210245: Digital Electronics and Logic Design

Unit VI

Introduction to Computer Architecture

- Introduction to Ideal Microprocessor
- Data Bus
- Address Bus
- Control Bus

Introduction to Ideal Microprocessor

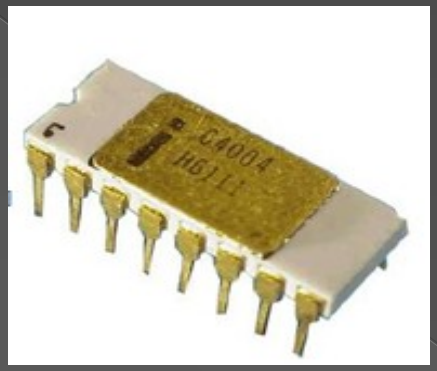
- Basics of Microprocessor
- Fairchild Semiconductor(1957)
- Founder of Intel(Robert,Gordan,Andrew)
- 3 man start up 1968 to industrial gaint by 1981.

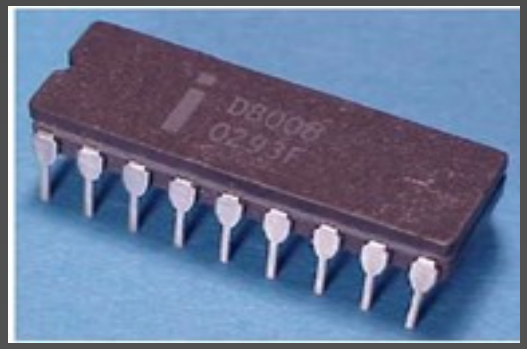
1970s Processors

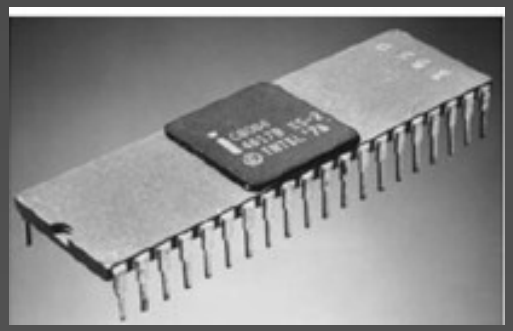
	4004	8008	8080	8086	8088
Introduced	1971	1972	1974	1978	1979
Clock speeds	108 kHz	108 kHz	2 MHz	5 MHz, 8 MHz, 10 MHz	5 MHz, 8 MHz
Bus width	4 bits	8 bits	8 bits	16 bits	8 bits
Number of transistors	2,300	3,500	6,000	29,000	29,000
Feature size (µm)	10		6	3	6
Addressable memory	640 Bytes	16 KB	64 KB	1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

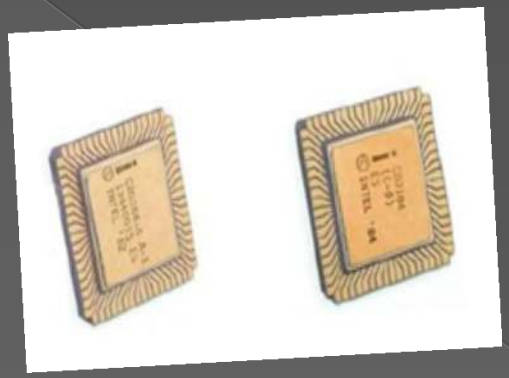
1980s Processors

	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

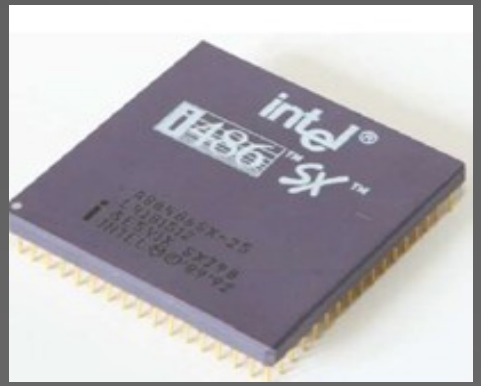

C4004
MCS11


i
D8008
0293F





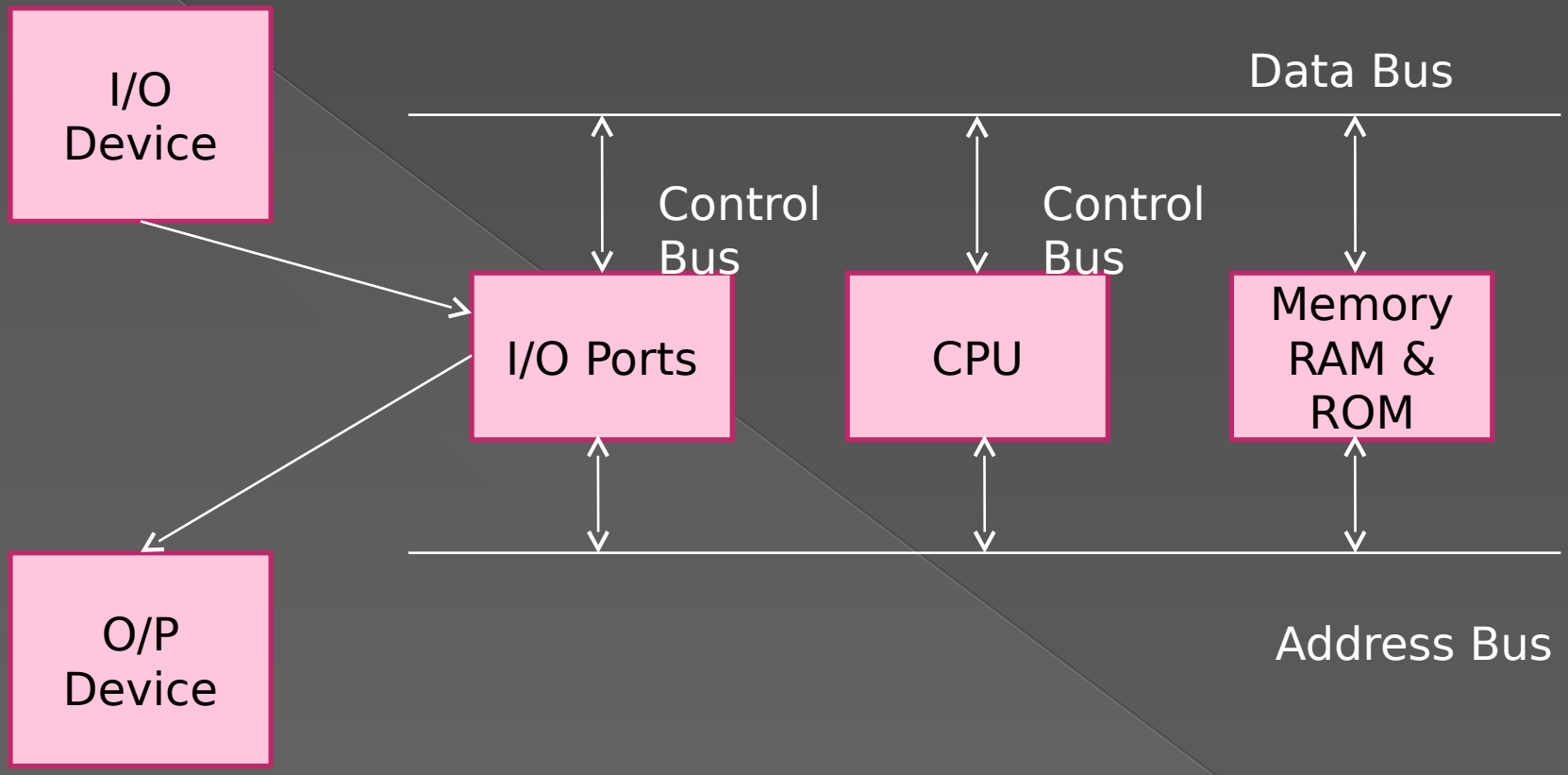

intel
i386 SL

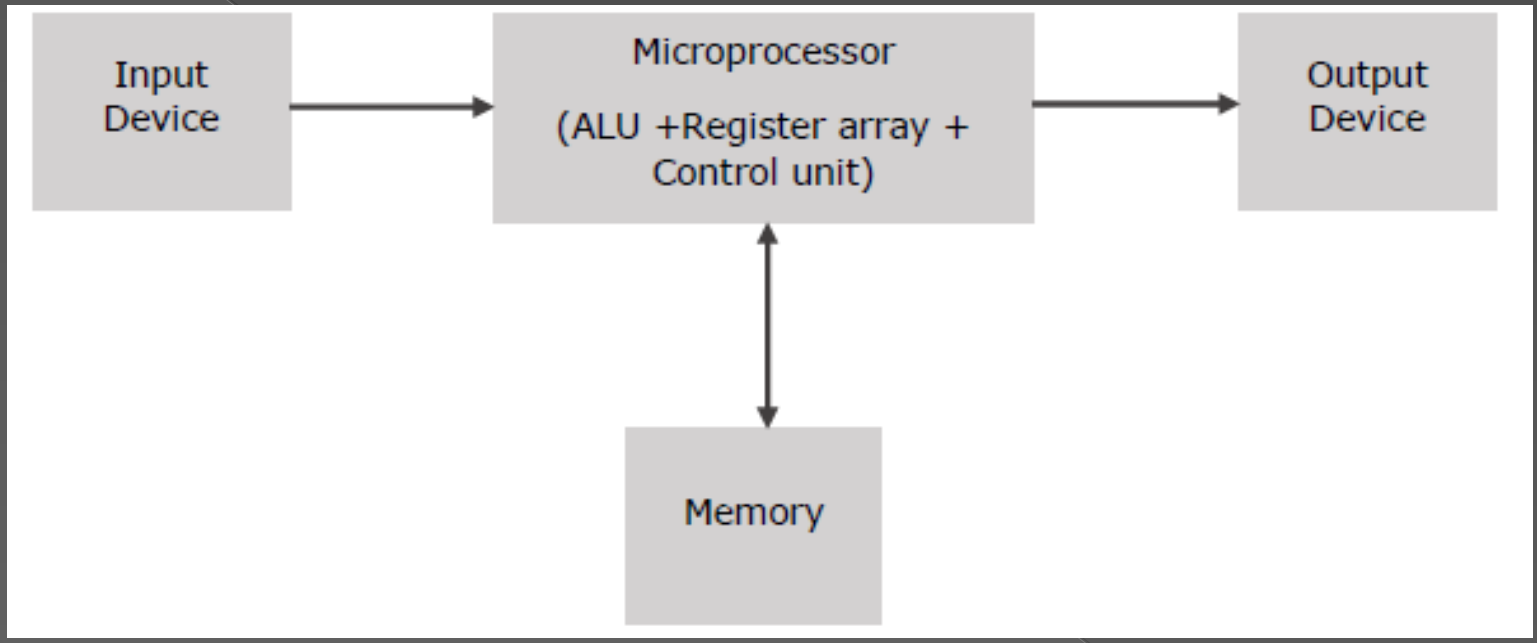

intel
i486 SX


intel
Xeon
Processor


intel
pentium 4

Block Diagram of µp :




Input Device
Microprocessor
(ALU +Register array + Control unit)
Output Device
Memory

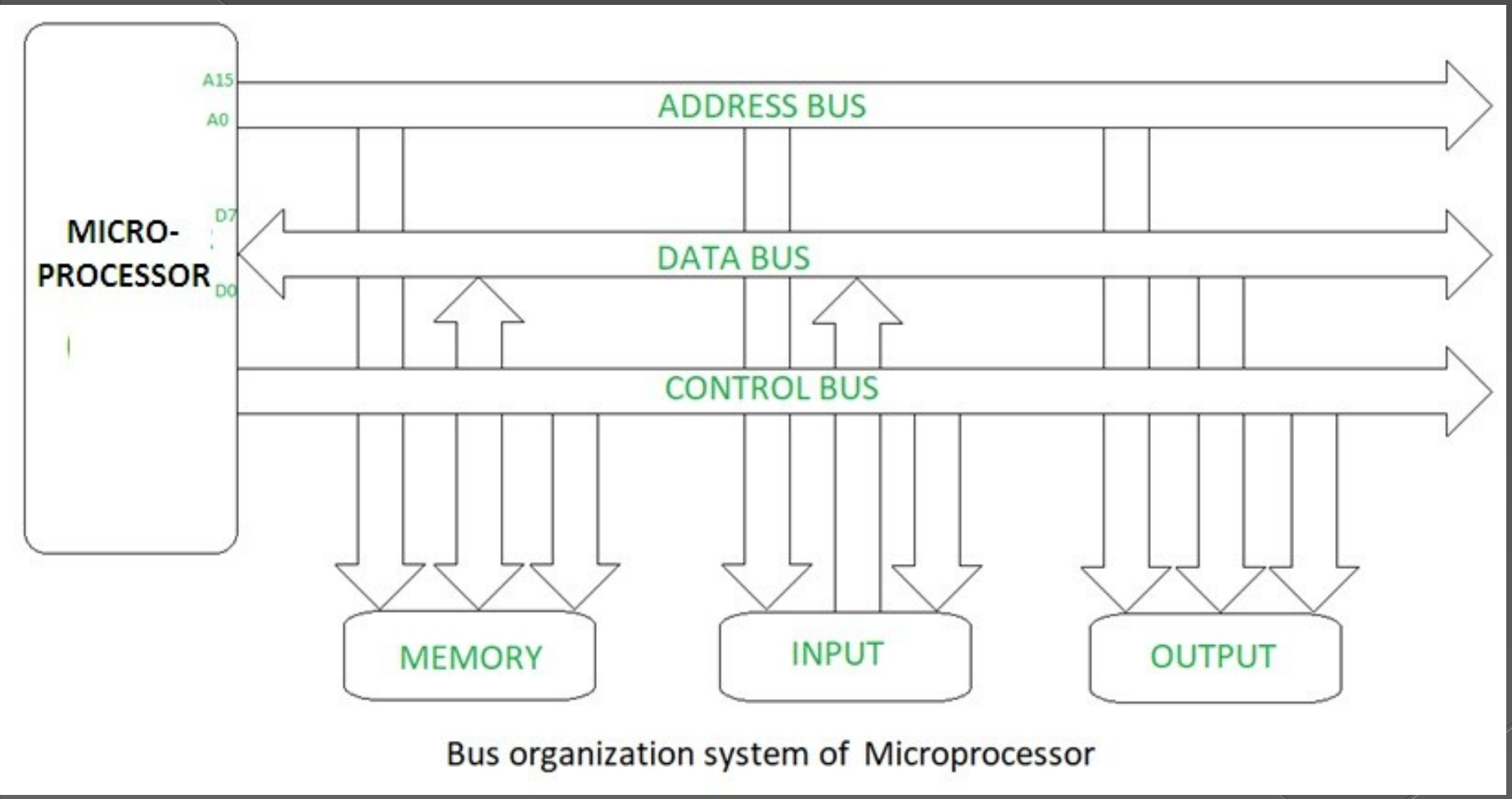
- The microprocessor or CPU reads each instruction from the memory, decodes it and executes it.
- It processes the data as required in the instructions. The processing is in the form of arithmetic and logical operations.
- The data is retrieved from memory or taken from an input device and the result of processing is stored in the memory or delivered to an appropriate output device, all as per the instructions.

