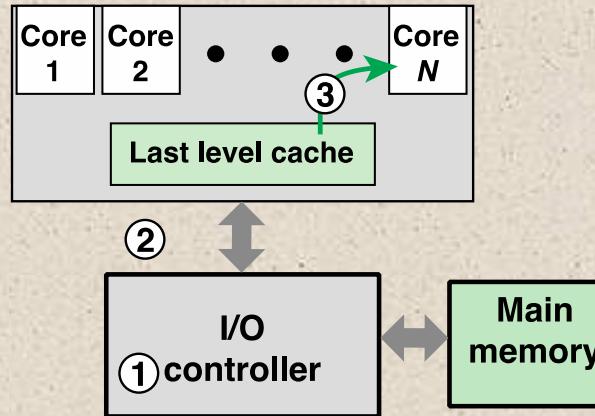
It has no need to access that data in the system buffer again

Cache injection

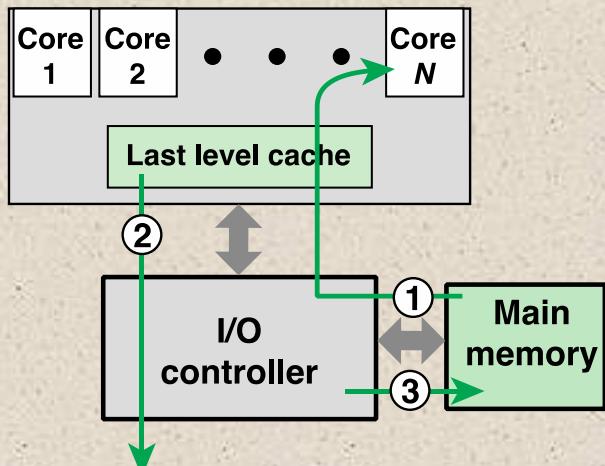
Implemented in Intel's Xeon processor line, referred to as Direct Data I/O



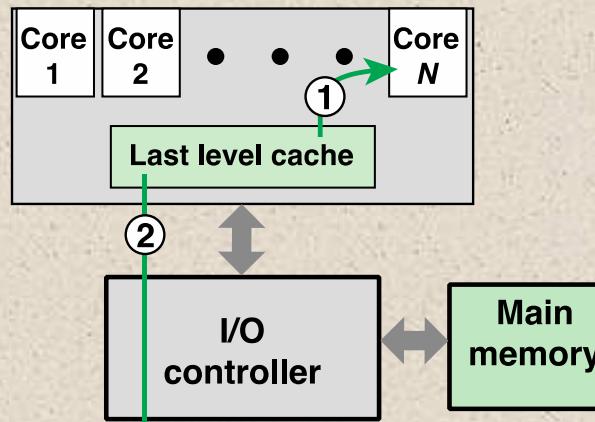
(a) Normal DMA transfer to memory



(b) DDIO transfer to cache



(c) Normal DMA transfer to I/O



(d) DDIO transfer to I/O

Figure 7.17 Comparison of DMA and DDIO



Evolution of the I/O Function

1. The CPU directly controls a peripheral device.
2. A controller or I/O module is added. The CPU uses programmed I/O without interrupts.
3. Same configuration as in step 2 is used, but now interrupts are employed. The CPU need not spend time waiting for an I/O operation to be performed, thus increasing efficiency.
4. The I/O module is given direct access to memory via DMA. It can now move a block of data to or from memory without involving the CPU, except at the beginning and end of the transfer.
5. The I/O module is enhanced to become a processor in its own right, with a specialized instruction set tailored for I/O
6. The I/O module has a local memory of its own and is, in fact, a computer in its own right. With this architecture a large set of I/O devices can be controlled with minimal CPU involvement.

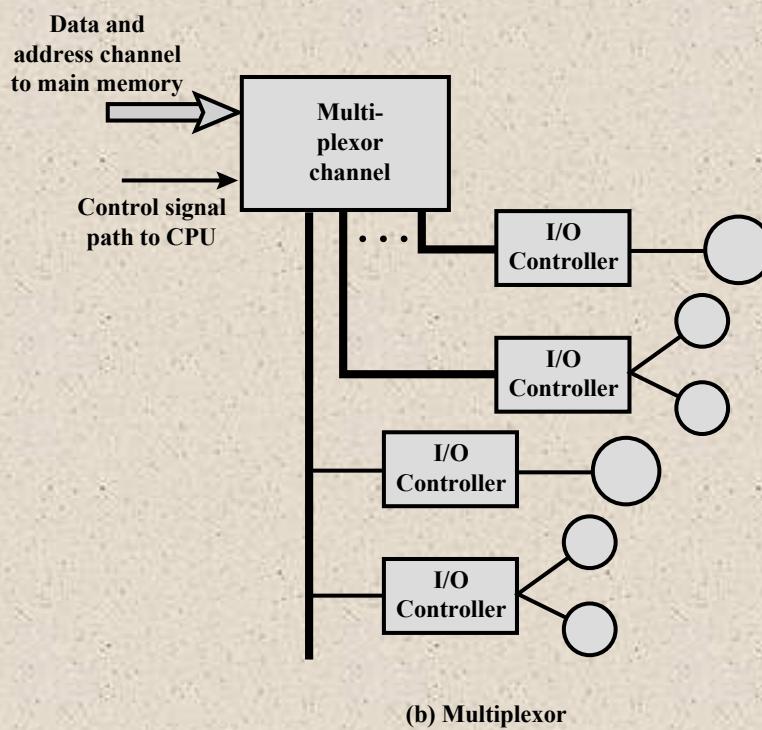
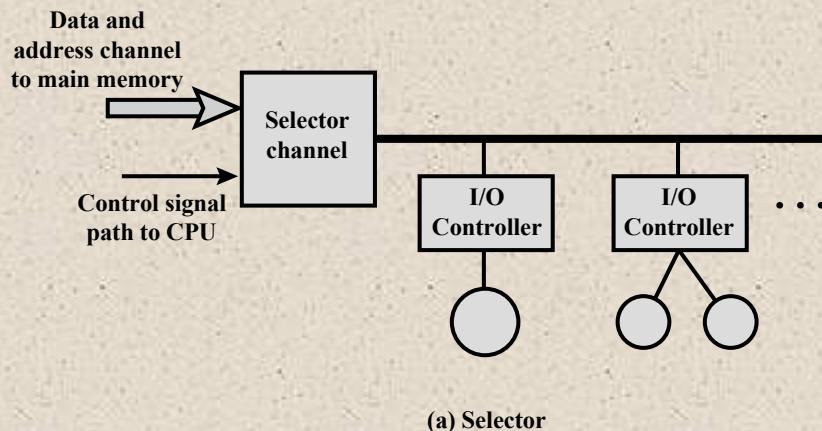


Figure 7.18 I/O Channel Architecture

+ Universal Serial Bus (USB)

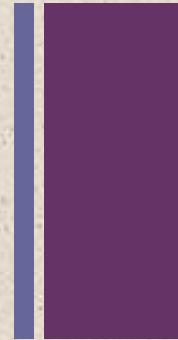
- Widely used for peripheral connections
- Is the default interface for slower speed devices
- Commonly used high-speed I/O
- Has gone through multiple generations
 - USB 1.0
 - Defined a *Low Speed* data rate of 1.5 Mbps and a *Full Speed* rate of 12 Mbps
 - USB 2.0
 - Provides a data rate of 480 Mbps
 - USB 3.0
 - Higher speed bus called *SuperSpeed* in parallel with the USB 2.0 bus
 - Signaling speed of *SuperSpeed* is 5 Gbps, but due to signaling overhead the usable data rate is up to 4 Gbps
 - USB 3.1
 - Includes a faster transfer mode called *SuperSpeed+*
 - This transfer mode achieves a signaling rate of 10 Gbps and a theoretical usable data rate of 9.7 Gbps
- Is controlled by a root host controller which attaches to devices to create a local network with a hierarchical tree topology

+ FireWire Serial Bus

- Was developed as an alternative to small computer system interface (SCSI) to be used on smaller systems, such as personal computers, workstations, and servers
- Objective was to meet the increasing demands for high I/O rates while avoiding the bulky and expensive I/O channel technologies developed for mainframe and supercomputer systems
- IEEE standard 1394, for a High Performance Serial Bus
- Uses a daisy chain configuration, with up to 63 devices connected off a single port
- 1022 FireWire buses can be interconnected using bridges
- Provides for hot plugging which makes it possible to connect and disconnect peripherals without having to power the computer system down or reconfigure the system
- Provides for automatic configuration
- No terminations and the system automatically performs a configuration function to assign addresses



SCSI



- Small Computer System Interface
- A once common standard for connecting peripheral devices to small and medium-sized computers
- Has lost popularity to USB and FireWire in smaller systems
- High-speed versions remain popular for mass memory support on enterprise systems
- Physical organization is a shared bus, which can support up to 16 or 32 devices, depending on the generation of the standard
 - The bus provides for parallel transmission rather than serial, with a bus width of 16 bits on earlier generations and 32 bits on later generations
 - Speeds range from 5 Mbps on the original SCSI-1 specification to 160 Mbps on SCSI-3 U3



Thunderbolt



- Most recent and fastest peripheral connection technology to become available for general-purpose use
- Developed by Intel with collaboration from Apple
- The technology combines data, video, audio, and power into a single high-speed connection for peripherals such as hard drives, RAID arrays, video-capture boxes, and network interfaces
- Provides up to 10 Gbps throughput in each direction and up to 10 Watts of power to connected peripherals

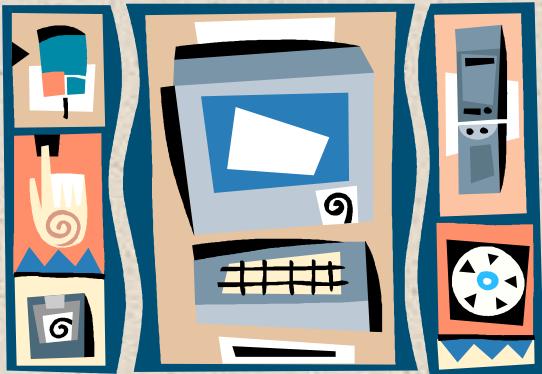


InfiniBand

- I/O specification aimed at the high-end server market
- First version was released in early 2001
- Heavily relied on by IBM zEnterprise series of mainframes
- Standard describes an architecture and specifications for data flow among processors and intelligent I/O devices
- Has become a popular interface for storage area networking and other large storage configurations
- Enables servers, remote storage, and other network devices to be attached in a central fabric of switches and links
- The switch-based architecture can connect up to 64,000 servers, storage systems, and networking devices

PCI Express

- High-speed bus system for connecting peripherals of a wide variety of types and speeds

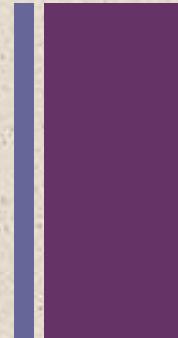


SATA

- Serial Advanced Technology Attachment
- An interface for disk storage systems
- Provides data rates of up to 6 Gbps, with a maximum per device of 300 Mbps
- Widely used in desktop computers and in industrial and embedded applications



Ethernet



- Predominant wired networking technology
- Has evolved to support data rates up to 100 Gbps and distances from a few meters to tens of km
- Has become essential for supporting personal computers, workstations, servers, and massive data storage devices in organizations large and small
- Began as an experimental bus-based 3-Mbps system
- Has moved from bus-based to switch-based
 - Data rate has periodically increased by an order of magnitude
 - There is a central switch with all of the devices connected directly to the switch
- Ethernet systems are currently available at speeds up to 100 Gbps

Wi-Fi

- Is the predominant wireless Internet access technology
- Now connects computers, tablets, smart phones, and other electronic devices such as video cameras TVs and thermostats
- In the enterprise has become an essential means of enhancing worker productivity and network effectiveness
- Public hotspots have expanded dramatically to provide free Internet access in most public places
- As the technology of antennas, wireless transmission techniques, and wireless protocol design has evolved, the IEEE 802.11 committee has been able to introduce standards for new versions of Wi-Fi at higher speeds
- Current version is 802.11ac (2014) with a maximum data rate of 3.2 Gbps



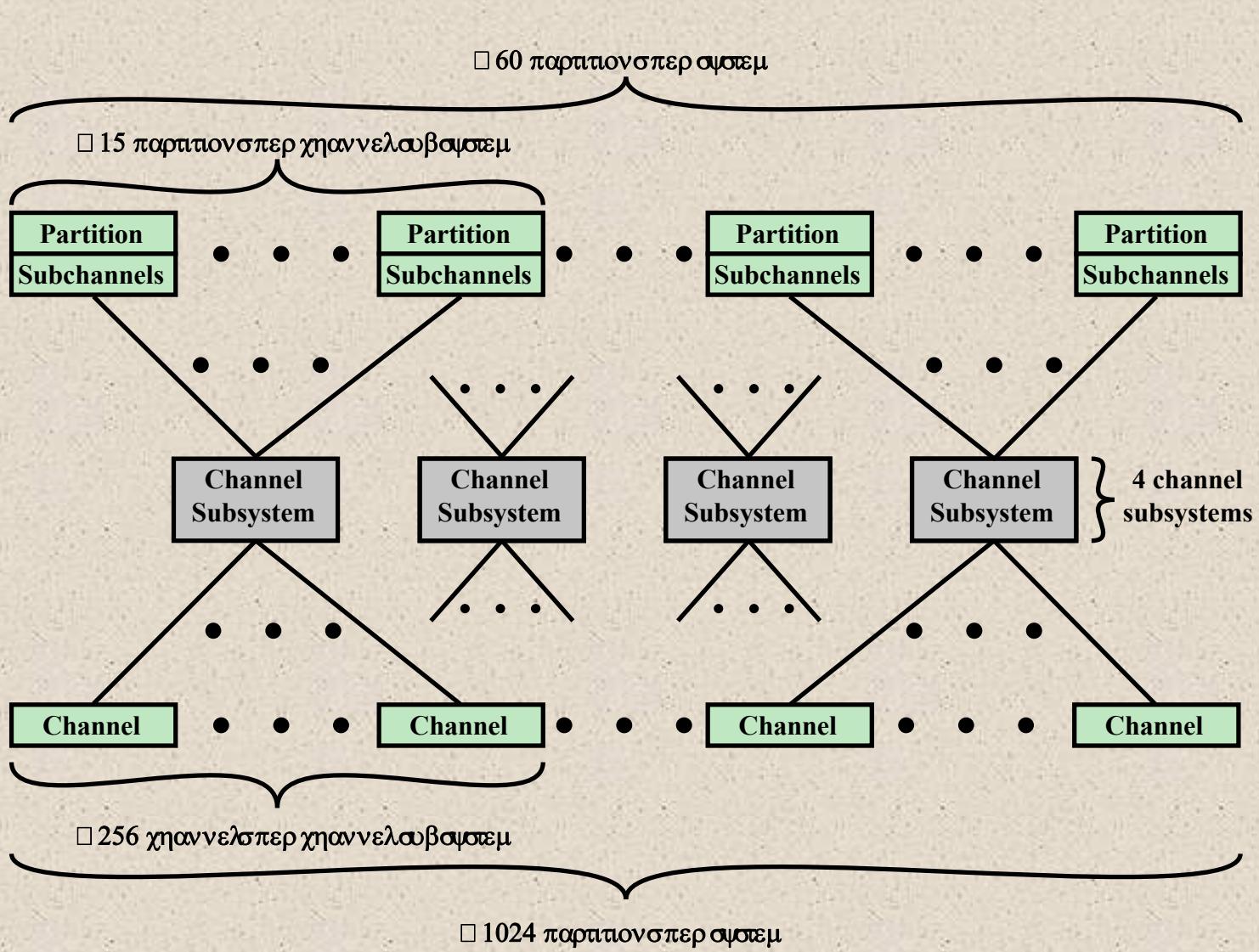
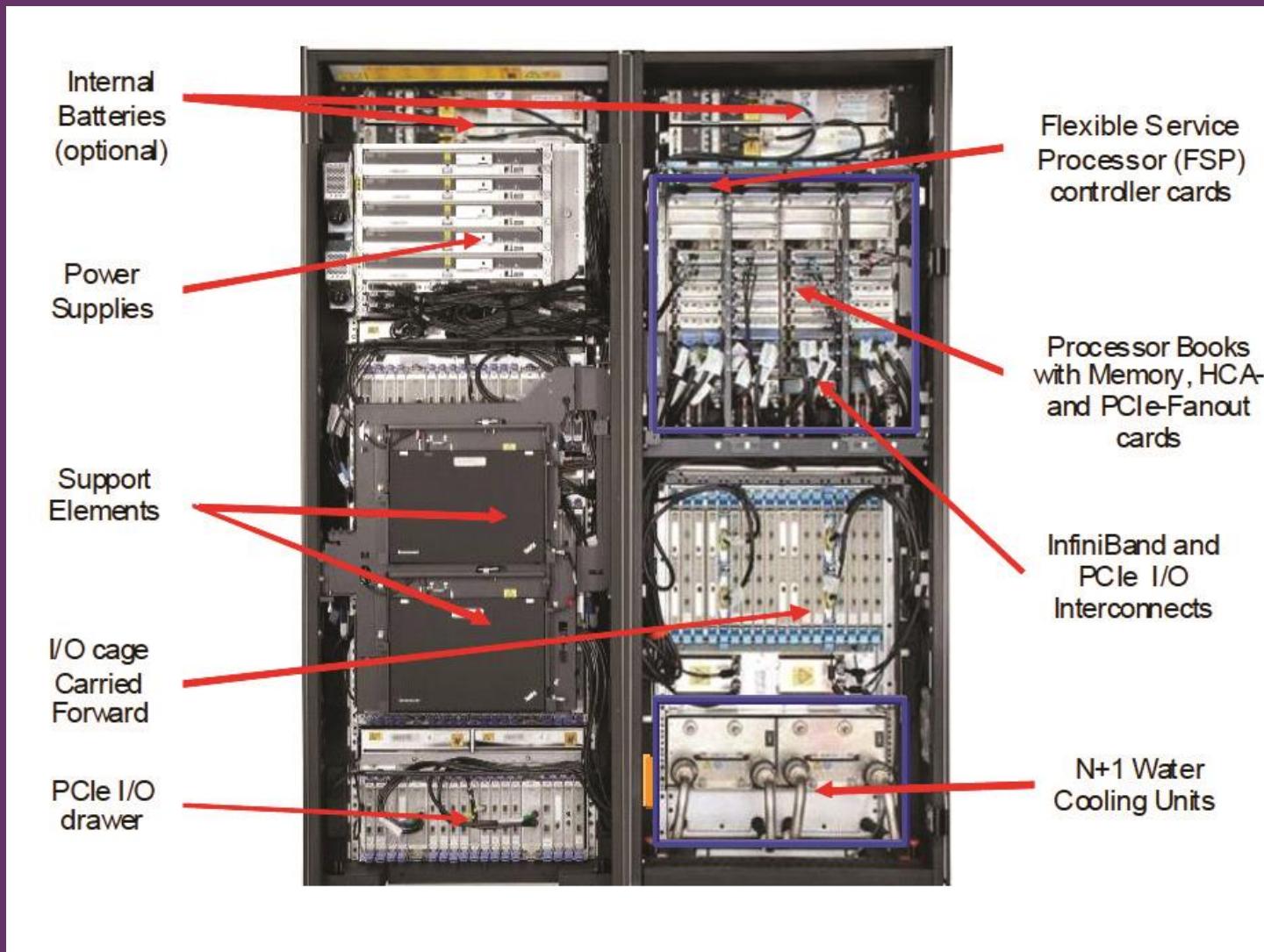


Figure 7.19 IBM EC12 I/O Channel Subsystem Structure

Figure 7.20

IBM zEC12 I/O Frames-Front View



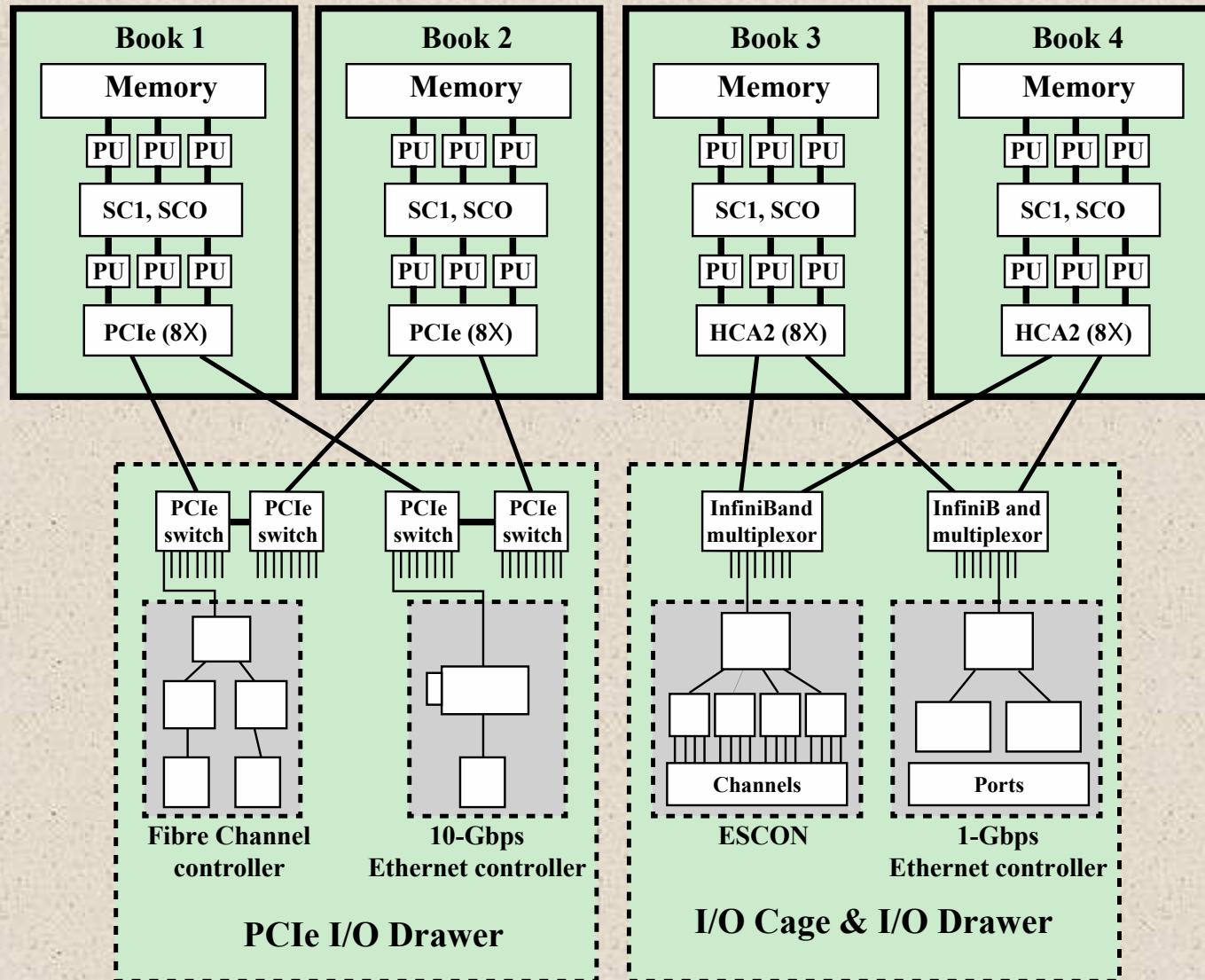


Figure 7.21 IBM EC12 I/O System Structure



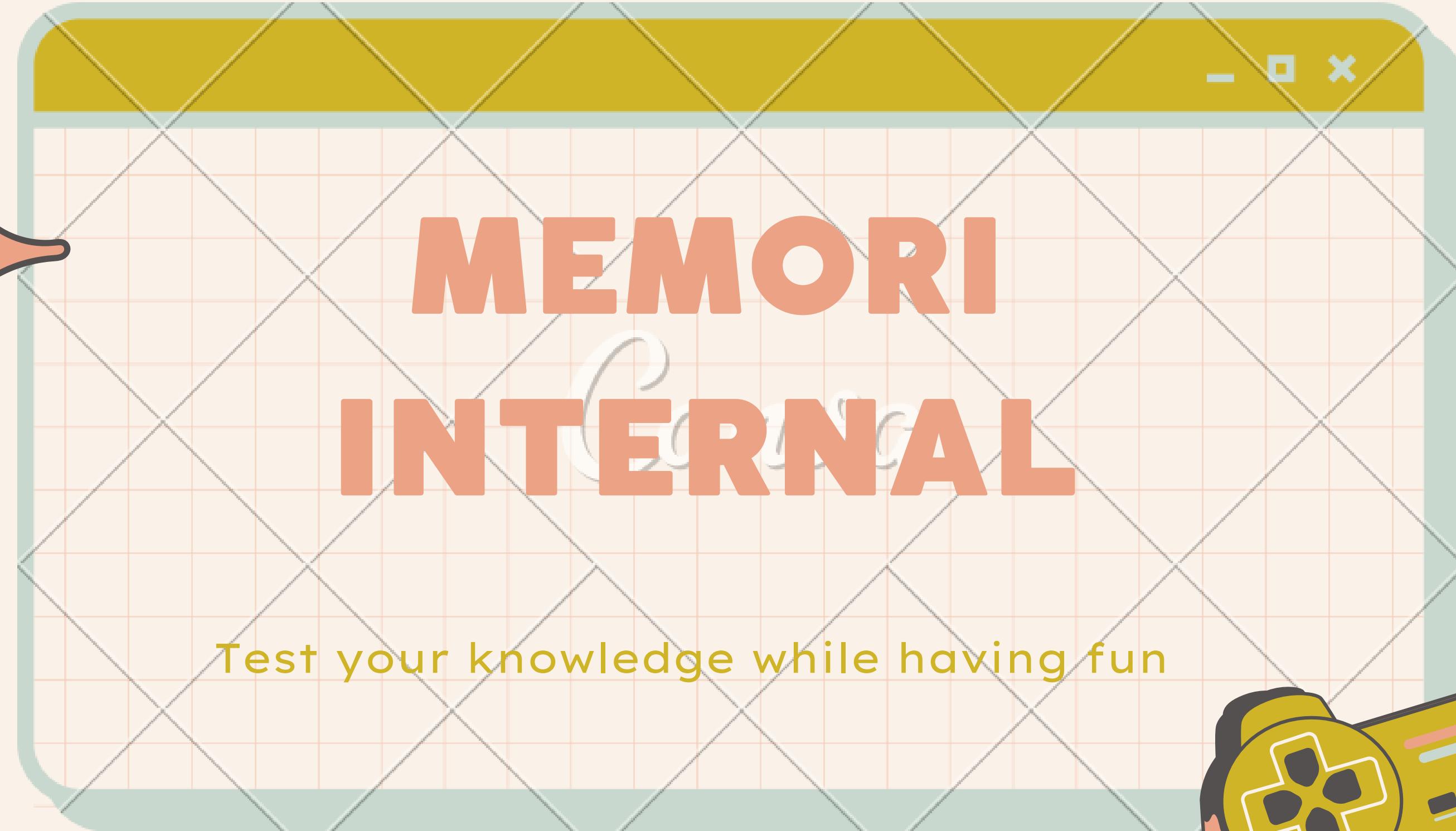
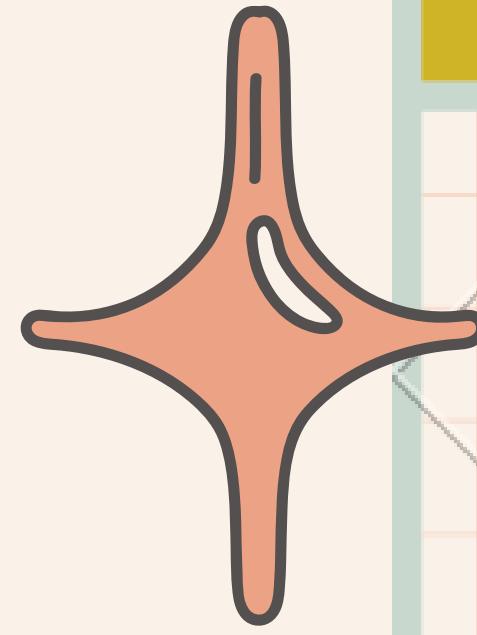
Summary

Chapter 7

- External devices
 - Keyboard/monitor
 - Disk drive
- I/O modules
 - Module function
 - I/O module structure
- Programmed I/O
 - Overview of programmed I/O
 - I/O commands/instructions
- Direct memory access
 - Drawbacks of programmed and interrupt-driven I/O
 - DMA function
 - Intel 8237A DMA controller

Input/Output

- Interrupt-driven I/O
 - Interrupt processing
 - Design issues
 - Intel 82C59A interrupt controller
 - Intel 82C55A programmable peripheral interface
- Direct Cache Access
 - DMA using shared last-level cache
 - Cache-related performance issues
 - Direct cache access strategies
 - Direct data I/O
- I/O channels and processors
 - The evolution of the I/O function
 - Characteristics of I/O channels