Buses

- There are a number of possible interconnection systems
- Single and multiple BUS structures are most common
- e.g. Control/Address/Data bus (PC)
- e.g. Unibus (DEC-PDP)

What is a Bus?

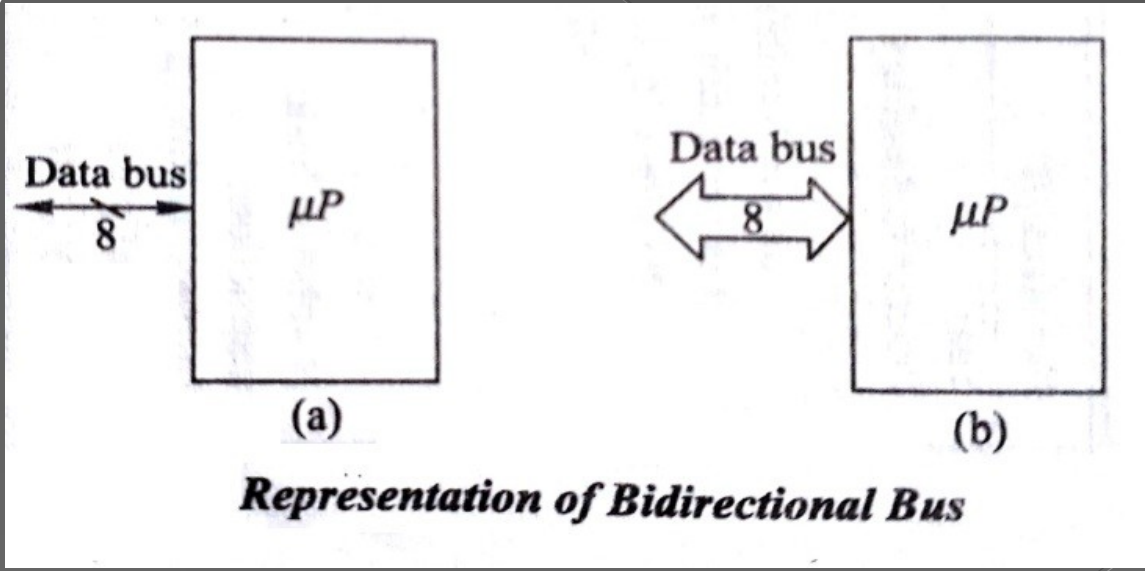
- A communication pathway connecting two or more devices
- Usually broadcast
- Often grouped
  - A number of channels in one bus
  - e.g. 32 bit data bus is 32 separate single bit channels
- Power lines may not be shown



Bus organization system of Microprocessor

Data Bus:

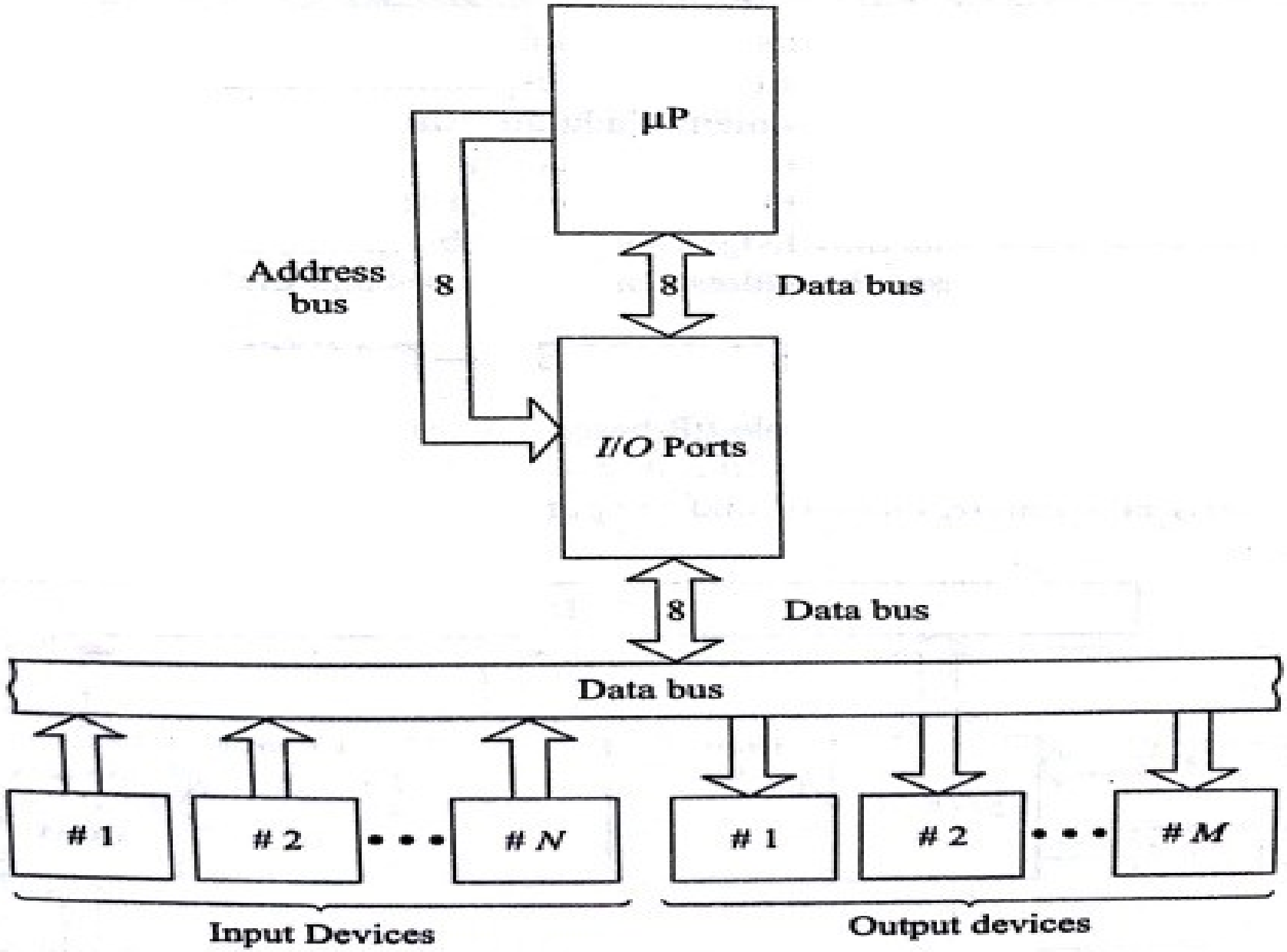
- Data lines that provide a path for moving data among system modules
- May consist of 32, 64, 128, or more separate lines
- The number of lines determines how many bits of data can be transferred at a time.



Representation of Bidirectional Bus

Address 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.
- Used to address I/O ports.

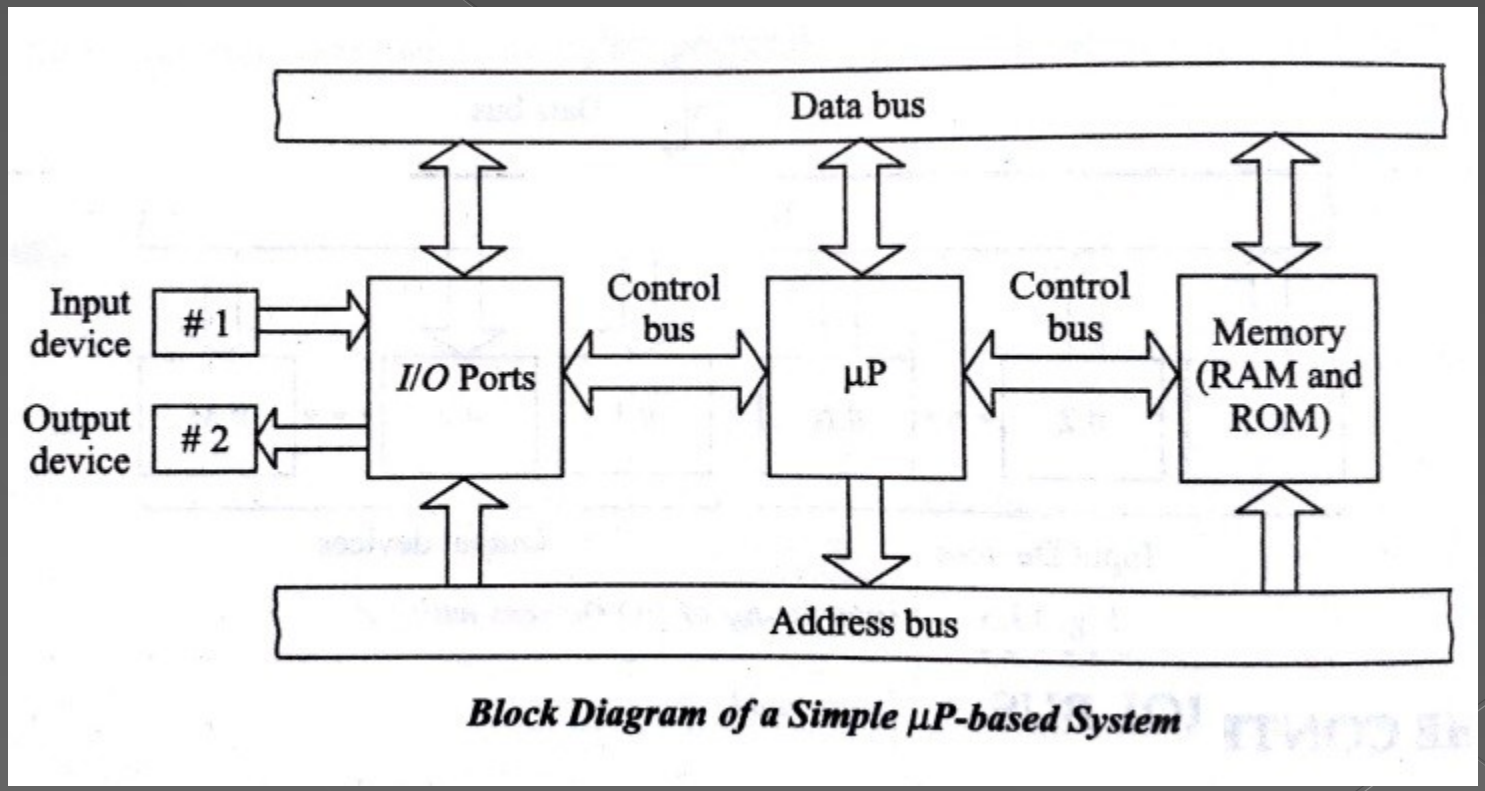


Interfacing of I/O Devices with µP

Control Bus

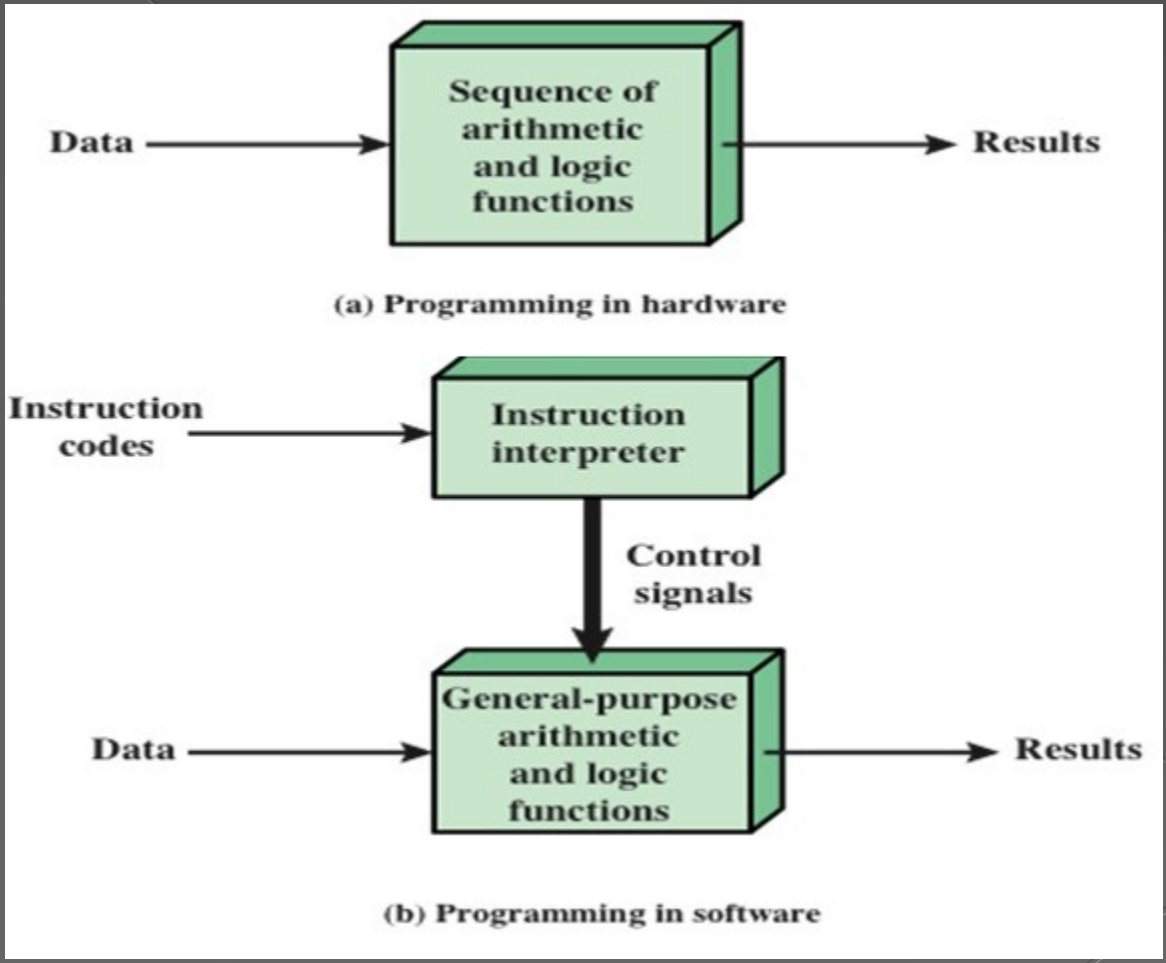
- Used to control the access and the use of the data and address lines.
- Data and Address lines are shared by all components there must be a means of controlling their use.
- Control signals transmit both command and timing information among system modules
  - Timing signals indicate the validity of data and address information
  - Command signals specify operations to be performed

Microprocessor based Systems – Basic Operation :



Block Diagram of a Simple µP-based System

Hardware and Software Approaches

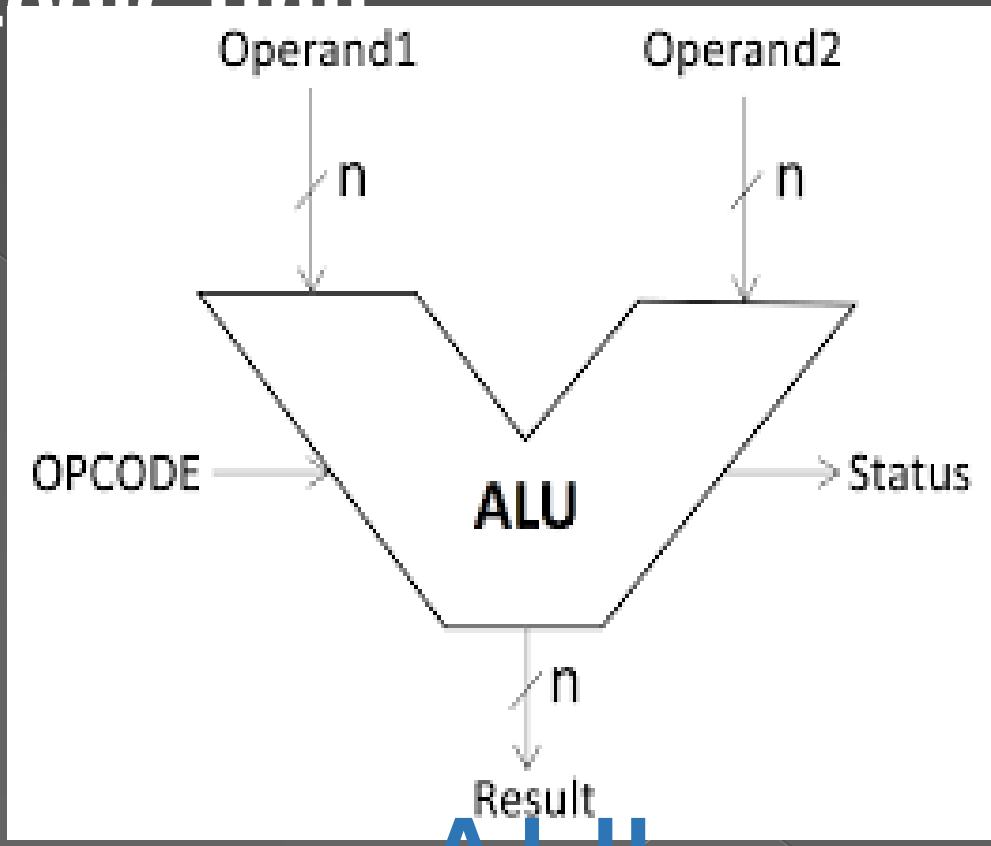


Major components:

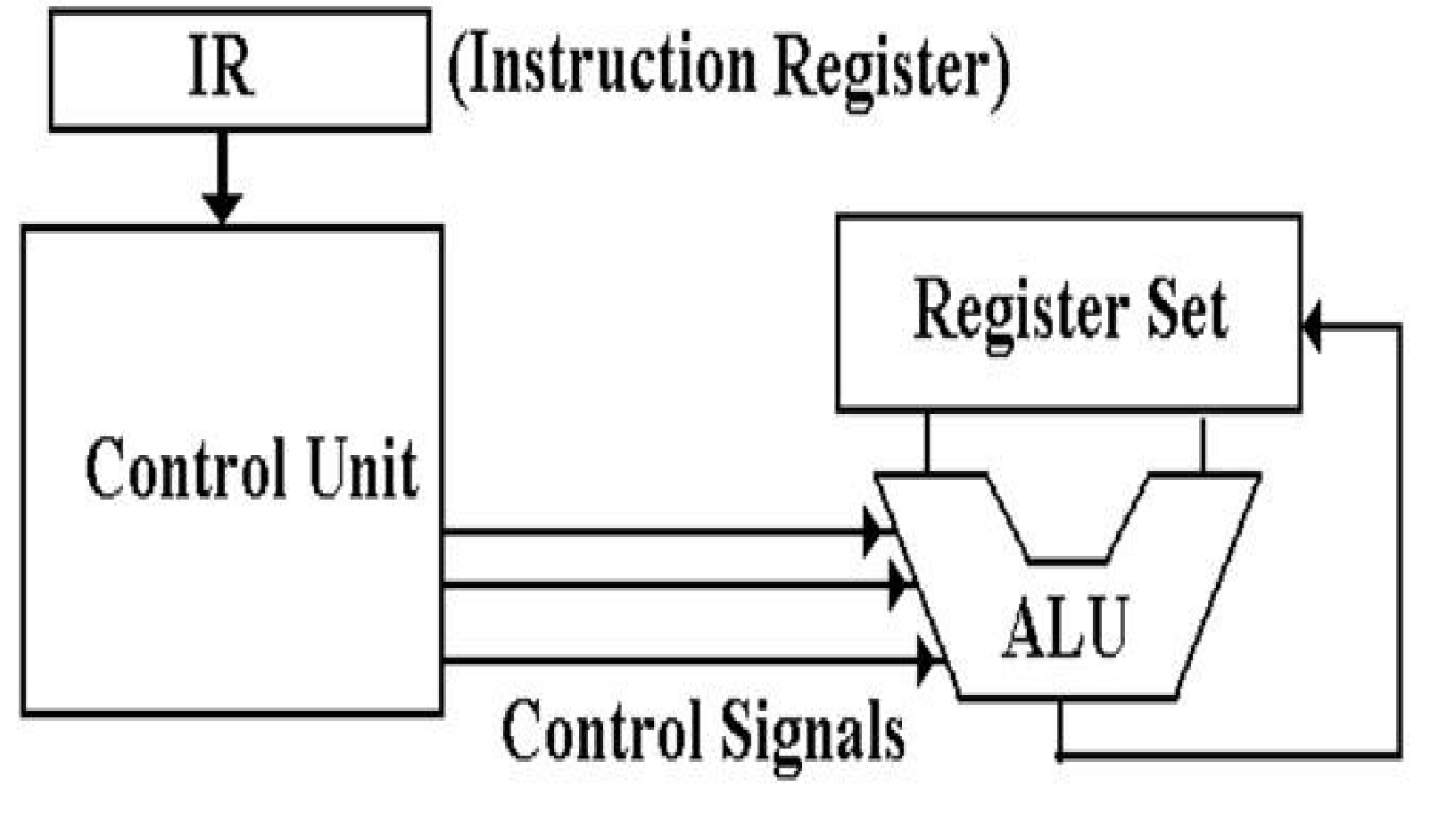
- **CPU**
  - Instruction interpreter
  - Module of general-purpose arithmetic and logic functions
- **I/O Components**
  - Input module
    - Contains basic components for accepting data and instructions and converting them into an internal form of signals usable by the system
  - Output module
   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