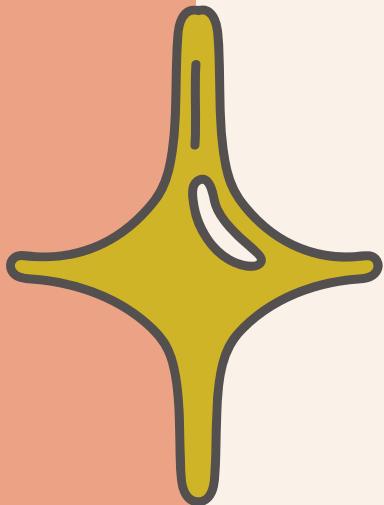




SEMICONDUCTOR MAIN MEMORY



You may start to think
about memory



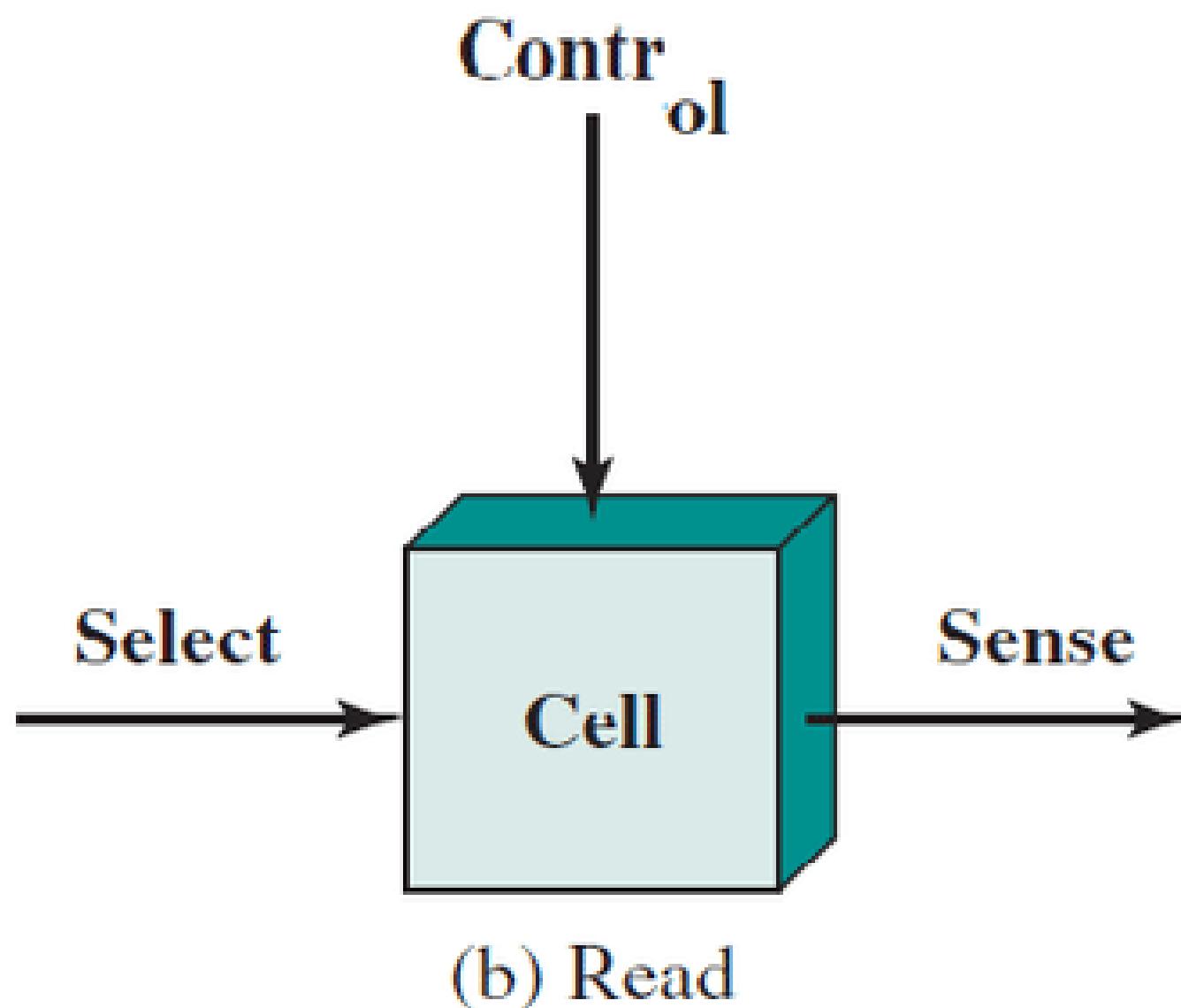
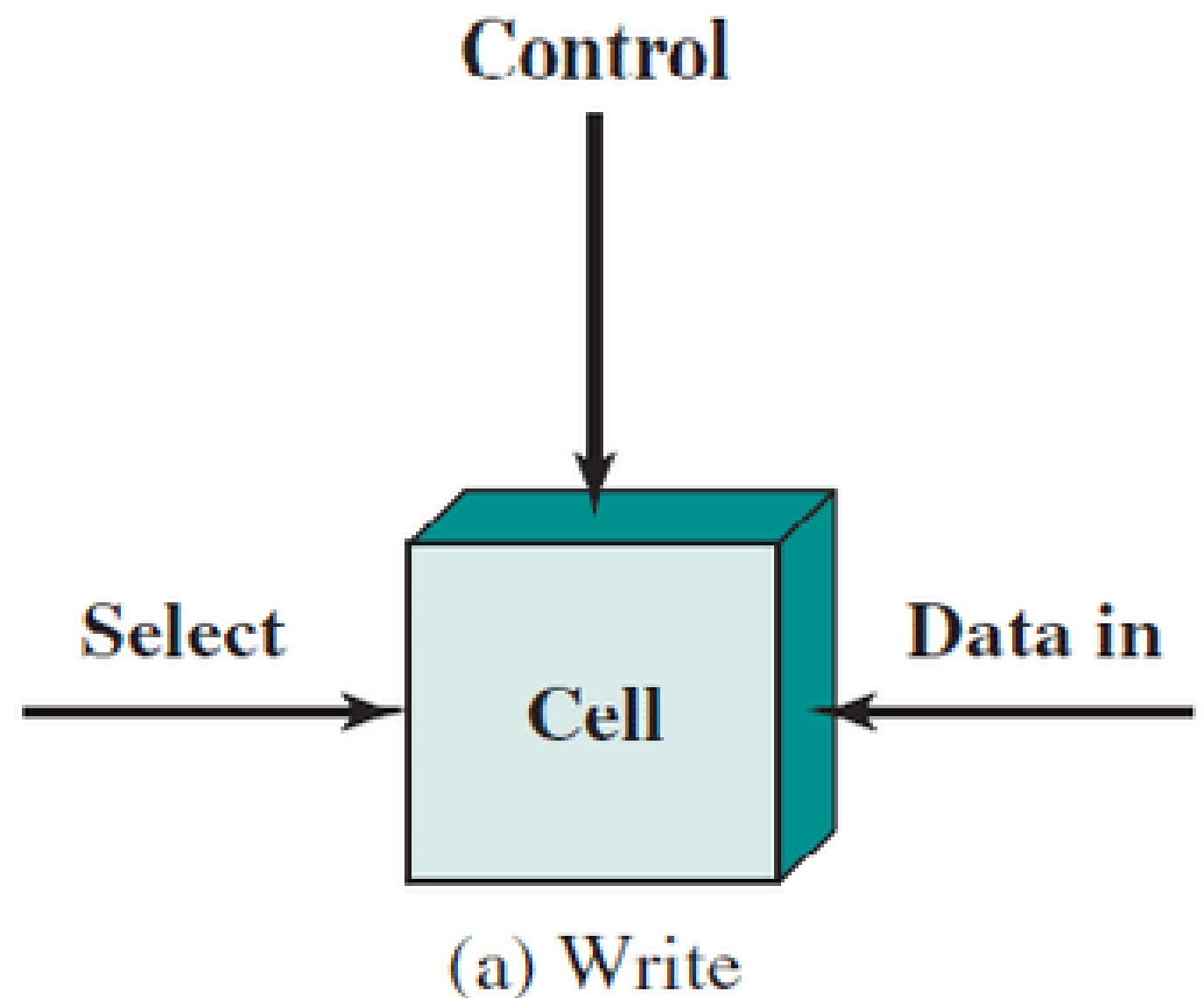


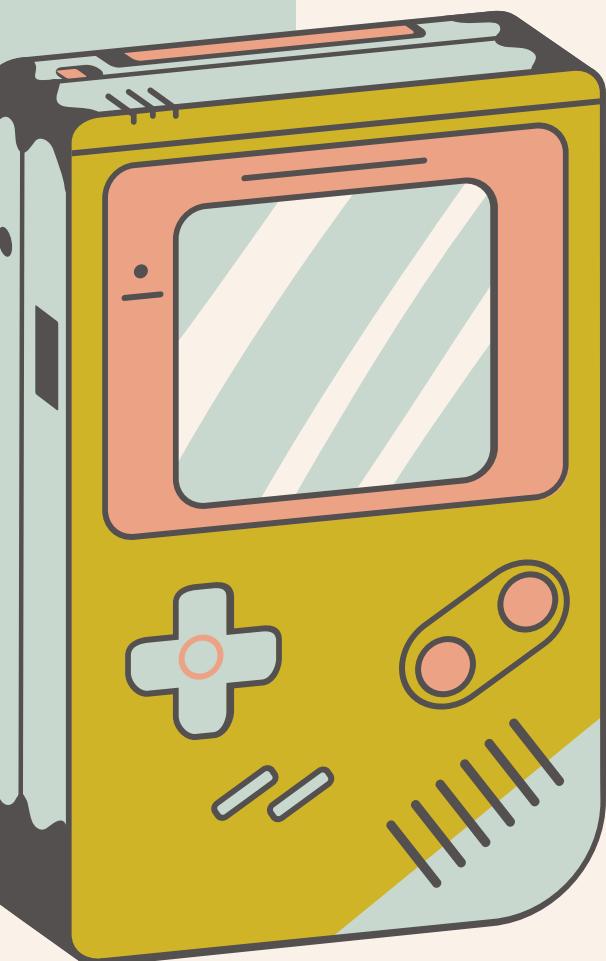
Figure 5.1 Memory Cell Operation

1. Memiliki elemen dasar berupa sel memori/ memory cell
2. Sel memori: memiliki 2 keadaan stabil semi-stabil untuk merepresentasikan biner 0 atau 1
3. Memiliki kemampuan untuk ditulis/menetapkan keadaan
4. Memiliki kemampuan untuk dibaca/membaca keadaan
5. Memiliki 3 terminal: control(r atau w), select dan data in/sense

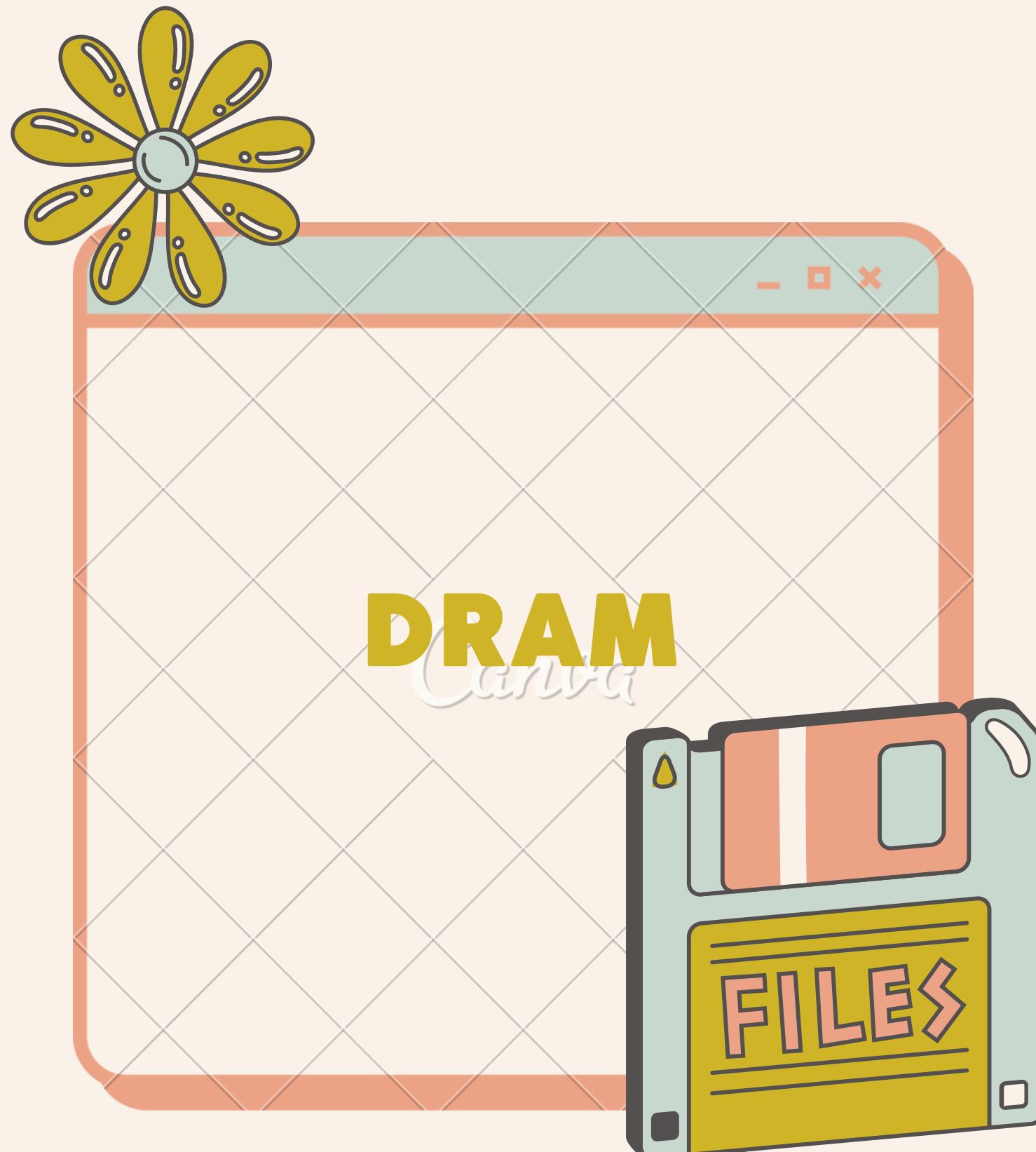


RAM DIBAGI MENJADI:

- Dynamic RAM (DRAM)
- Static RAM (SRAM)

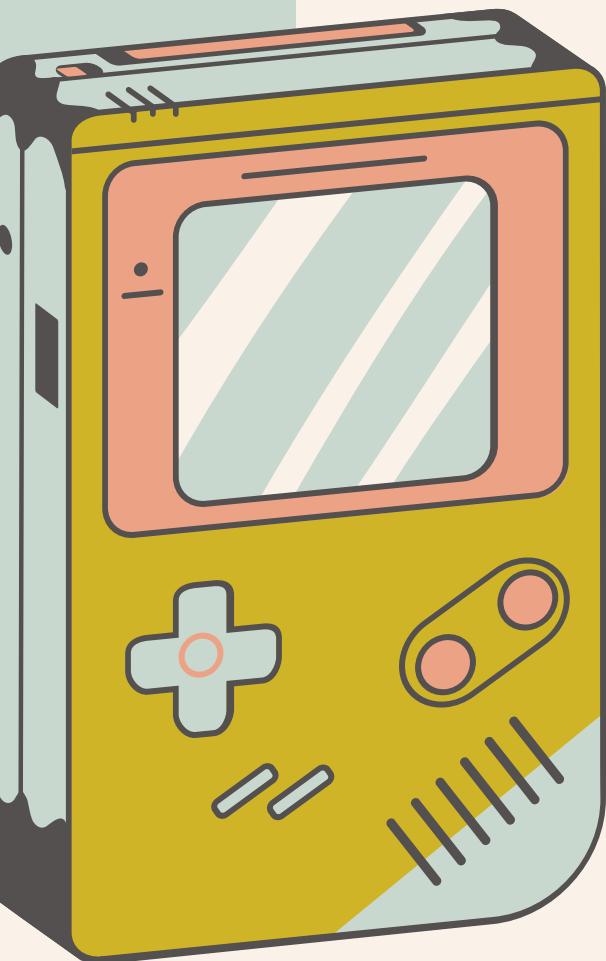


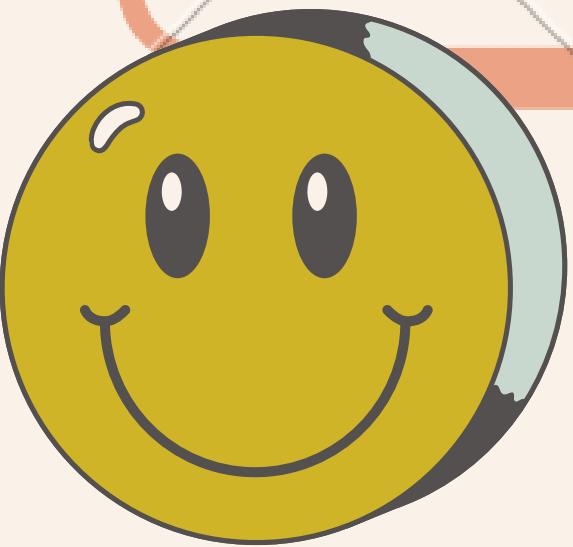
1. Menyimpan setiap bit data dalam sebuah kapasitor.
2. DRAM lebih padat dibandingkan SRAM
3. Ada atau tidak adanya muatan dalam kapasitor ditafsirkan dengan biner 1 atau 0.
4. Membutuhkan penyegaran muatan berkala, untuk menjaga penyimpanan data.
5. DRAM hanya dapat menyimpan data apabila ada tenaga/power yang diberikan.
6. Ketika tidak ada tenaga maka data yang disimpan juga akan hilang



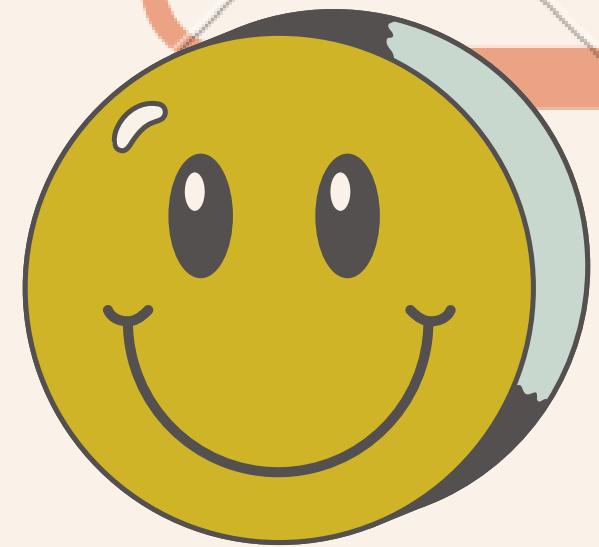
SRAM

- Selama dialiri power listrik data akan tetap utuh, beda dengan DRAM yang secara periodik harus direfresh.
- Berharga mahal karena super cepat dalam transfer data.
- Menggunakan transistor tanpa kapasitor, sehingga tidak ada daya yang bocor.
- Elemen logika yang digunakan dalam prosesor
- Nilai biner disimpan menggunakan konfigurasi gerbang logika flip-flop tradisional





- 1) Berisi pola data permanen yang tidak dapat diubah atau ditambahkan
- 2) Tidak ada sumber daya yang diperlukan untuk mempertahankan nilai bit di memori
- 3) Data atau program secara permanen ada dalam memori utama dan tidak perlu dimuat dari perangkat penyimpanan sekunder
- 4) Data sebenarnya ditransfer ke dalam chip sebagai bagian dari proses fabrikasi

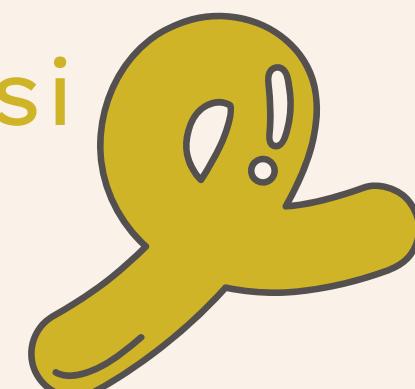


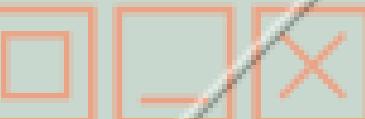
KERUGIAN DARI NO 4

- a) Tidak ada ruang untuk kesalahan, jika satu bit salah maka seluruh batch ROM harus dibuang
- b) Langkah penyisipan data mencakup biaya tetap yang relatif besar

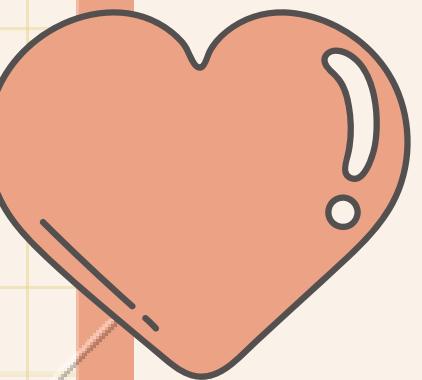
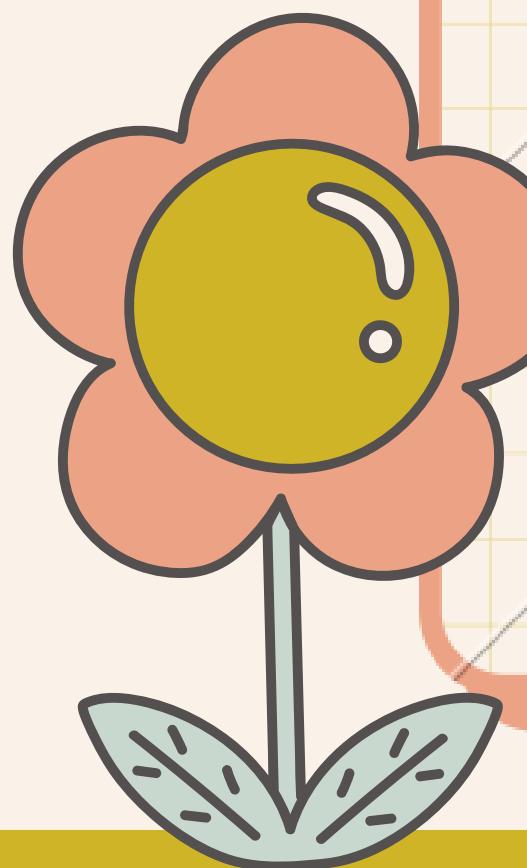
PROGRAMMA ROM WRITER

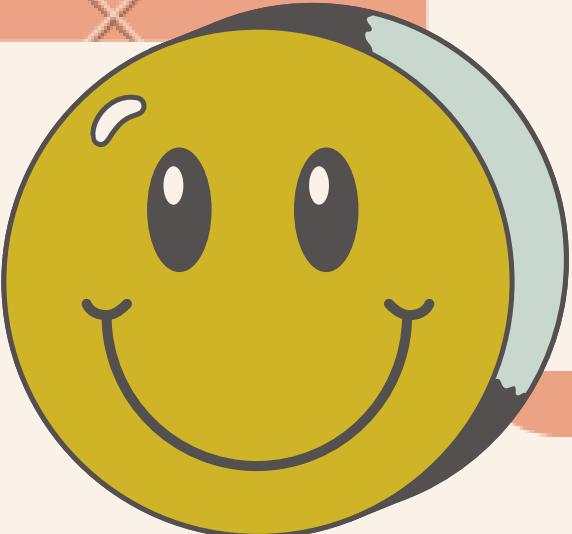
- 1) Alternatif yang lebih murah
- 2) Non-volatile dan dapat ditulis hanya sekali
- 3) Proses penulisan dapat dilakukan secara elektrik dan dapat dilakukan oleh pemasok atau pelanggan pada waktu yang lebih lambat dari chip asli fabrikasi ()
- 4) Diperlukan peralatan khusus untuk proses penulisan
- 5) Memberikan fleksibilitas dan kenyamanan
- 6) Menarik untuk berjalannya produksi volume tinggi





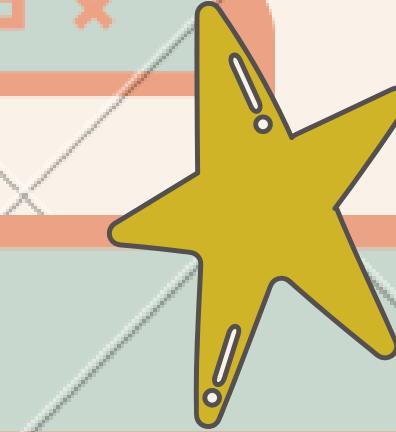
READ-MOSTLY MEMORY

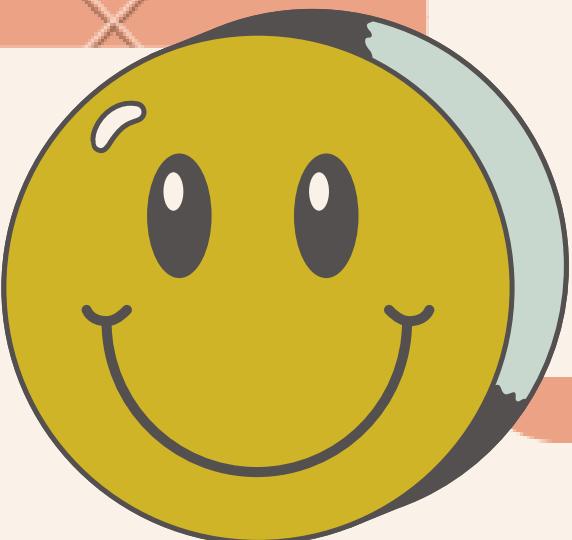




EPROM

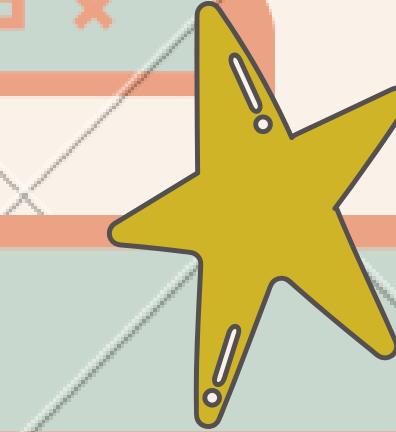
- 1) Erasable programmable read-only memory
- 2) Proses penghapusan dapat dilakukan berulang kali
- 3) Lebih mahal dari PROM tetapi memiliki keuntungan dari kemampuan pembaruan ganda

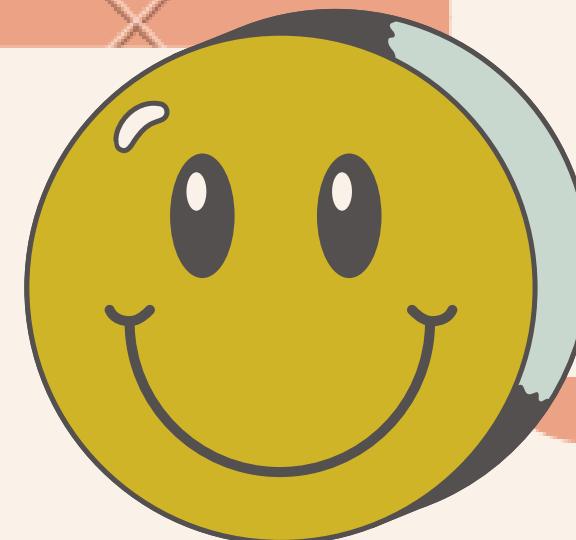




EEPROM

- 1) Electrically erasable programmable read-only memory
- 2) Dapat ditulis setiap waktu tanpa menghapus isi sebelumnya
- 3) Menggabungkan keuntungan dari non-volatilitas dengan fleksibilitas menjadi tempat yang updatable
- 4) Lebih mahal dari EPROM





FLASH MEMORY

- 1) Berada di antara EPROM dan EEPROM pada biaya dan fungsionalitasnya
- 2) Menggunakan an electrical erasing technology, tidak menyediakan byte-level erasure
- 3) Microchip is organized so that a section of memory cells are erased in a sing



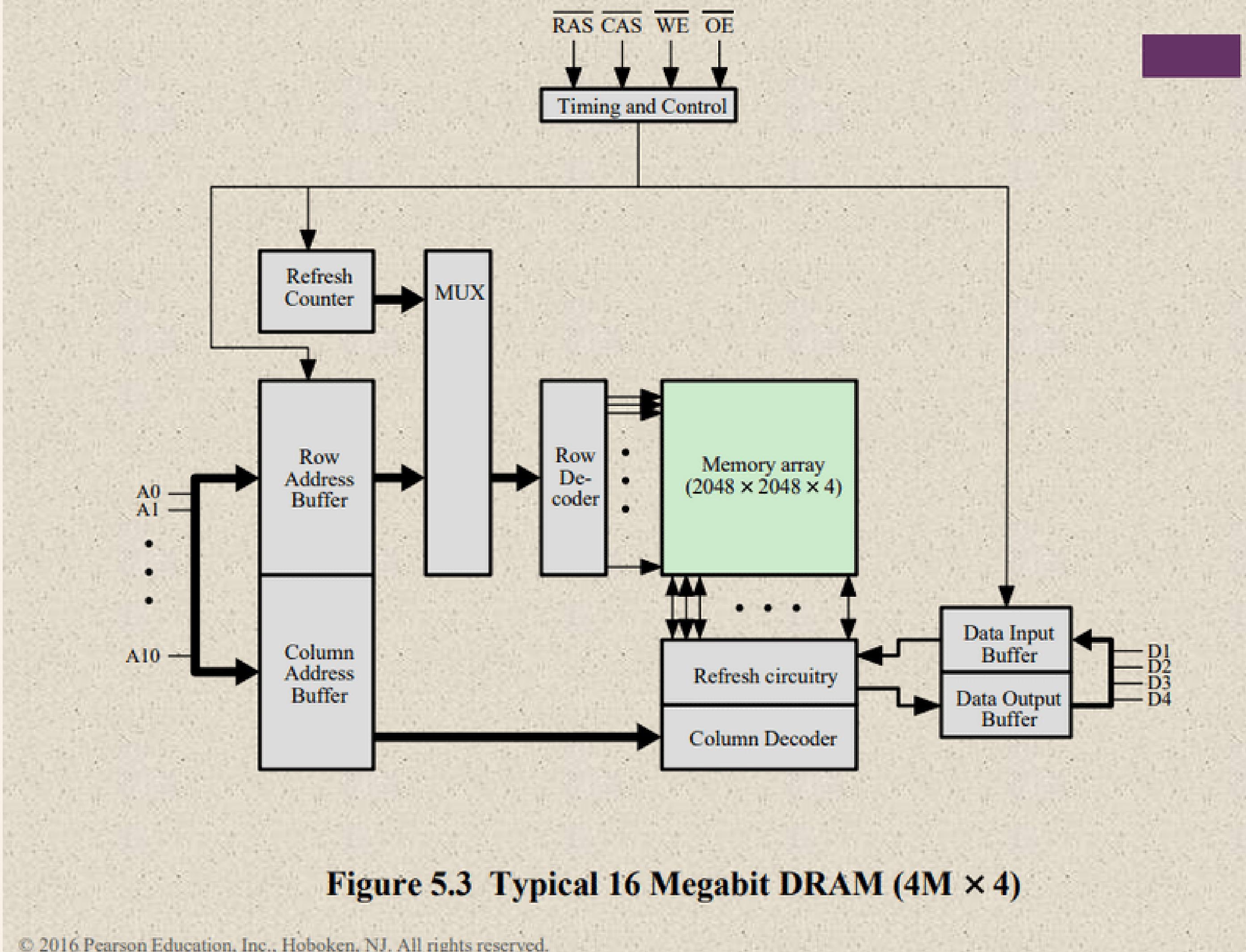


Figure 5.3 Typical 16 Megabit DRAM (4M × 4)