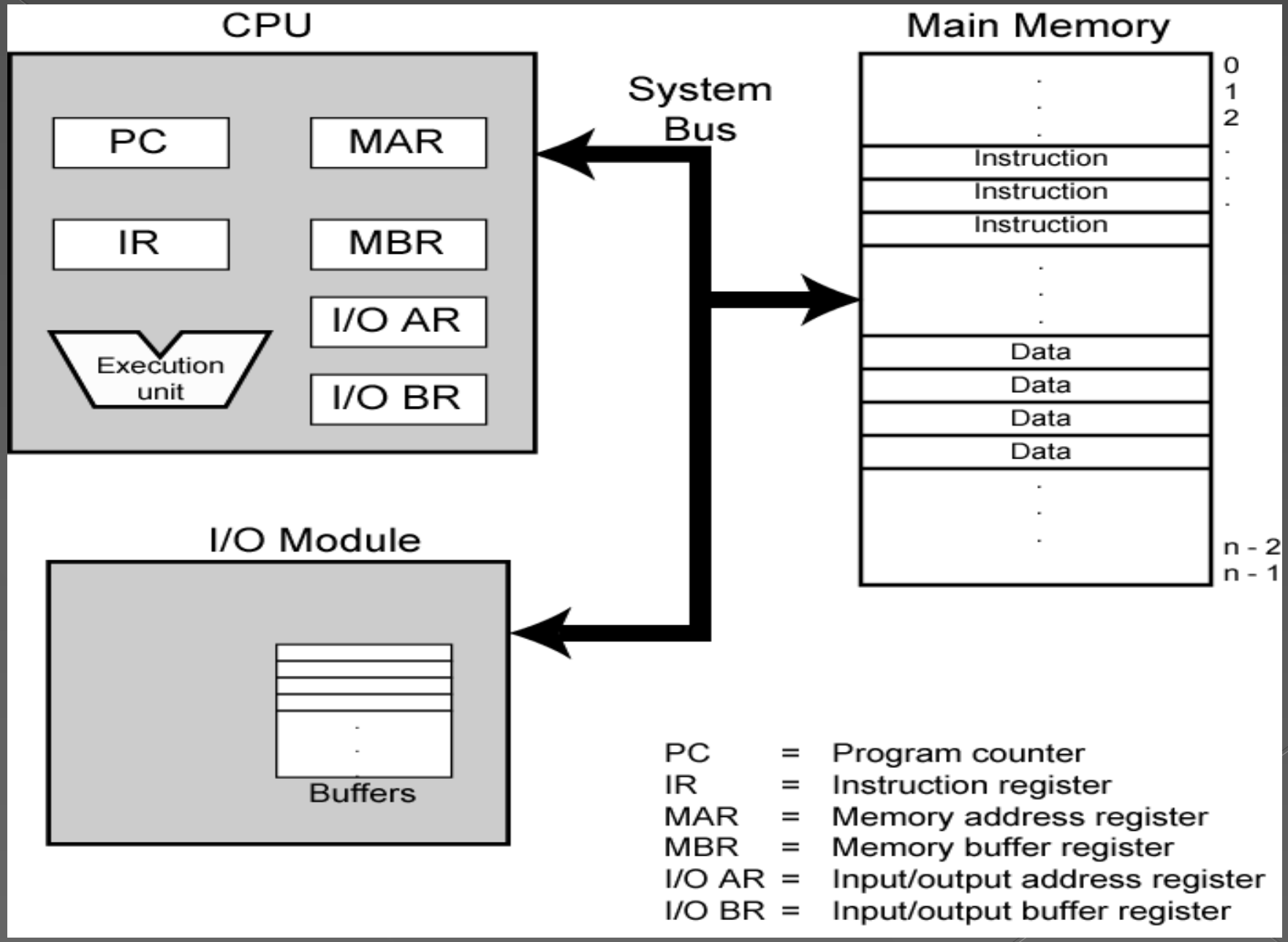
Processor Components

- **ALU- Arithmetic Logic Unit**
- **Registers**
- **Control Unit**



Control Unit



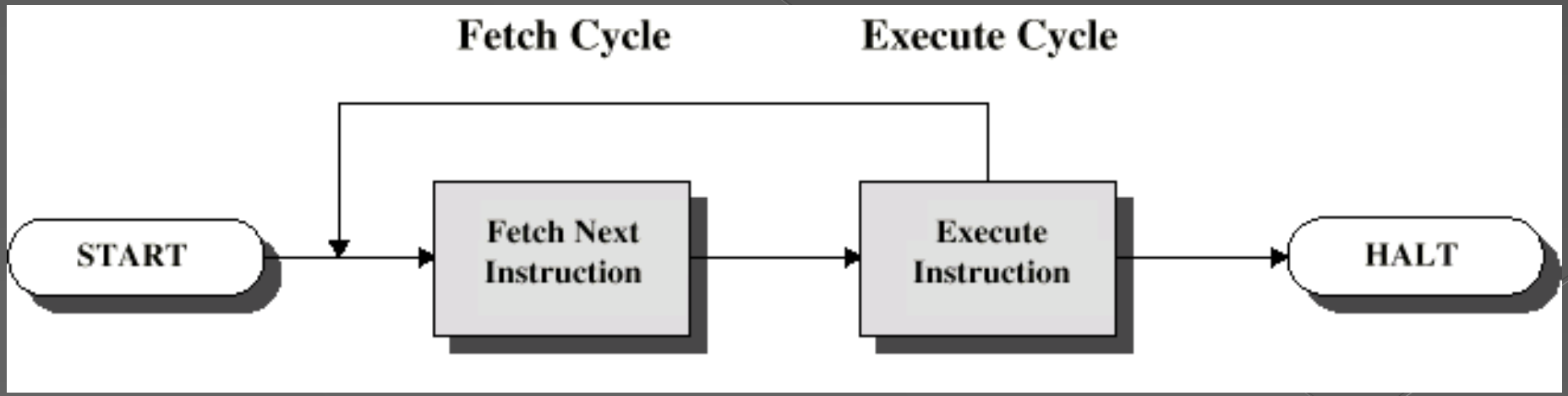

CPU
PC
MAR
IR
MBR
I/O AR
Execution unit
I/O BR
System Bus
Main Memory
0
1
2
Instruction
Instruction
Instruction
Data
Data
Data
Data
n - 2
n - 1
I/O Module
Buffers
PC = Program counter
IR = Instruction register
MAR = Memory address register
MBR = Memory buffer register
I/O AR = Input/output address register
I/O BR = Input/output buffer register

Cont...

- The Control Unit and the Arithmetic and Logic Unit constitute the Central Processing Unit
- Data and instructions need to get into the system and results out
  - Input/output
- Temporary storage of code and results is needed
  - Main memory

Microprocessor based Systems

- Instruction Cycle
- Two steps:
  - Fetch
  - Execute



Fetch Cycle

- Program Counter (PC) holds address of next instruction to fetch
- Processor fetches instruction from memory location pointed to by PC
- Increment PC
  - Unless told otherwise
- Instruction loaded into Instruction Register (IR)
- Processor interprets instruction and performs required actions

Execute Cycle

- Processor-memory
  - data transfer between CPU and main memory
- Processor I/O
  - Data transfer between CPU and I/O module
- Data processing
  - Some arithmetic or logical operation on data
- Control
  - Alteration of sequence of operations
  - e.g. jump
- Combination of above

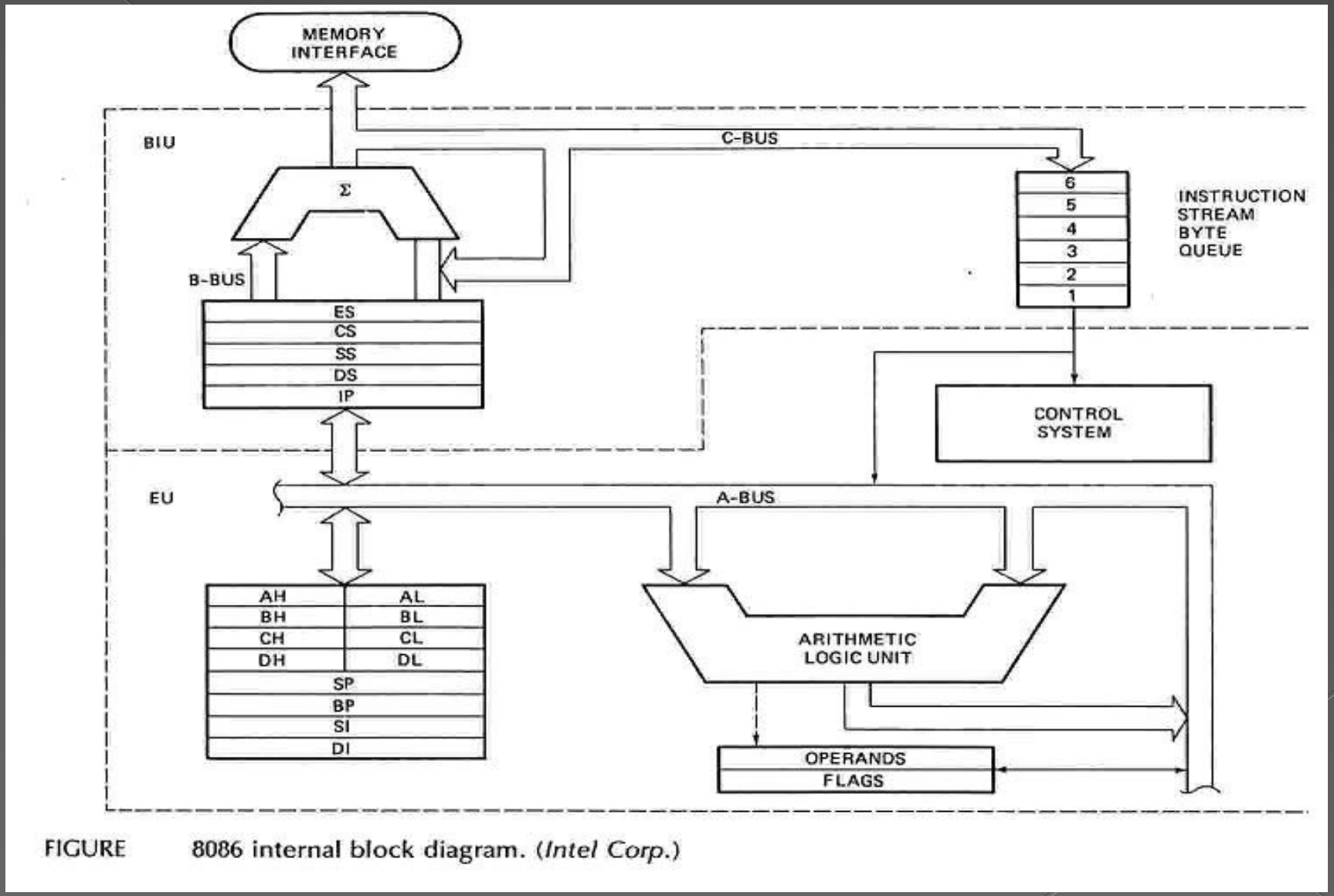


FIGURE 8086 internal block diagram. (*Intel Corp.*)

Internal architecture of 8086

- 8086 has two blocks **BIU** and **EU**. (Independent)
- The BIU handles all transactions of data and addresses on the buses for EU.
- The BIU performs all bus operations such as instruction fetching, reading and writing operands for memory and calculating the addresses of the memory operands. The instruction bytes are transferred to the instruction queue.
- It is also called as external world interface of processor
- EU executes instructions from the instruction system byte queue.

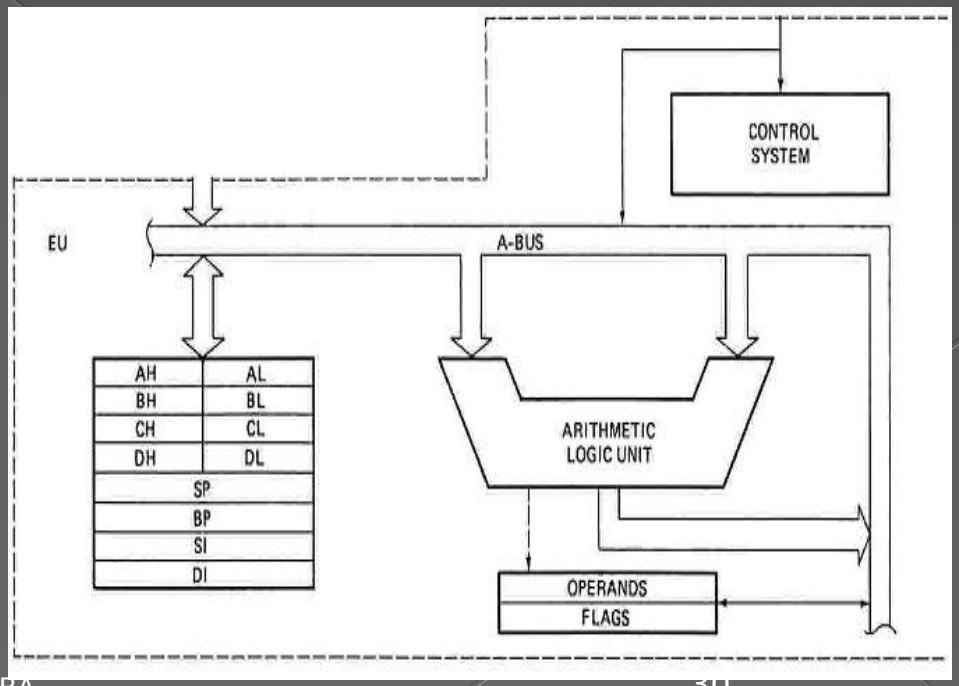
- **Both units operate asynchronously to give the 8086 an overlapping instruction fetch and execution mechanism which is called as Pipelining. This results in efficient use of the system bus and system performance.**
- **BIU contains Instruction queue, Segment registers, Instruction pointer, Address adder.**
- **EU contains Control circuitry, Instruction decoder, ALU, Pointer and Index register, Flag register.**

EXECUTION UNIT

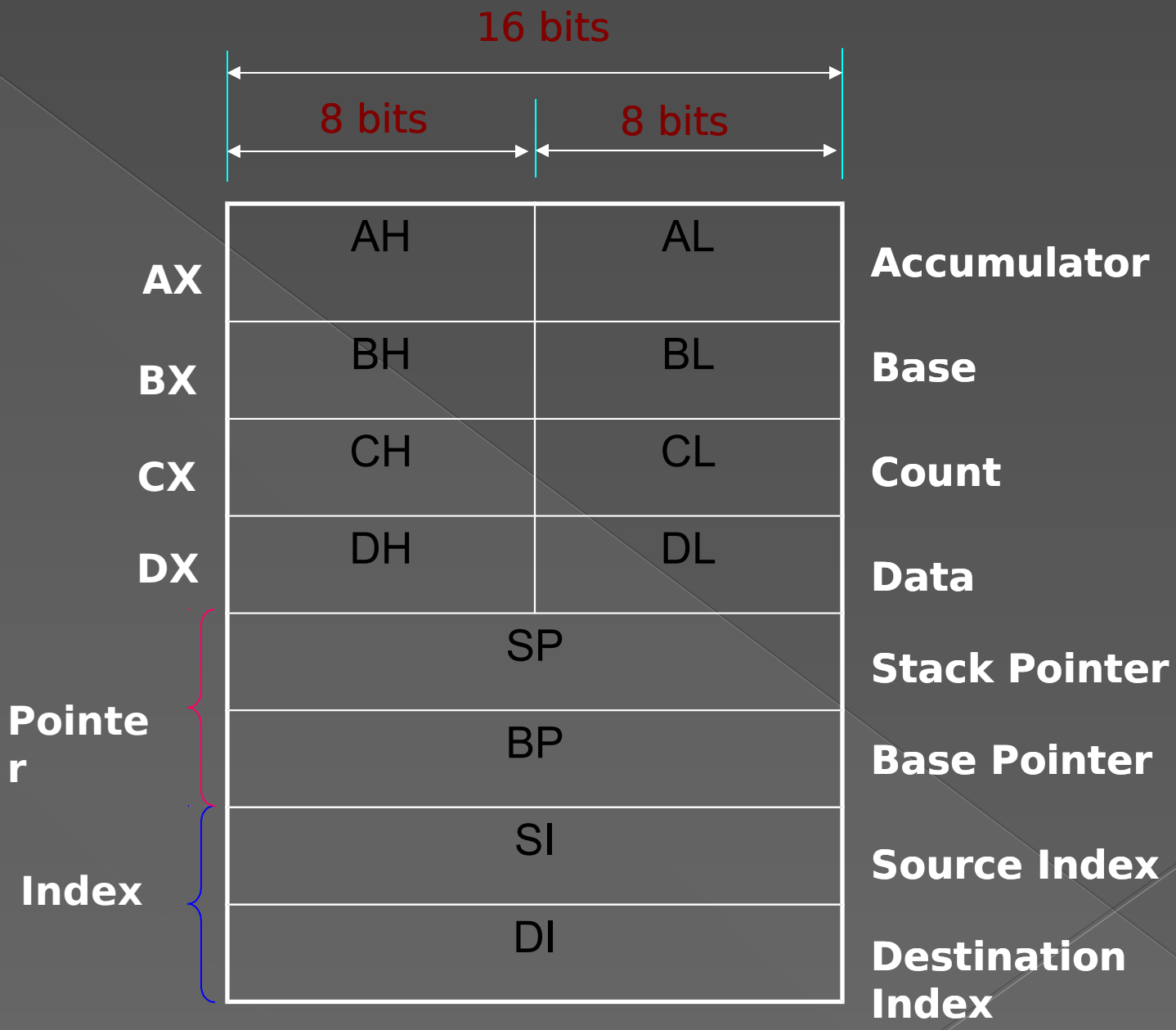
- Decodes instructions fetched by the BIU
- Generate control signals,
- Executes instructions.
- EU is called execution heart of the processor