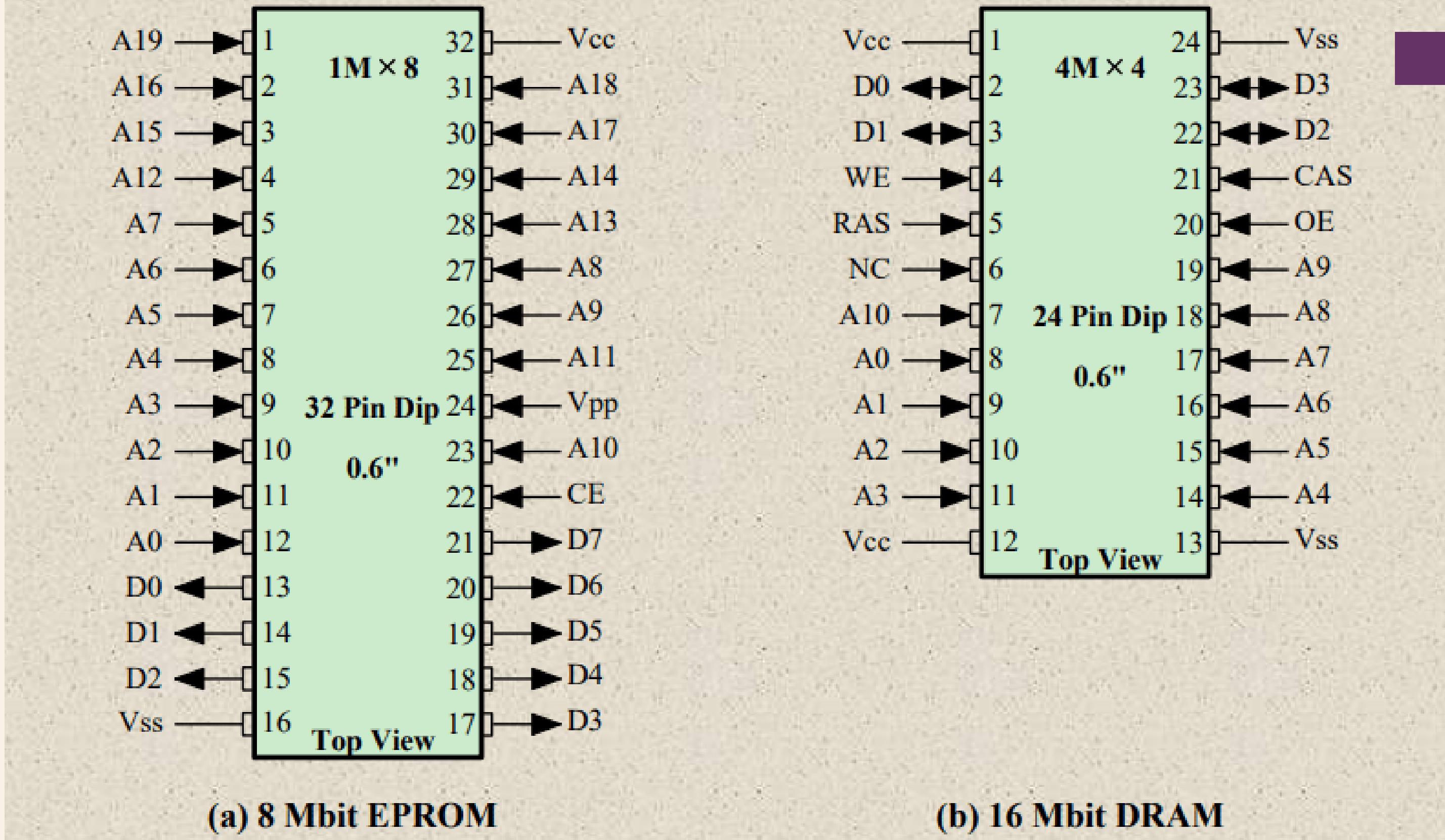


Figure 5.4 Typical Memory Package Pins and Signals

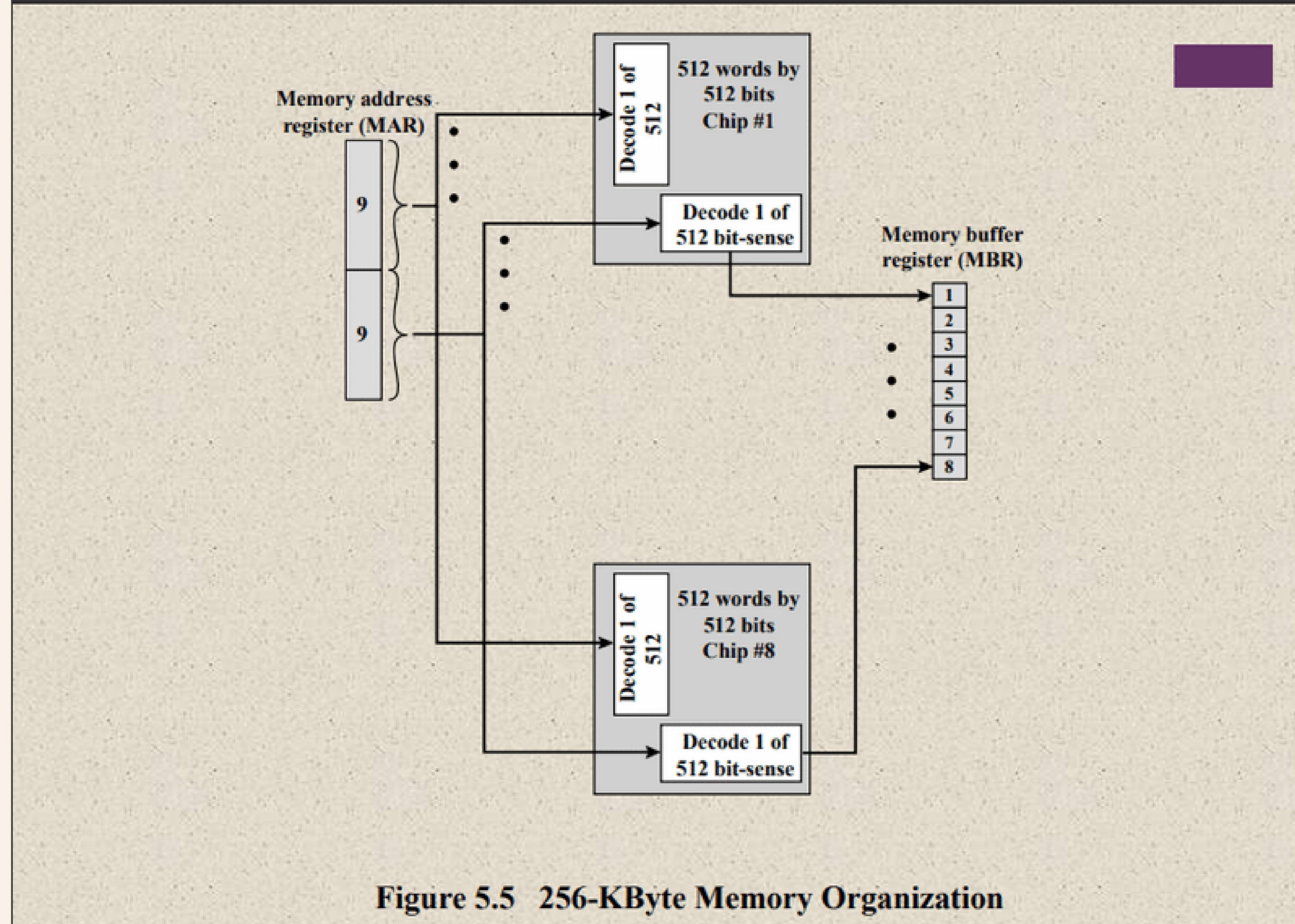


Figure 5.5 256-KByte Memory Organization

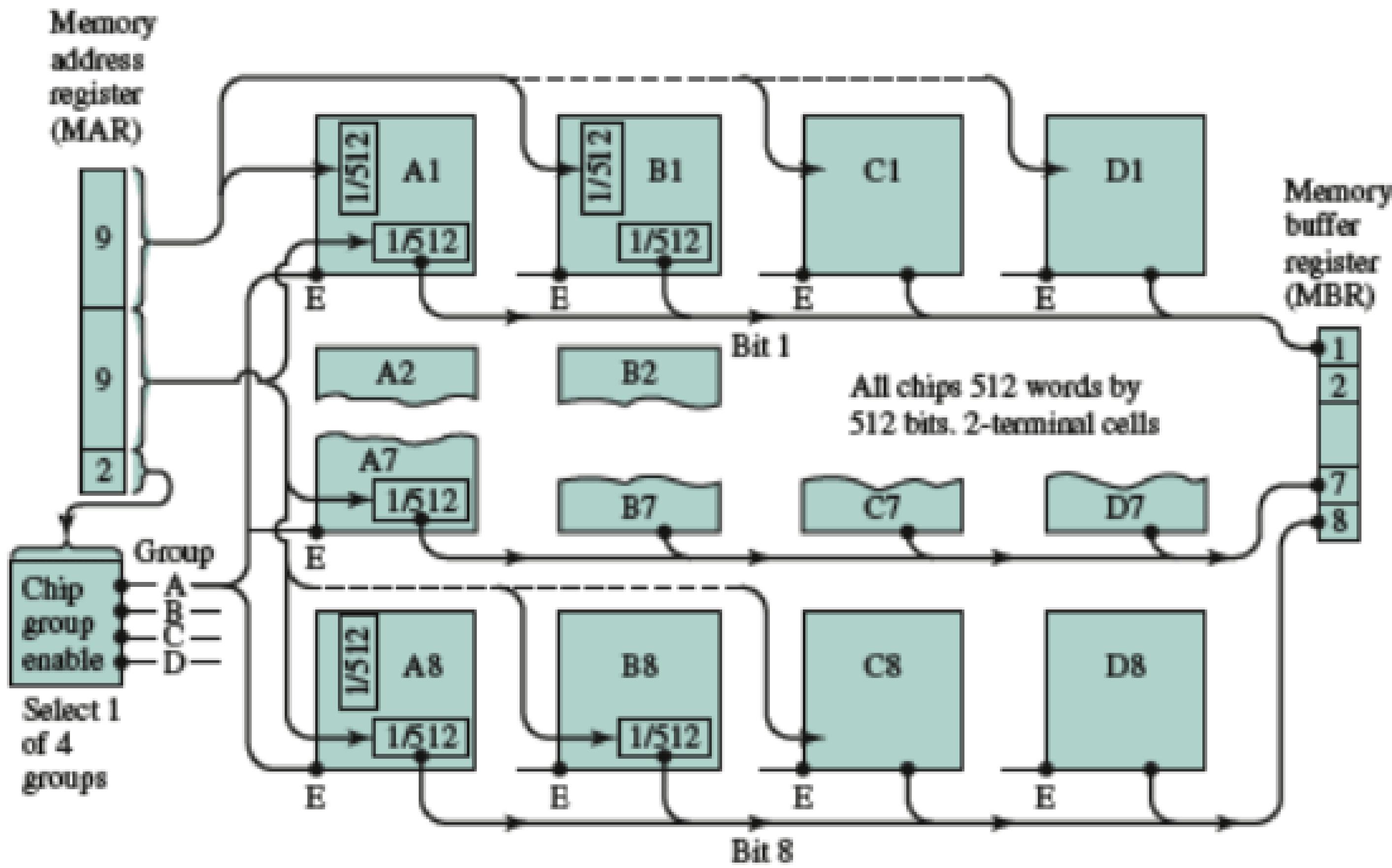
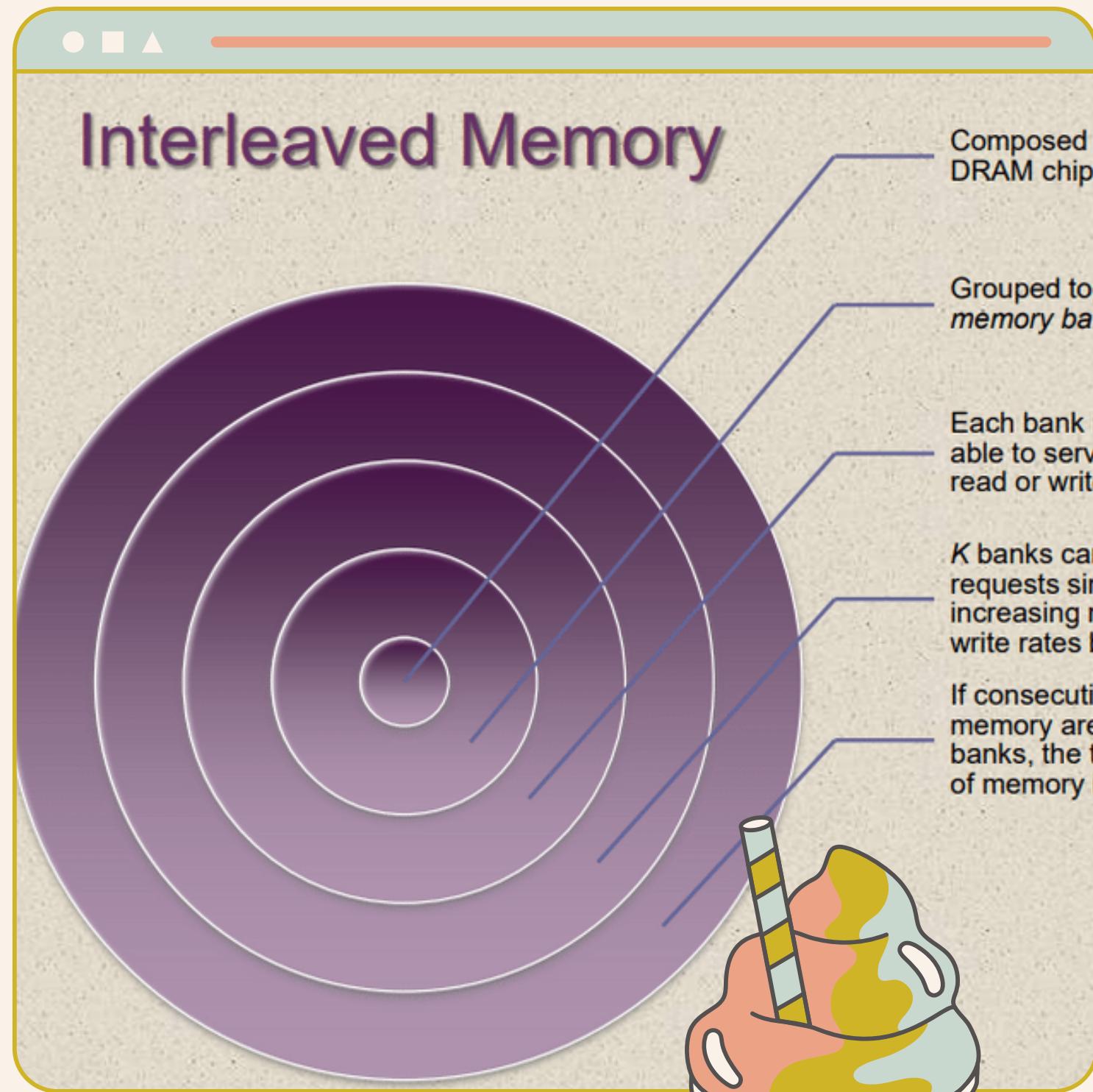