The main parts are:

- Control Circuitry
- Instruction decoder
- ALU



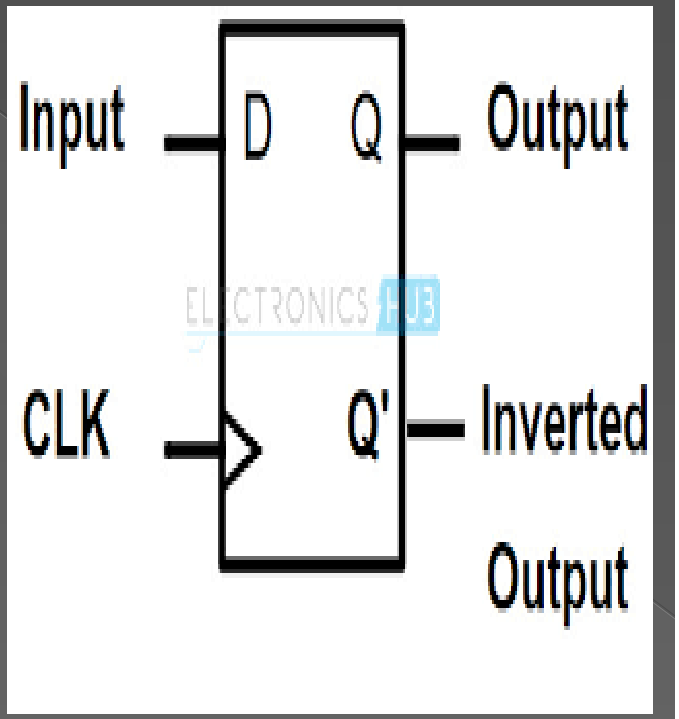
EXECUTION UNIT – General Purpose Registers

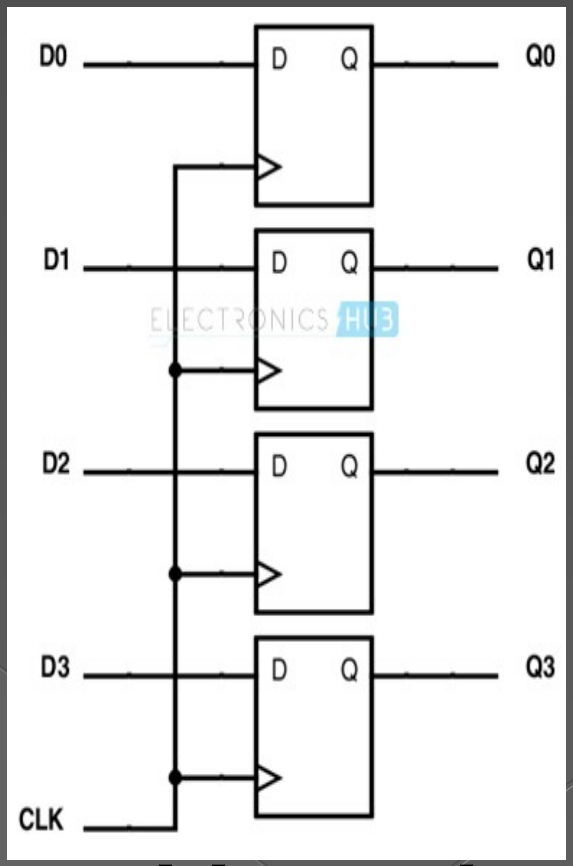


Registers

AX	AH	AL
BX	BH	BL
CX	CH	CL
DX	DH	DL

General data register





Register as data storage

4 bit Register

Pointer And Index Registers

- used to keep offset addresses.
- Used in various forms of memory addressing.
- In the case of SP and BP the default reference to form a physical address is the Stack Segment (SS-will be discussed under the BIU)
- The index registers (SI & DI) and the BX generally default to the Data segment register (DS).

SP: Stack pointer

– Used with SS to access the stack segment

BP: Base Pointer

– Primarily used to access data on the stack

– Can be used to access data in other segments

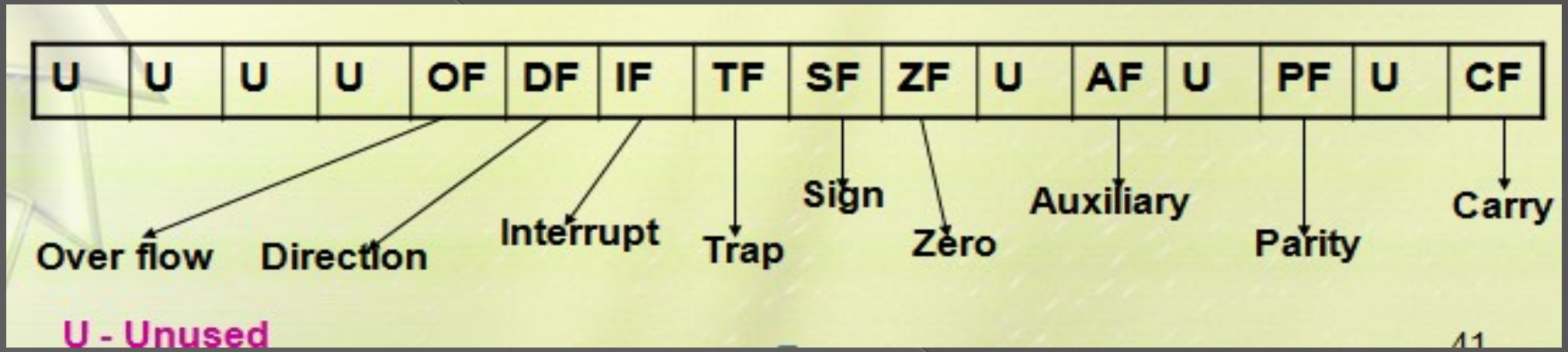
- SI: Source Index register
  – is required for some string operations
  – When string operations are performed, the SI register points to memory locations in the data segment which is addressed by the DS register. Thus, SI is associated with the DS in string operations.

- DI: Destination Index register
  – is also required for some string operations.
  – When string operations are performed, the DI register points to memory locations in the data segment which is addressed by the ES register. Thus, DI is associated with the ES in string operations.

- The SI and the DI registers may also be used to access data stored in arrays

EXECUTION UNIT – Flag Register

- A flag is a flip flop which indicates some conditions produced by the execution of an instruction or controls certain operations of the EU .
- In 8086 The EU contains
  - a 16 bit flag register
  - 9 of the 16 are active flags and remaining 7 are undefined.
  - 6 flags indicates some conditions- status flags
  - 3 flags –control Flags


U U U U OF DF IF TF SF ZF U AF U PF U CF
Over flow
Direction
Interrupt
Trap
Sign
Zero
Auxiliary
Parity
Carry
U - Unused

EXECUTION UNIT – Flag Register

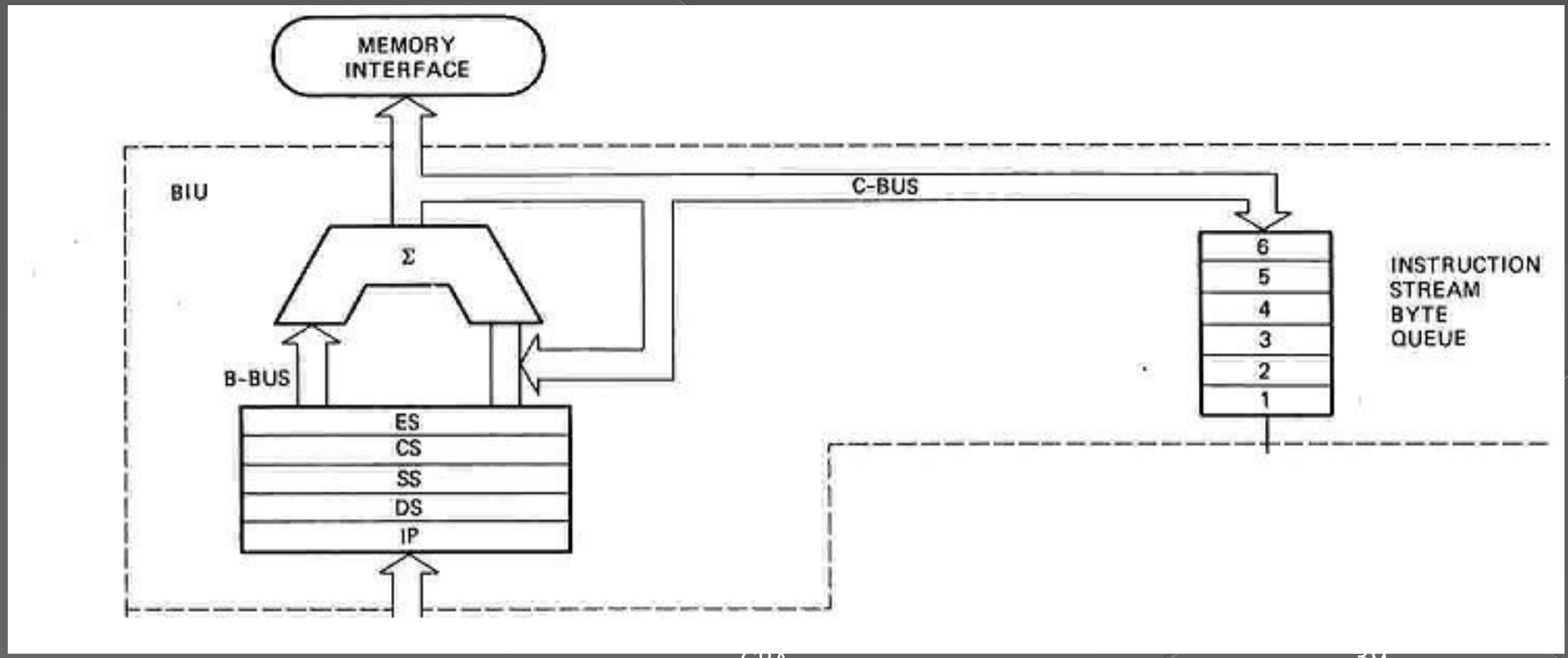
Flag	Purpose
Carry (CF)	Holds the carry after addition or the borrow after subtraction. Also indicates some error conditions, as dictated by some programs and procedures .
Parity (PF)	PF=0;odd parity, PF=1;even parity.
Auxiliary (AF)	Holds the carry (half – carry) after addition or borrow after subtraction between bit positions 3 and 4 of the result (for example, in BCD addition or subtraction.)
Zero (ZF)	Shows the result of the arithmetic or logic operation. Z=1; result is zero. Z=0; The result is 0
Sign (SF)	Holds the sign of the result after an arithmetic/logic instruction execution. S=1; negative, S=0

Flag	Purpose
Trap (TF)	A control flag. Enables the trapping through an on-chip debugging feature.
Interrupt (IF)	A control flag. Controls the operation of the INTR (interrupt request) I=0; INTR pin disabled. I=1; INTR pin enabled.
Direction (DF)	A control flag. It selects either the increment or decrement mode for DI and /or SI registers during the string instructions.
Overflow (OF)	Overflow occurs when signed numbers are added or subtracted. An overflow indicates the result has exceeded the capacity of the Machine

BUS INTERFACE UNIT (BIU)

Contains

- **6-byte Instruction Queue (Q)**
- **The Segment Registers (CS, DS, ES, SS).**
- **The Instruction Pointer (IP).**
- **The Address Summing block (Σ)**



Functional Units of Microprocessor –

- Does the calculations
- Everything else in the computer is there to service this unit
- Handles integers
- May handle floating point (real) numbers
- May be separate FPU (maths co-processor)
- May be on chip separate FPU (486DX +)