Figure 5.6 1-Mbyte Memory Organization

INTERLEAVED MEMORY

- 1) Composed of a collection of DRAM chips
- 2) Dikelompokkan bersama untuk membentuk memory bank
- 3) Setiap bank secara independen Mampu melayani memori membaca atau menulis permintaan
- 4) Bank K dapat melayani K permintaan secara bersamaan, meningkatkan memori membaca atau menulis tarif dengan faktor K
- 5) Jika kata-kata berturut-turut dari memori disimpan dalam berbagai bank, transfer blok memori dipercepat



DDR DRAM

Maya Amelia

G6401201045

ORGANISASI DRAM LANJUTAN

- Salah satu hambatan sistem saat menggunakan prosesor berperforma tinggi yaitu interface ke main internal memory
- Sejumlah peningkatan pun terjadi pada arsitektur DRAM dasar yang telah dieksplorasi
- skema yg mendominasi di pasar adalah SDRAM dan DDR-DRAM

SDRAM

DDR-DRAM

RDRAM

Synchronous DRAM (SDRAM)

Salah satu bentuk DRAM yang paling banyak digunakan

Type RAM ini dibuat sekitar tahun 1996

Pertukaran data dengan prosesor yang disinkronkan ke sinyal clock eksternal dan berjalan pada kecepatan penuh bus prosesor/memori tanpa memaksakan untuk menunggu



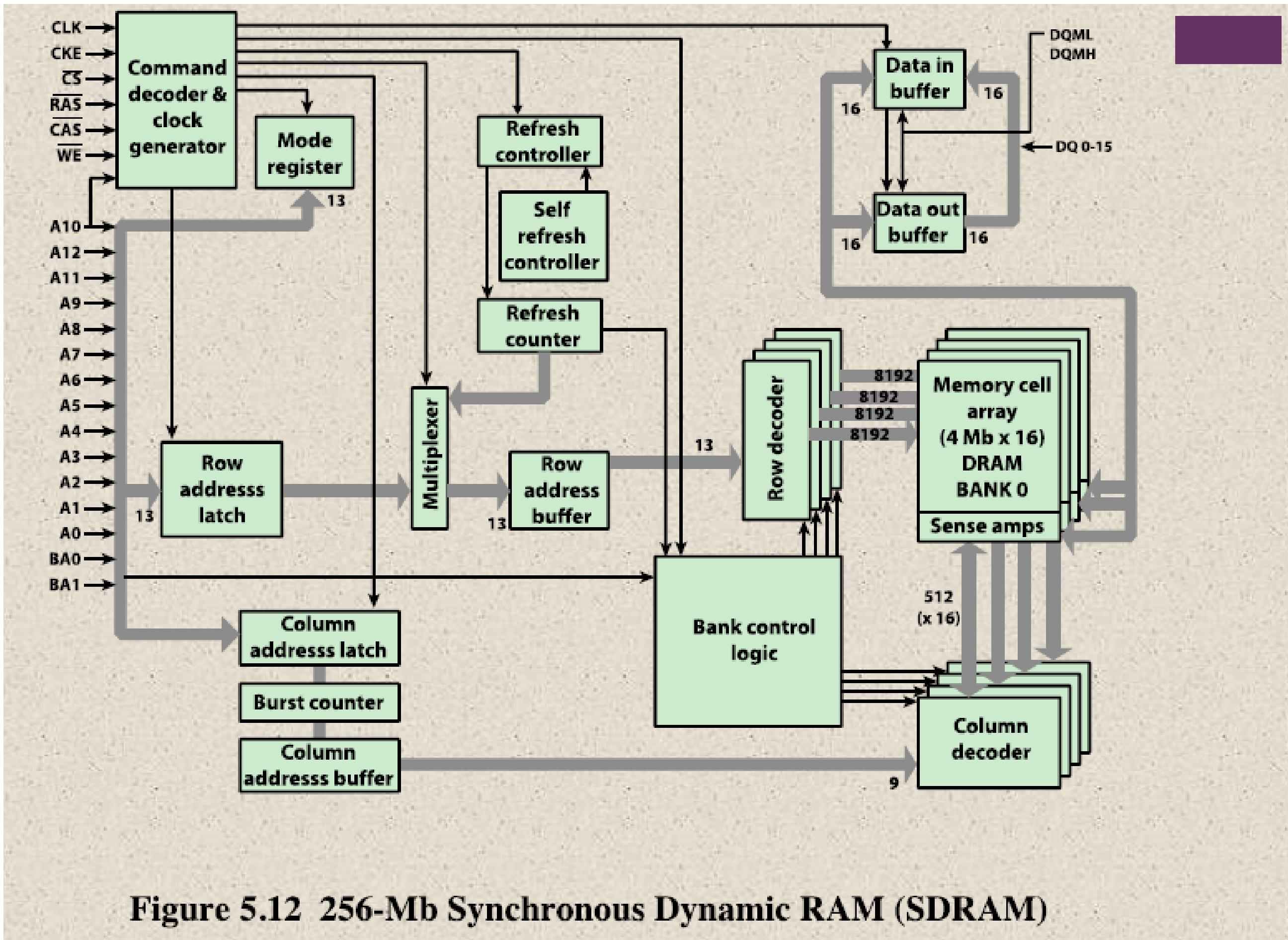


Figure 5.12 256-Mb Synchronous Dynamic RAM (SDRAM)

Table 5.3
SDRAM
Pin
Assignments

A0 to A12	Address inputs
BA0, BA1	Bank address lines
CLK	Clock input
CKE	Clock enable
$\overline{\text{CS}}$	Chip select
$\overline{\text{RAS}}$	Row address strobe
$\overline{\text{CAS}}$	Column address strobe
$\overline{\text{WE}}$	Write enable
DQ0 to DQ15	Data input/output
DQM	Data mask

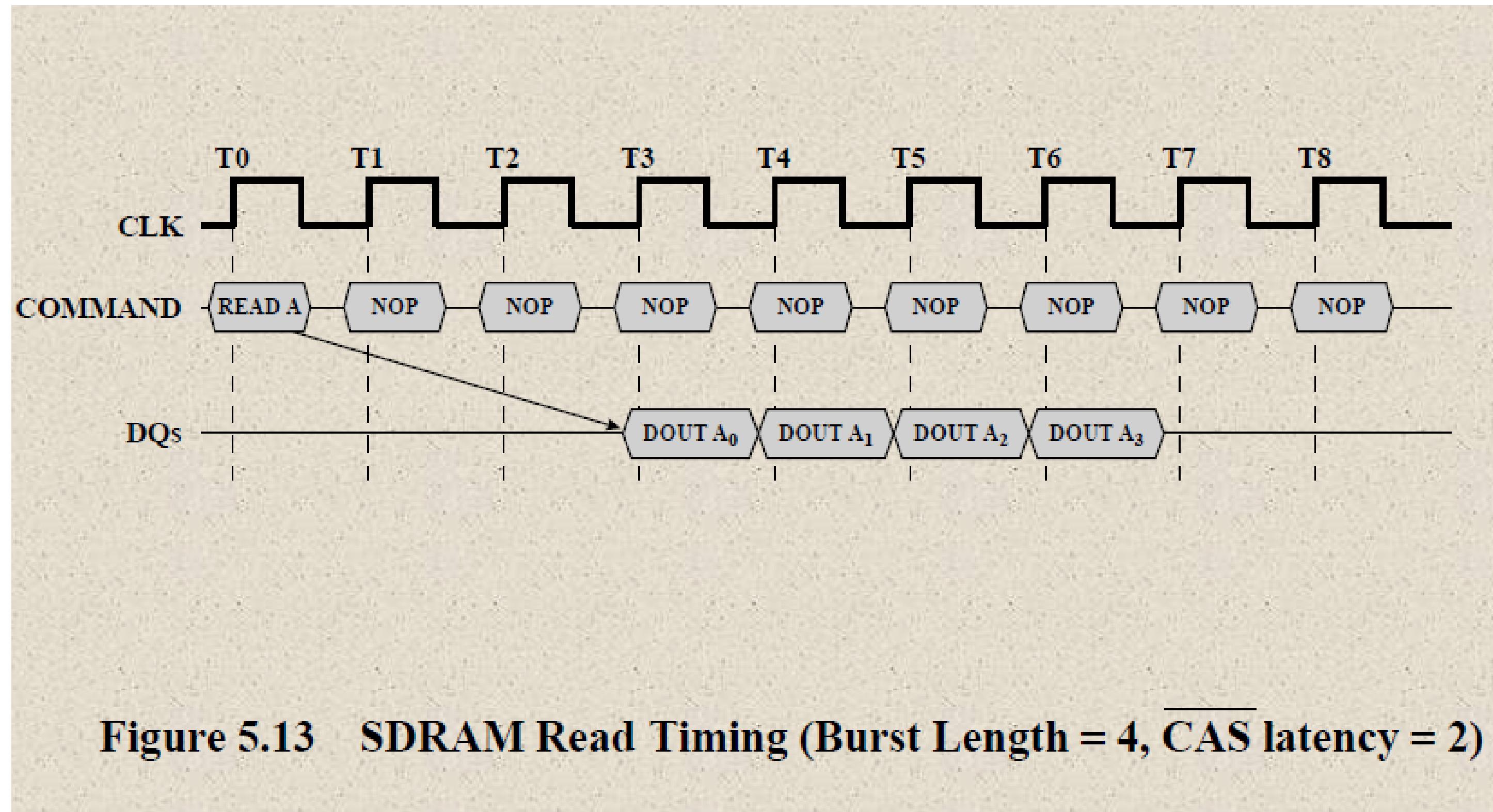


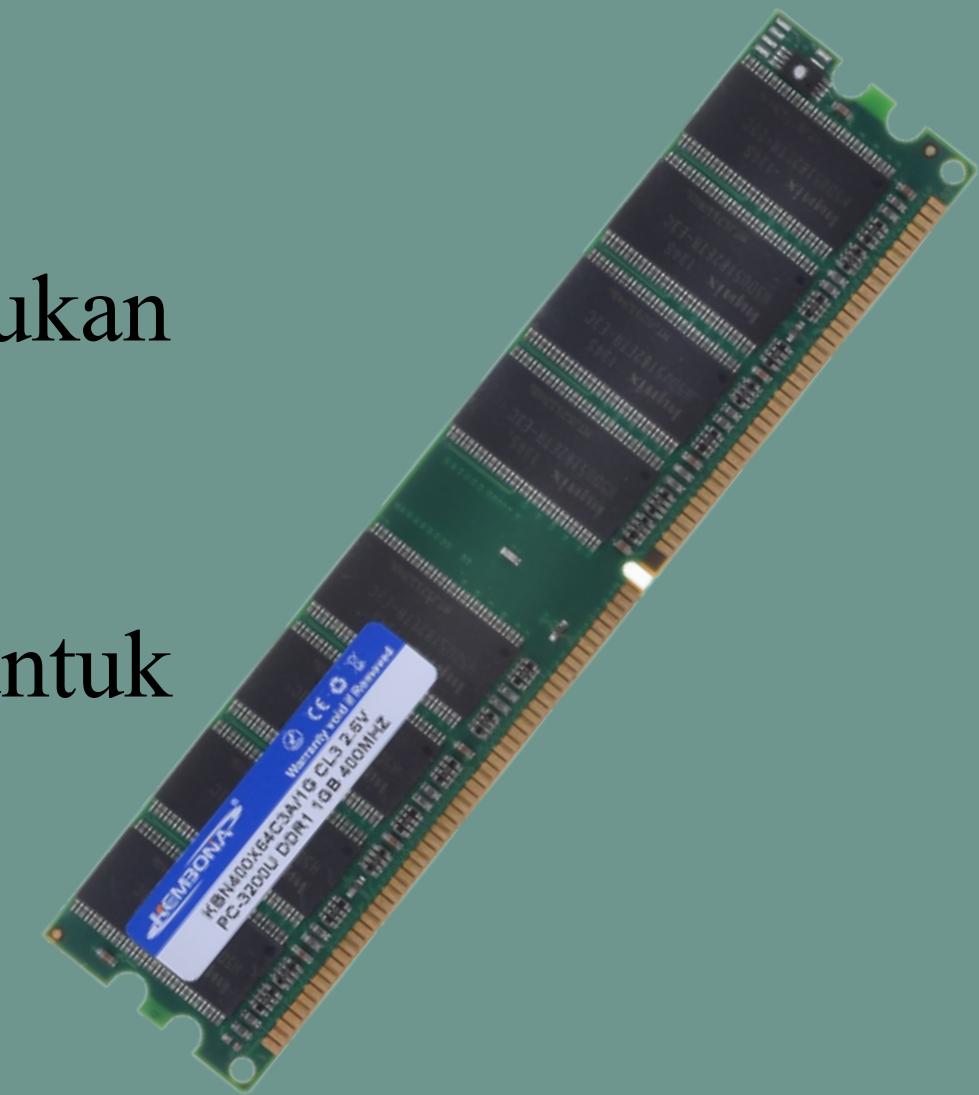
Figure 5.13 SDRAM Read Timing (Burst Length = 4, $\overline{\text{CAS}}$ latency = 2)