Typical Schematic Symbol of an ALU

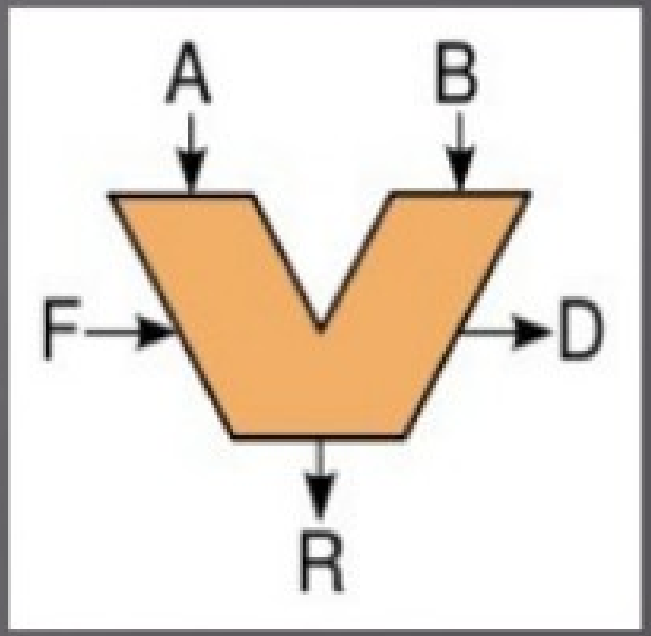
A and B: the inputs to the ALU (aka operands)

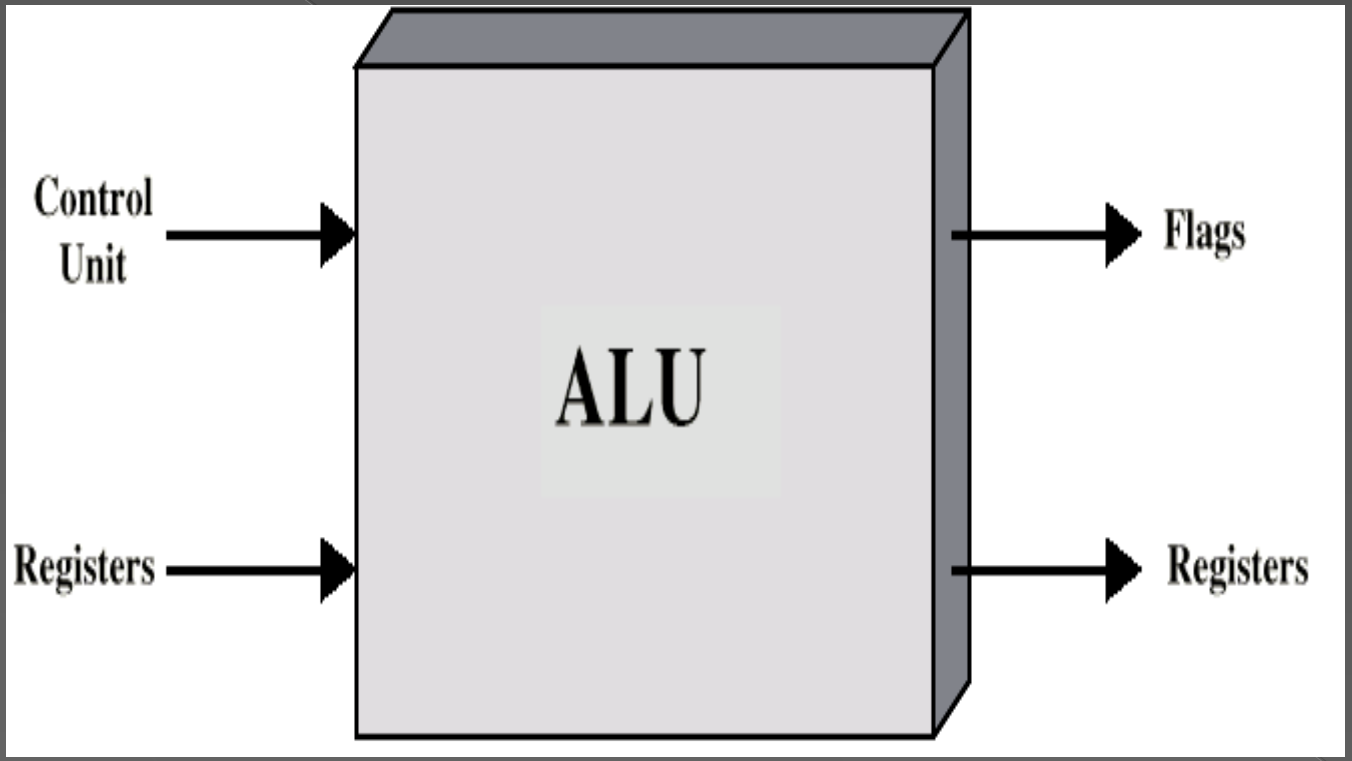
R: Output or Result

F: Code or Instruction from the Control Unit (aka as op-code)

D: Output status; it indicates cases such as:

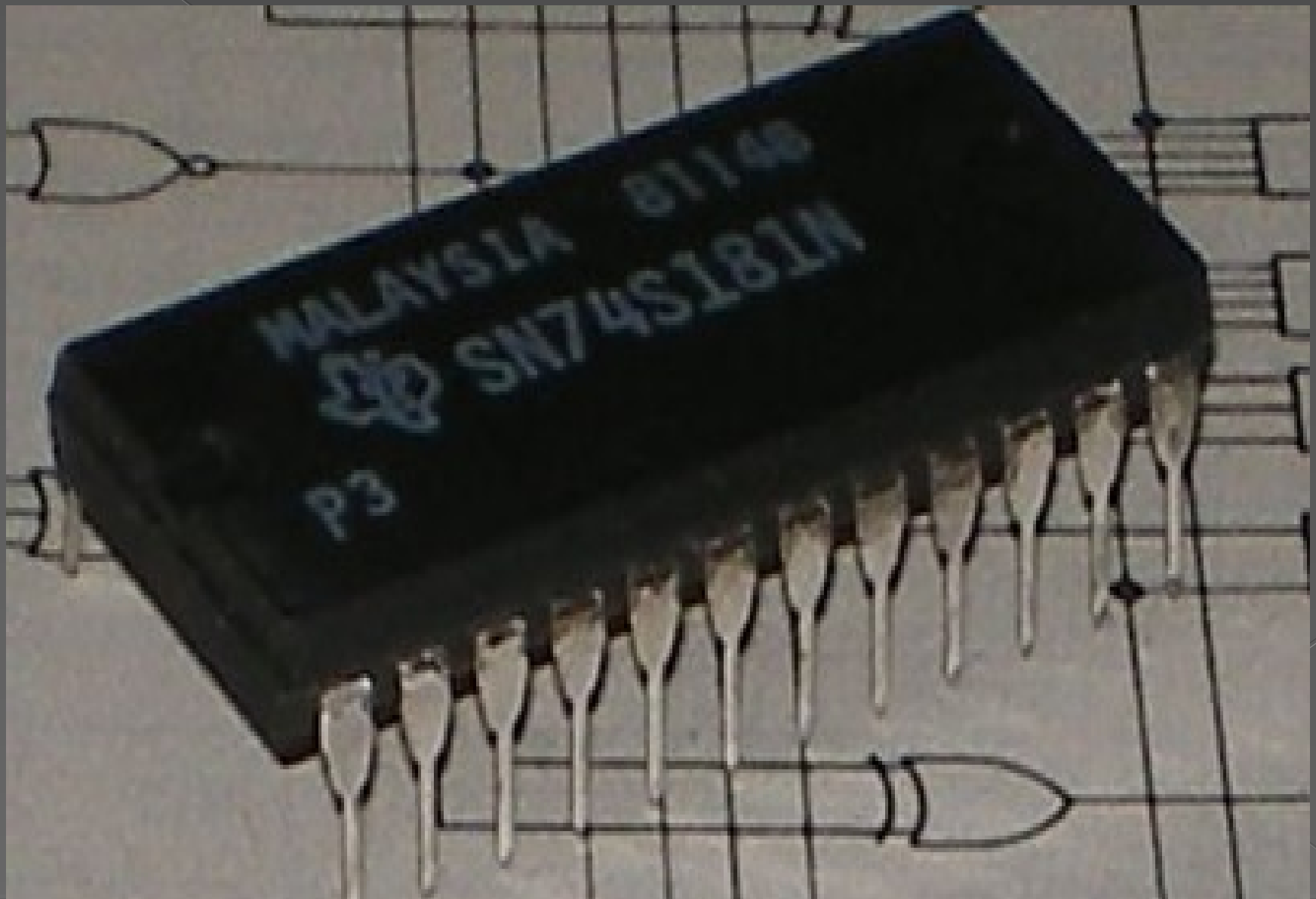
- carry-in
- carry-out,
- overflow,
- division-by-zero
- And . . .



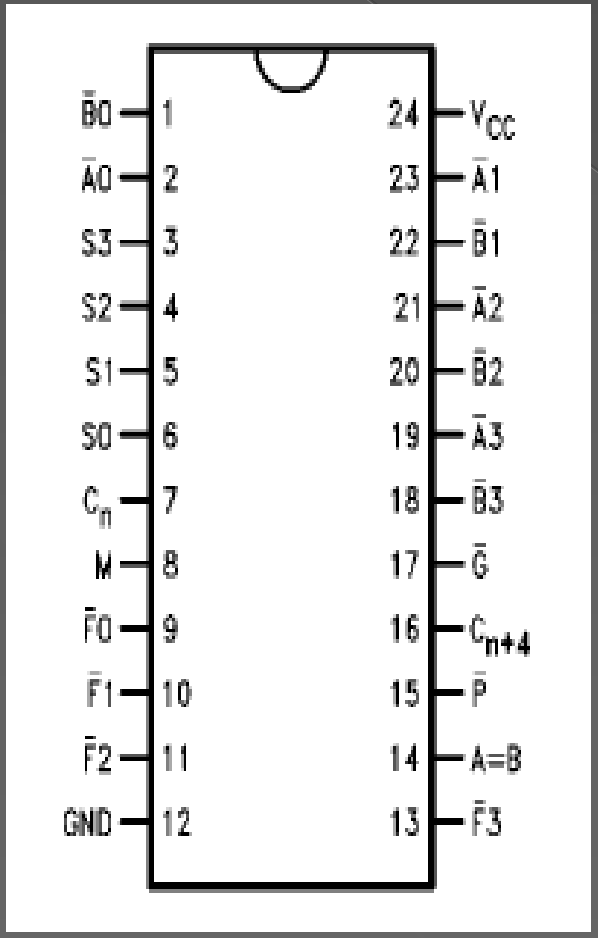

Control
Unit
Registers
ALU
Flags
Registers

ALU using IC 74181

- 74LS181 4-Bit Arithmetic Logic Unit
- 4-bit Arithmetic Logic
- 16 logic operations
- Provides 16 arithmetic operations
- Provides all 16 logic operations of two variables
- Full lookahead for high speed arithmetic operation on long words


MALAYSIA
SN74S181N
P3

Cont..



Pin Descriptions

Pin Names	Description
$\overline{A}0$–$\overline{A}3$	Operand Inputs (Active LOW)
$\overline{B}0$–$\overline{B}3$	Operand Inputs (Active LOW)
S0–S3	Function Select Inputs
M	Mode Control Input
C_n	Carry Input
$\overline{F}0$–$\overline{F}3$	Function Outputs (Active LOW)
A = B	Comparator Output
$\overline{G}$	Carry Generate Output (Active LOW)
$\overline{P}$	Carry Propagate Output (Active LOW)
C_{n+4}	Carry Output

Cont..

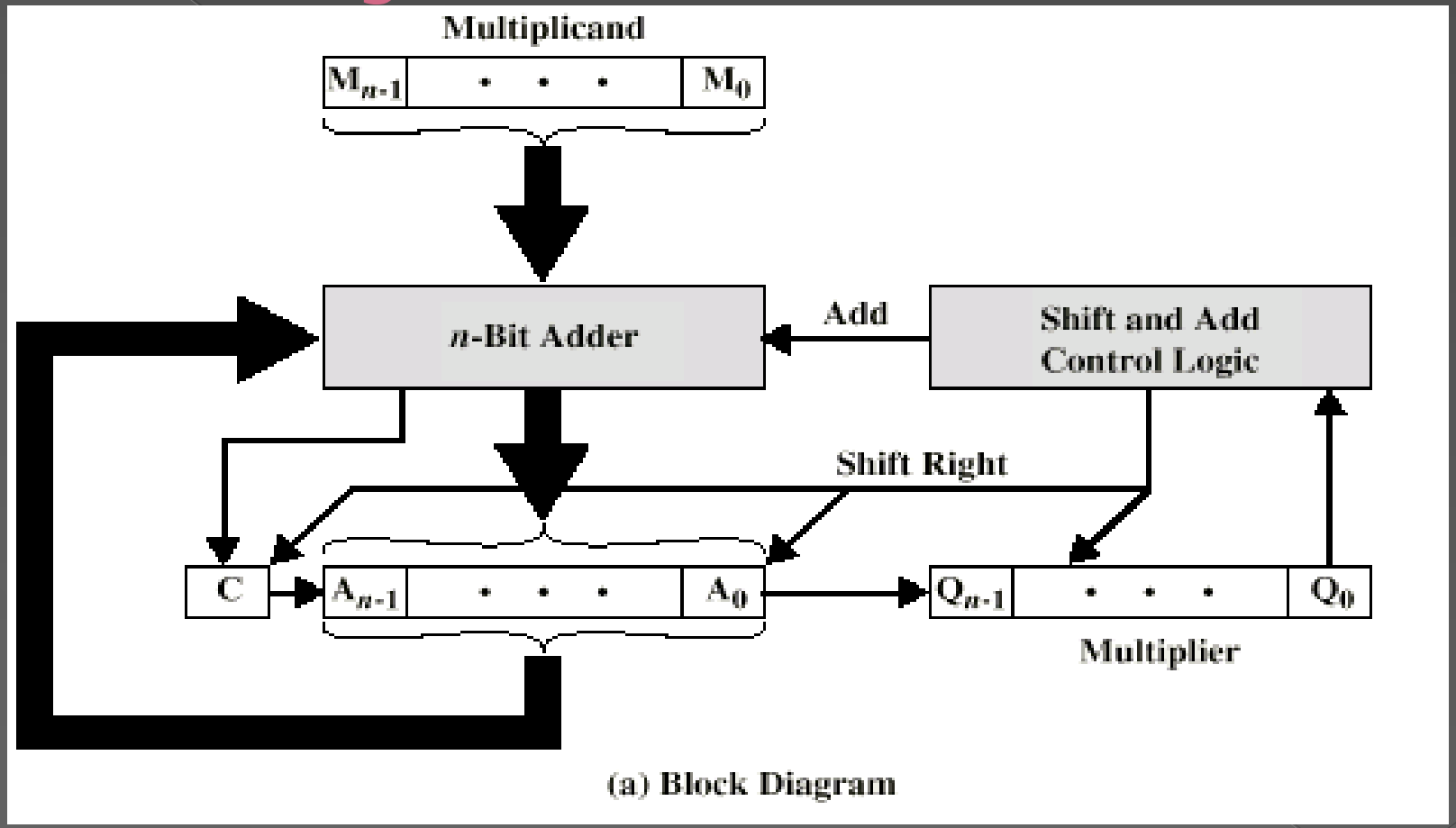
- M=High=Logical operation
- M=Low=Arithmetic operation