Double Data Rate SDRAM (DDR SDRAM)

Dikembangkan oleh JEDEC Solid State Technology Association perusahaan membuat chip DDR, yang banyak digunakan di komputer desktop dan server

DDR mencapai kecepatan data yang lebih tinggi dalam tiga cara:

1. transfer data disinkronkan ke tepi naik dan turun clock, bukan hanya tepi naik
2. DDR menggunakan kecepatan clock yang lebih tinggi di bus untuk meningkatkan kecepatan transfer
3. skema buffering digunakan



	DDR1	DDR2	DDR3	DDR4
Ambil buffer (bit)	2	4	8	8
Tingkat tegangan (V)	2.5	1.8	1.5	1.2
Bus sisi depan kecepatan data (Mbps)	200–400	400–1066	800–2133	2133–4266

Tabel 5.4
Karakteristik DDR

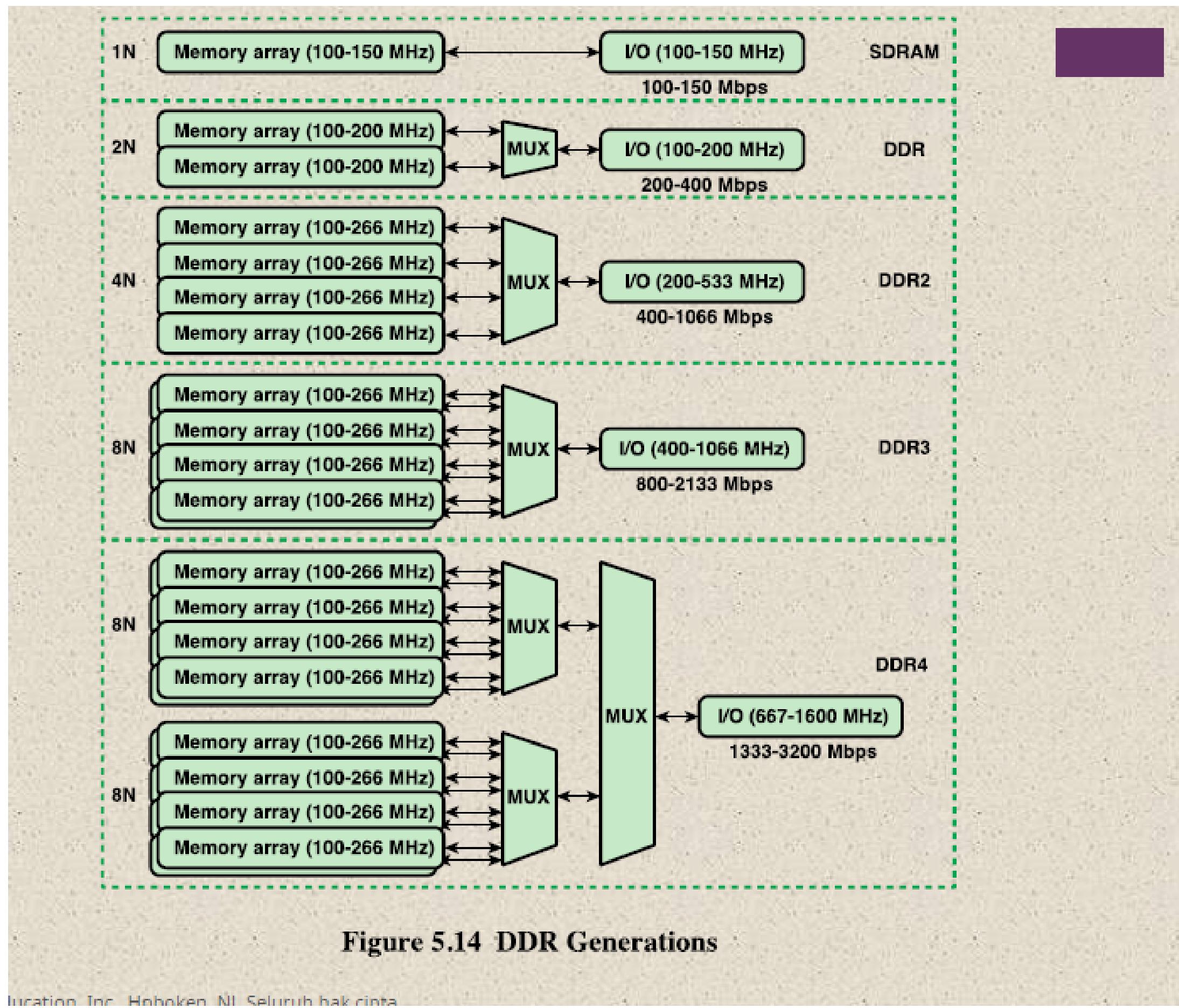
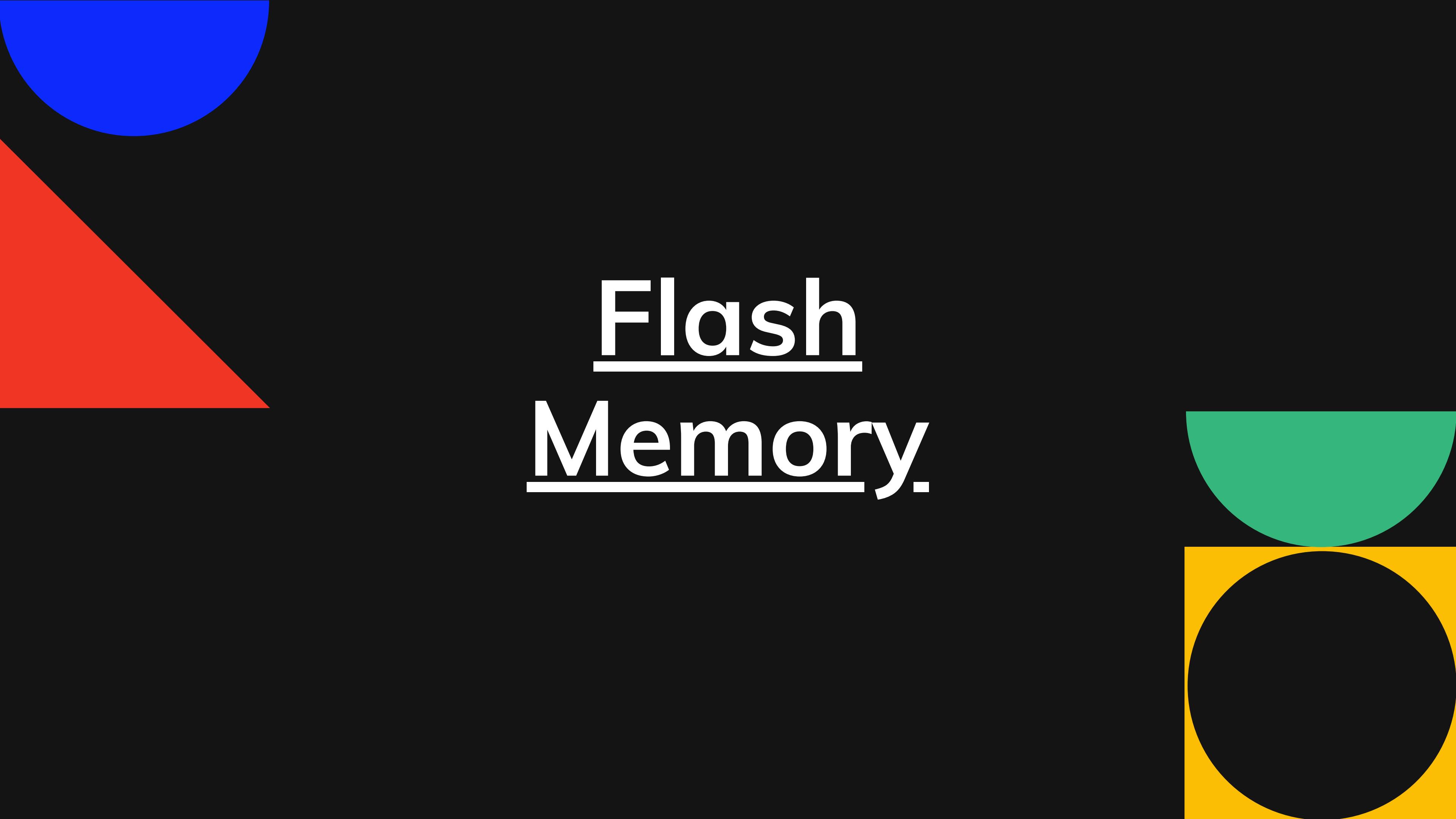


Figure 5.14 DDR Generations

RDRAM (Rambus Dynamic RAM)

- Type RAM ini dibuat sekitar tahun 1999
- Merupakan RAM menggunakan teknologi baru yang dikembangkan oleh perusahaan bernama Rambus.
- Mempunyai kemampuan bandwidth yang menyamai kebutuhan bandwidth pada processor intel pentium 4.
- Tipe pengolahan serial dibanding SDRAM & DDR RAM yang mengolah secara paralel.



Flash
Memory.

Flash Memory

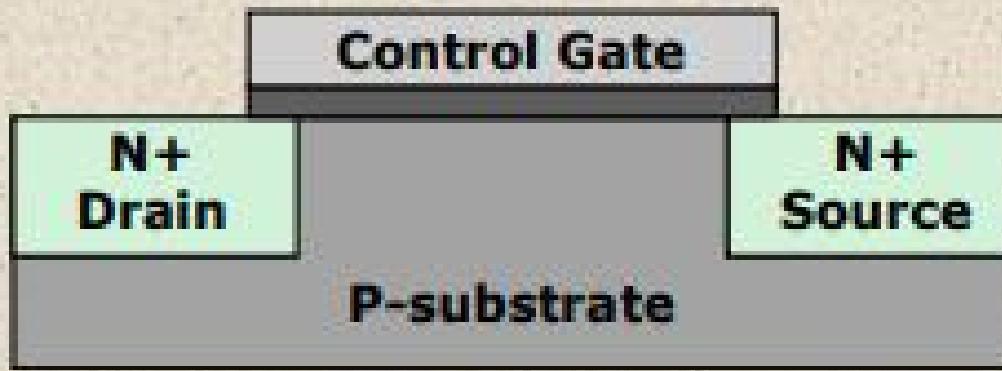
Pertama kali digunakan pada pertengahan 1980

Digunakan pada memori internal dan eksternal

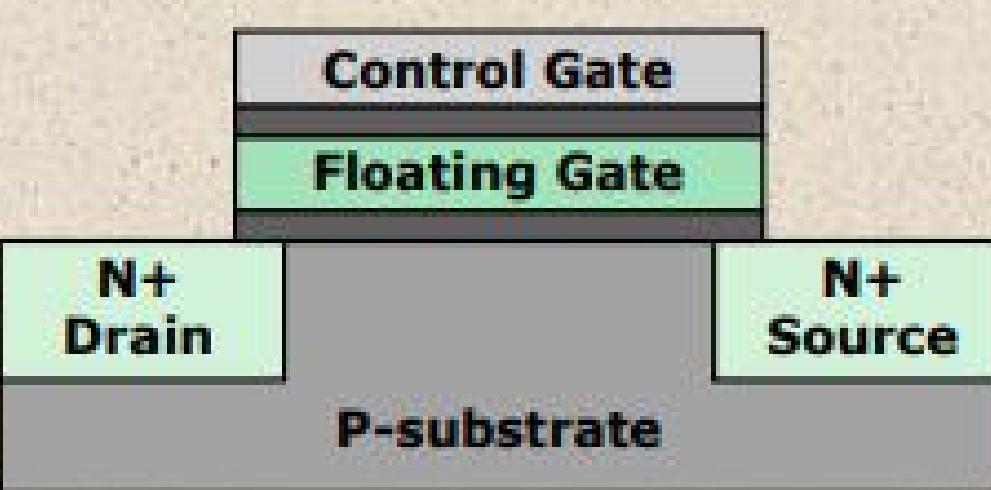
Flash memory berada ditengah EPROM dan EEPROM dalam segi biaya dan fungsionalnya

Selain itu menggunakan teknologi electrical erasing seperti EEPROM, sehingga dapat menghapus bagian dari block memori.

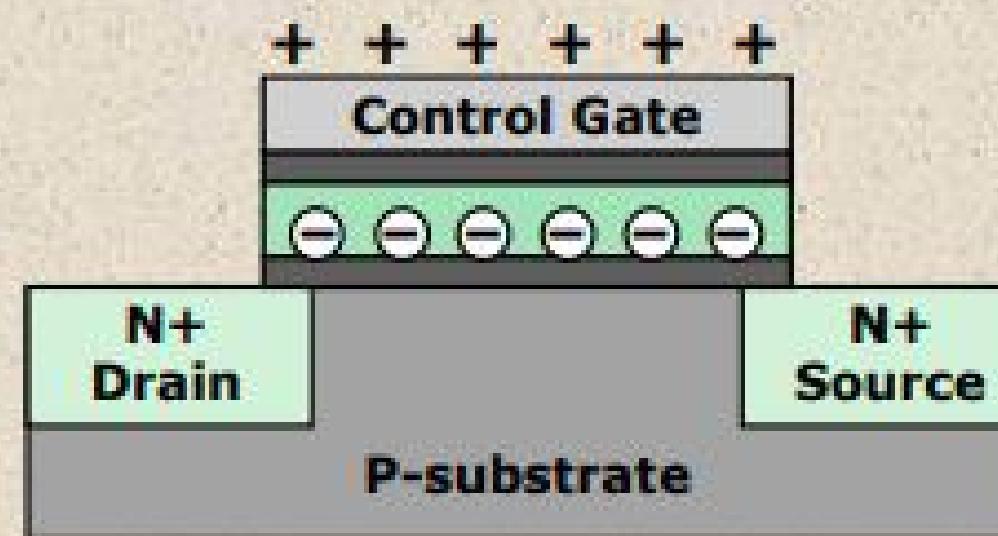
Dinamakan Flash Memori karena penggunaan microchip yang diatur agar dapat menghapus bagian dari memori cell dalam satu kali tindakan



(a) Transistor structure



(b) Flash memory cell in one state



(c) Flash memory cell in zero state

Figure 5.15 Flash Memory Operation

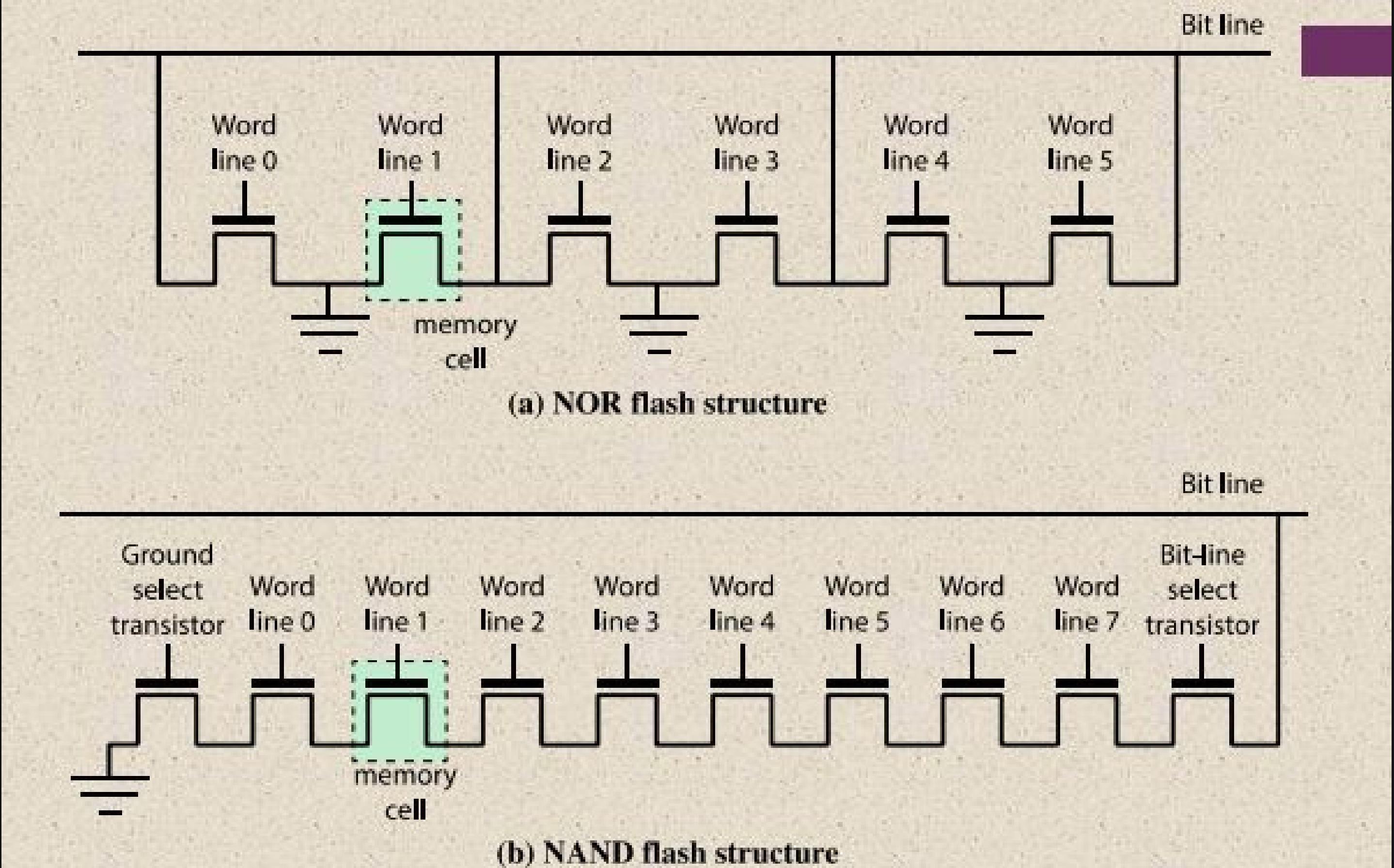
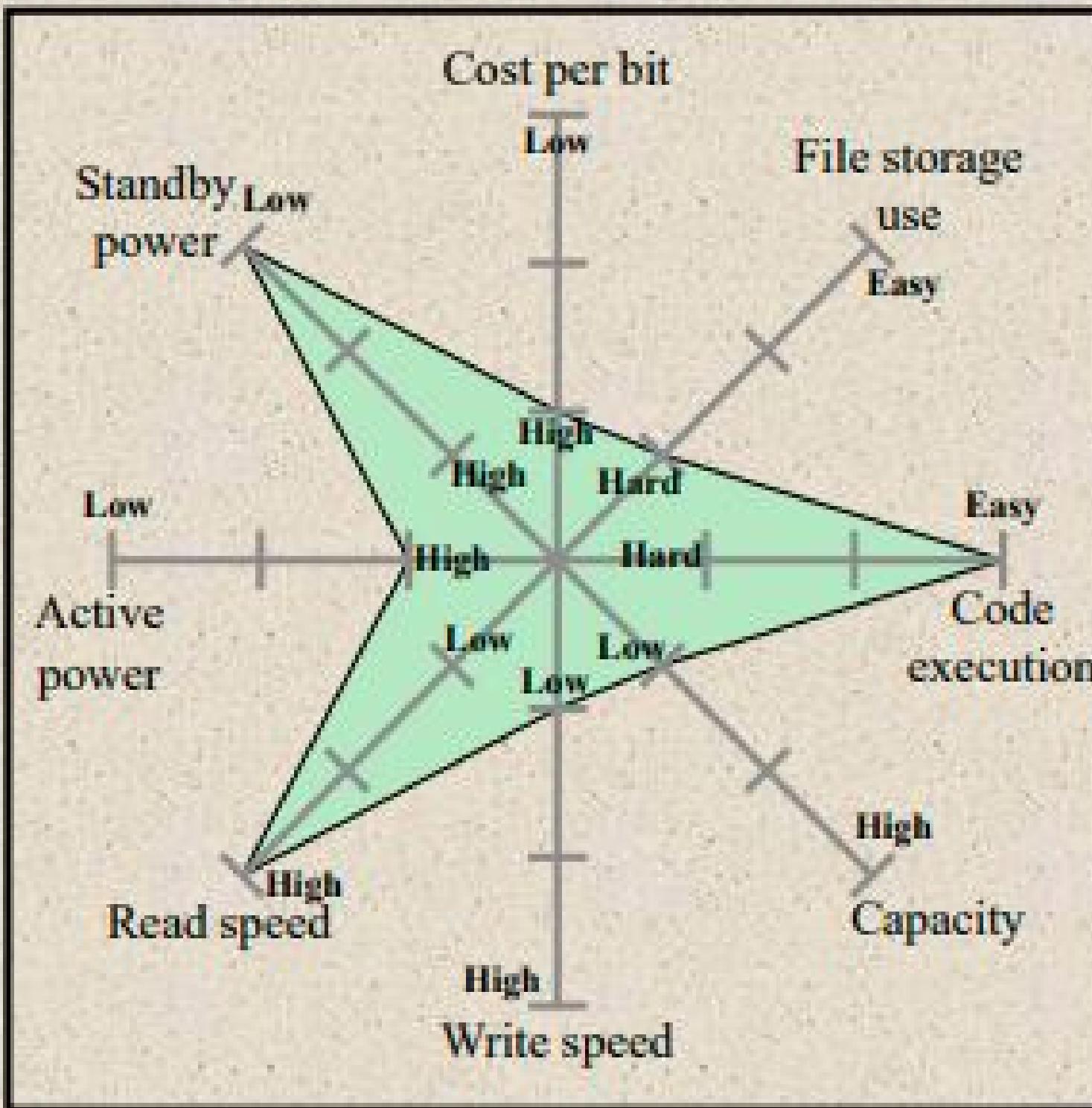
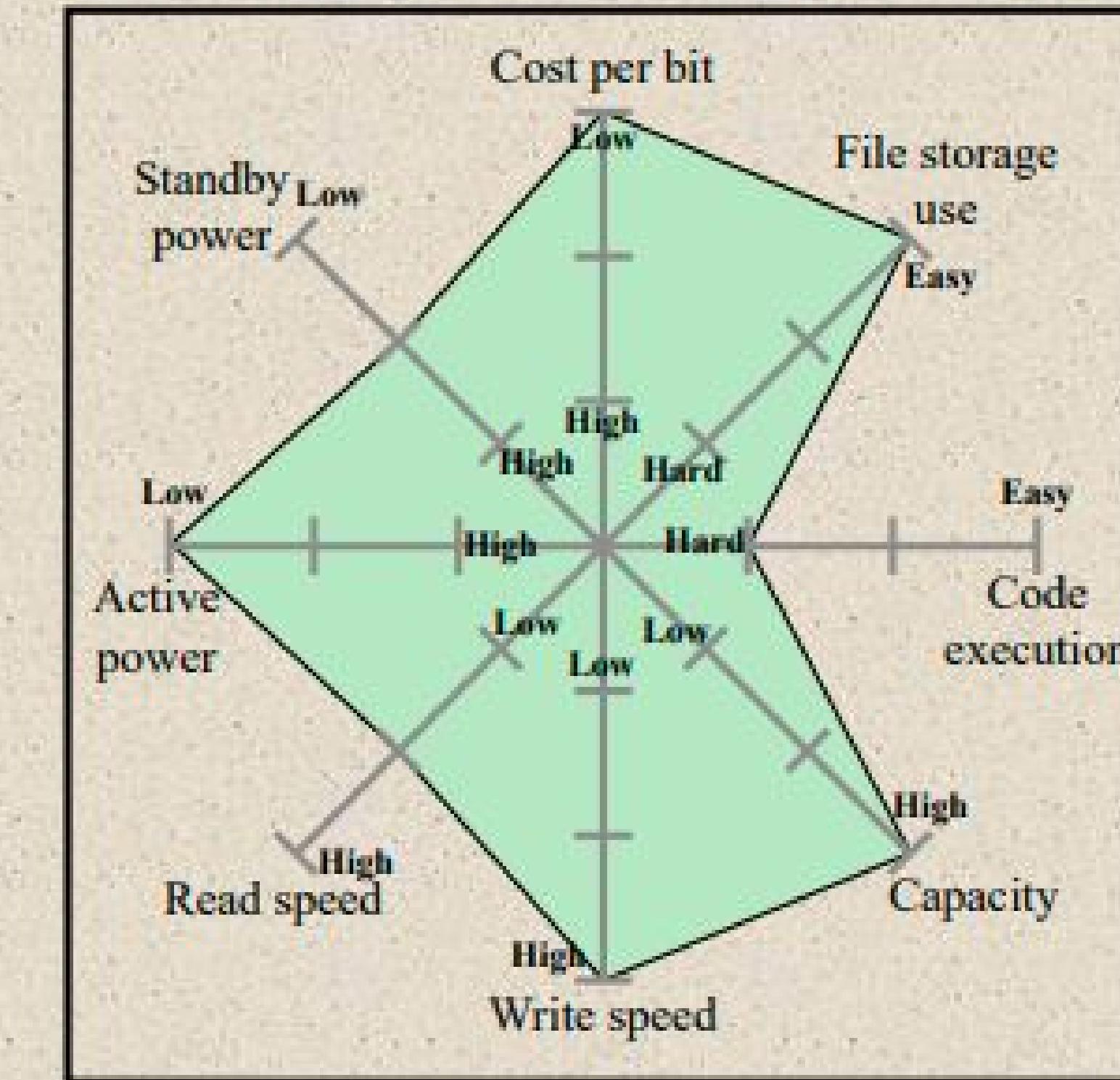


Figure 5.16 Flash Memory Structures



(a) NOR



(b) NAND

Figure 5.17 Kiviat Graphs for Flash Memory

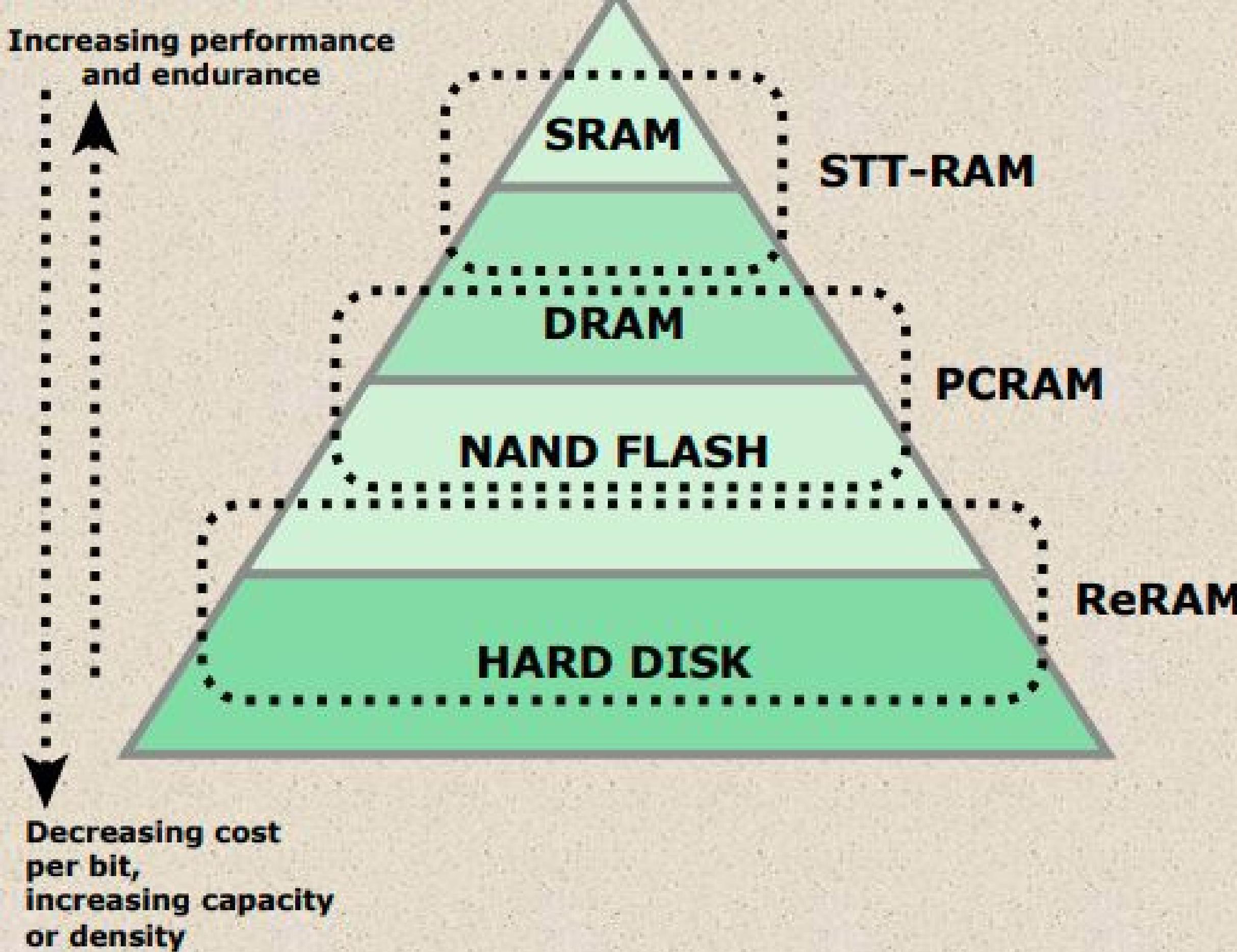


Figure 5.18 Nonvolatile RAM within the Memory Hierarchy