- Device can be use with either active low inputs producing active low output

- Active HIGH inputs producing Active High

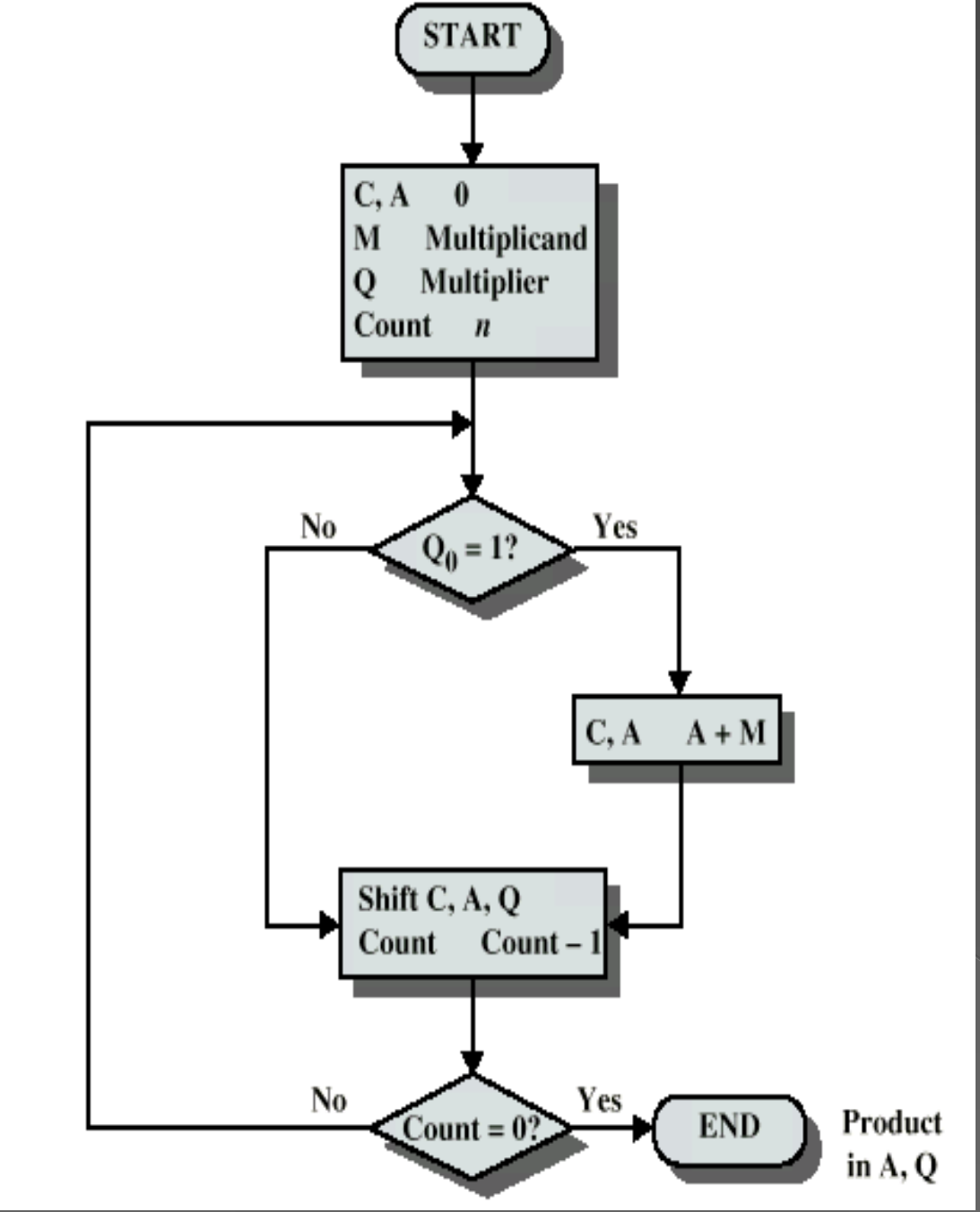
Selection				Active-low inputs & outputs		Active-high inputs & outputs	
S3	S2	S1	S0	Logic (M = 1)	Arithmetic (M = 0) (Cn = 0)	Logic (M = 1)	Arithmetic (M = 0) (Cn = 1)
0	0	0	0	$\overline{A}$	A minus 1	$\overline{A}$	A
0	0	0	1	$\overline{AB}$	AB minus 1	$\overline{A+B}$	$A+B$
0	0	1	0	$\overline{A}+B$	$A\overline{B}$ minus 1	$\overline{A}B$	$A+\overline{B}$
0	0	1	1	Logical 1	-1	Logical 0	-1
0	1	0	0	$\overline{A+B}$	A plus $(A+\overline{B})$	$\overline{AB}$	A plus $(A\overline{B})$
0	1	0	1	$\overline{B}$	AB plus $(A+\overline{B})$	$\overline{B}$	$(A+B)$ plus $(A\overline{B})$
0	1	1	0	$\overline{A\oplus B}$	A minus B minus 1	$A\oplus B$	A minus B
0	1	1	1	$A+\overline{B}$	$A+\overline{B}$	$A\overline{B}$	$A\overline{B}$ minus 1
1	0	0	0	$\overline{A}B$	A plus $(A+B)$	$\overline{A}+B$	A plus AB
1	0	0	1	$A\oplus B$	A plus B	$\overline{A\oplus B}$	A plus B plus 1
1	0	1	0	B	$A\overline{B}$ plus $(A+B)$	B	$(A+\overline{B})$ plus AB
1	0	1	1	$A+B$	$A+B$	AB	AB minus 1
1	1	0	0	Logical 0	A plus A	Logical 1	A plus A
1	1	0	1	$A\overline{B}$	AB plus A	$A+\overline{B}$	$(A+B)$ plus A
1	1	1	0	AB	$A\overline{B}$ plus A	$A+B$	$(A+\overline{B})$ plus A
1	1	1	1	A	A	A	A minus 1

4-bit Multiplier circuit using ALU and shift registers.



(a) Block Diagram

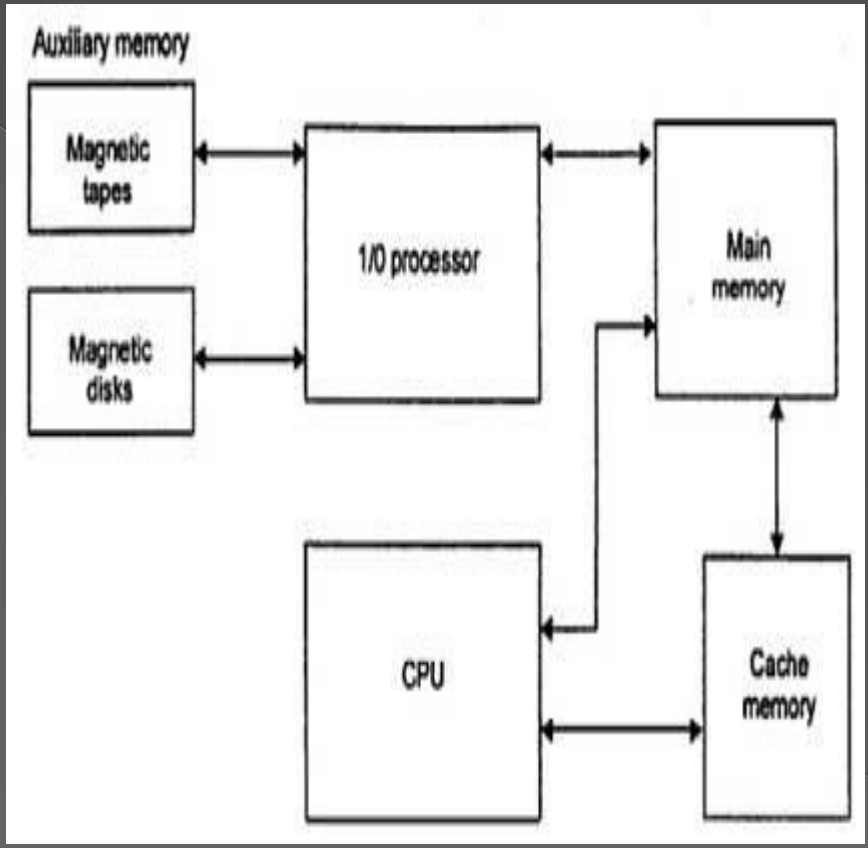
- Complex
- Work out partial product for each digit
- Take care with place value (column)
- Add partial products


START
C, A 0
M Multiplicand
Q Multiplier
Count n
No
$Q_0 = 1$?
Yes
C, A A + M
Shift C, A, Q
Count Count – 1
No
Count = 0?
Yes
END
Product in A, Q

C	A	Q	M		
0	0000	1101	1011	Initial Values	
0	1011	1101	1011	Add	First Cycle
0	0101	1110	1011	Shift	
0	0010	1111	1011	Shift	Second Cycle
0	1101	1111	1011	Add	Third Cycle
0	0110	1111	1011	Shift	
1	0001	1111	1011	Add	Fourth Cycle
0	1000	1111	1011	Shift	

Memory Organization

- Memory Hierarchy
- **Memory is used for storing programs and data that are required to perform a specific task.**
- For CPU to operate at its maximum speed, it required an uninterrupted and high speed access to these **memories that contain programs and data. Some of the criteria need to be taken into consideration while deciding which memory is to be used:**
- • Cost
- • Speed
- • Memory access time
- • Data transfer rate
- • Reliability

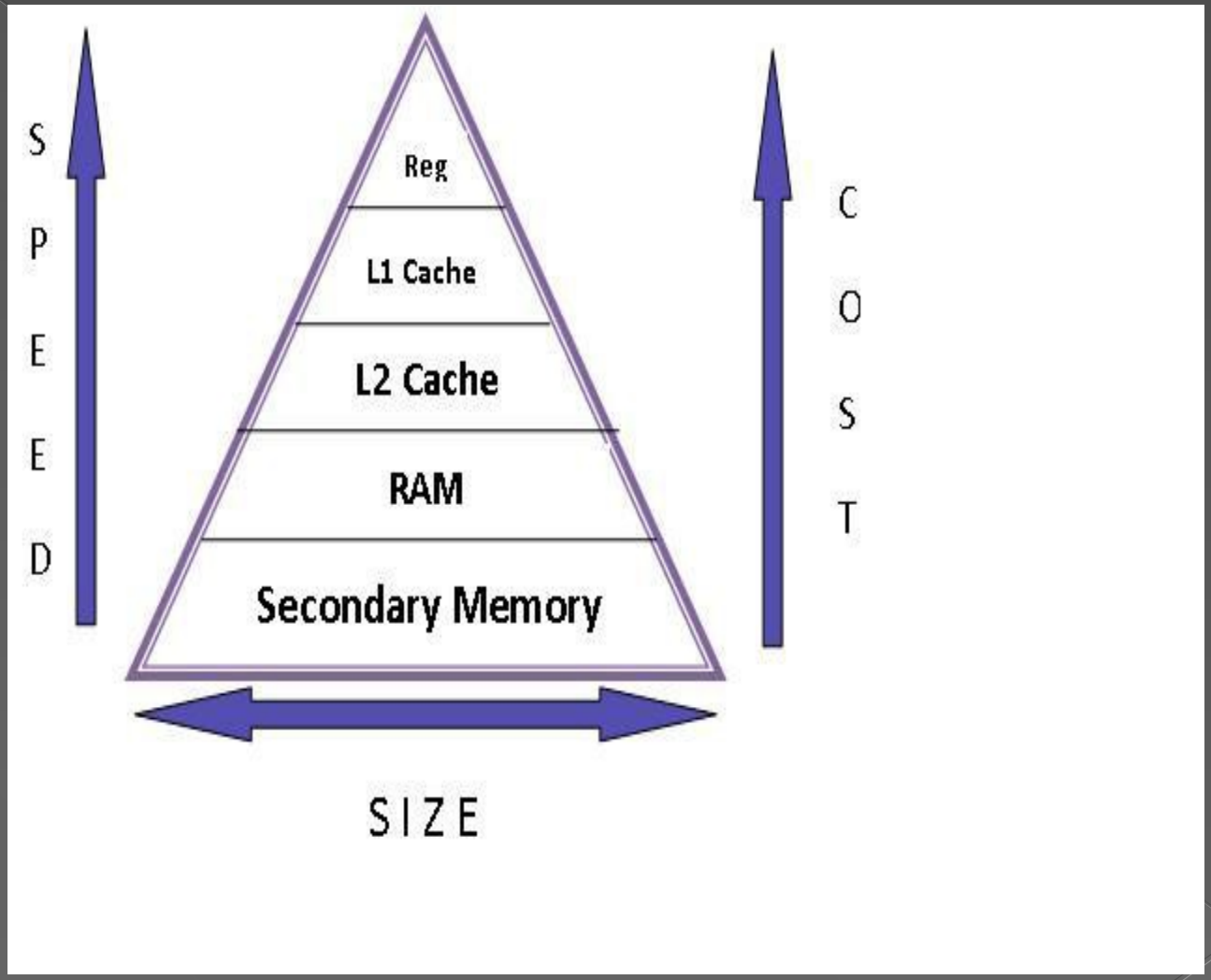

Auxiliary memory
Magnetic tapes
Magnetic disks
1/0 processor
Main memory
CPU
Cache memory

- A computer system contains various types of memories like auxiliary memory, cache memory, and main memory.

- **Auxiliary Memory**

- **Main Memory**

- **Cache Memory**


SPEED
Reg
L1 Cache
L2 Cache
RAM
Secondary Memory
COST
SIZE

Main Memory

- Central storage unit in a computer system
- Large memory
- Made up of Integrated chips

Types:

- RAM (Random access memory)
- ROM (Read only memory)

Cont..

- RAM (Random Access Memory)
- Random access memory (RAM) is the best known form of computer memory. RAM is considered "random access" because you can access any memory cell directly if you know the row and column that intersect at that cell.
- Types of RAM:-
- •Static RAM (SRAM)
- •Dynamic RAM (DRAM)

Static RAM (SRAM)

- a bit of data is stored using the state of a flip-flop.
- Retains value indefinitely, as long as it is kept powered.
- Mostly uses to create cache memory of CPU.
- Faster and more expensive than DRAM.
- •

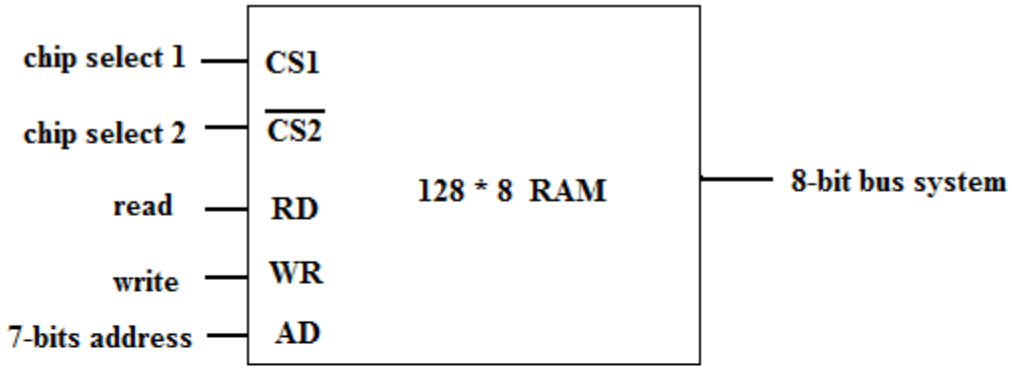
Dynamic RAM (DRAM)

- Each cell stores bit with a capacitor and transistor.
- Large storage capacity
- Needs to be refreshed frequently.
- Used to create main memory.
- Slower and cheaper than SRAM.

RAM and ROM Chips

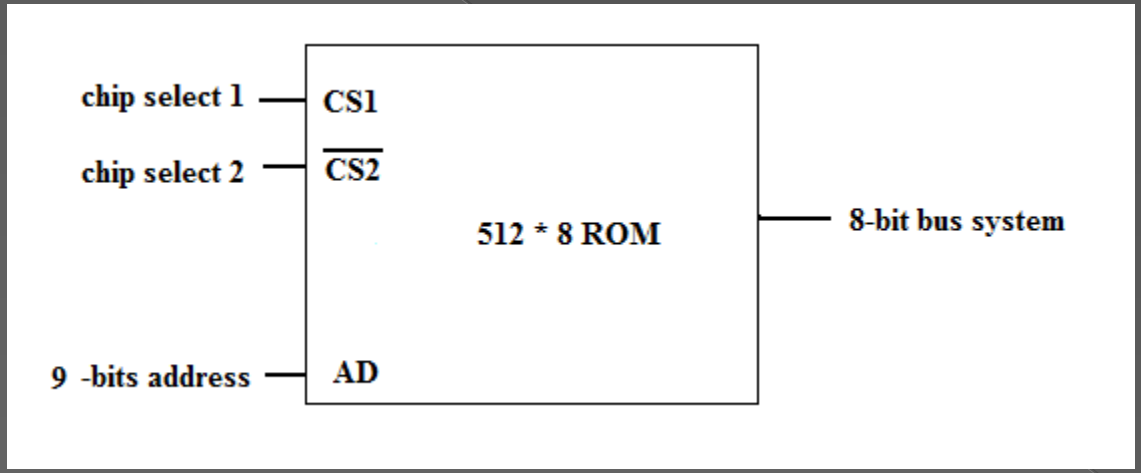
- A RAM chip is better suited for communication with the CPU if it has one or more control inputs that select the chip when needed
- •The Block diagram of a RAM chip is shown next slide, the capacity of the memory is 128 words of 8 bits (one byte) per word

RAM



CS1	$\overline{CS2}$	RD	WD	Memory Function	State of data bus
0	0	*	*	Inhibit	High-impedance
0	1	*	*	Inhibit	High-impedance
1	0	0	0	Inhibit	High-impedance
1	0	0	1	Write	Input data to RAM
1	0	1	*	Read	Output data from RAM
1	1	*	*	Inhibit	High-impedance

ROM

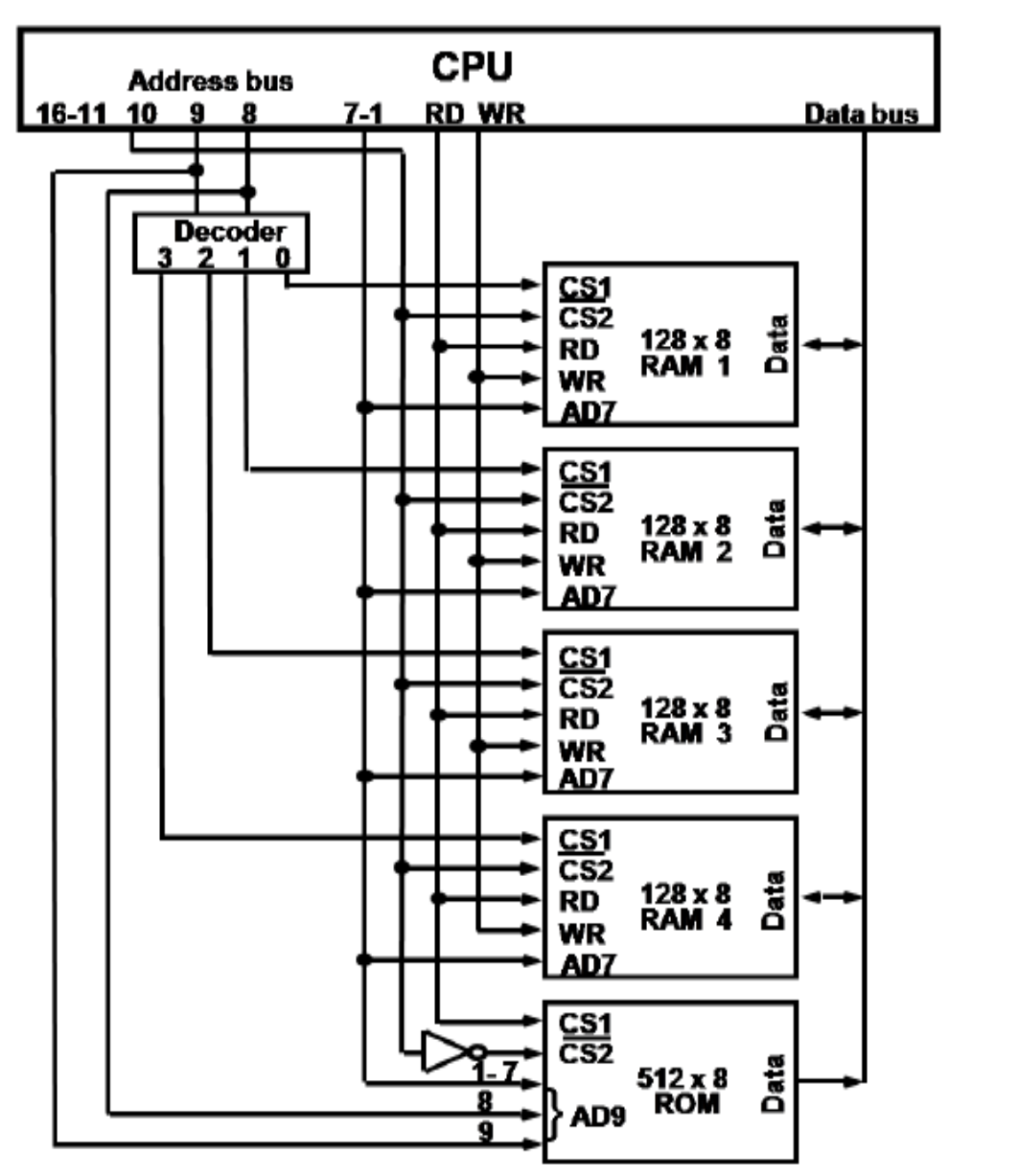

chip select 1
CS1
chip select 2
CS2
512 * 8 ROM
8-bit bus system
9 -bits address
AD

Memory Address Map

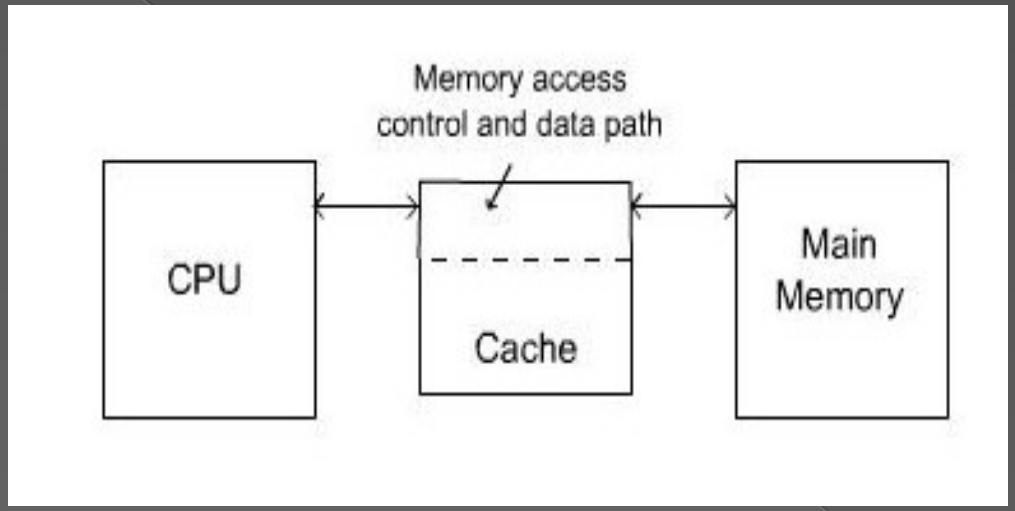
- Memory Address Map is a pictorial representation of assigned address space for each chip in the system
- To demonstrate an example, assume that a computer system needs 512 bytes of RAM and 512 bytes of ROM
- •he RAM have 128 byte and need seven address lines, where the ROM have 512 bytes and need 9 address lines

Component	Hexadecimal Address	10	9	8	7	6	5	4	3	2	1
RAM1	0000-007F	0	0	0	*	*	*	*	*	*	*
RAM2	0080-00FF	0	0	1	*	*	*	*	*	*	*
RAM3	0100-017F	0	1	0	*	*	*	*	*	*	*
RAM4	0180-01FF	0	1	1	*	*	*	*	*	*	*
ROM	0200-03FF	1	*	*	*	*	*	*	*	*	*

- The hexadecimal address assigns a range of hexadecimal equivalent address for each chip
- Line 8 and 9 represent four distinct binary combination to specify which RAM we chose
- When line 10 is 0, CPU selects a RAM. And when it’s 1, it selects the ROM


CPU
Address bus
16-11 10 9 8
7-1
RD WR
Data bus
Decoder
3 2 1 0
CS1
CS2
RD
WR
AD7
128 x 8
RAM 1
Data
CS1
CS2
RD
WR
AD7
128 x 8
RAM 2
Data
CS1
CS2
RD
WR
AD7
128 x 8
RAM 3
Data
CS1
CS2
RD
WR
AD7
128 x 8
RAM 4
Data
CS1
CS2
1-7
8
9
AD9
512 x 8
ROM
Data

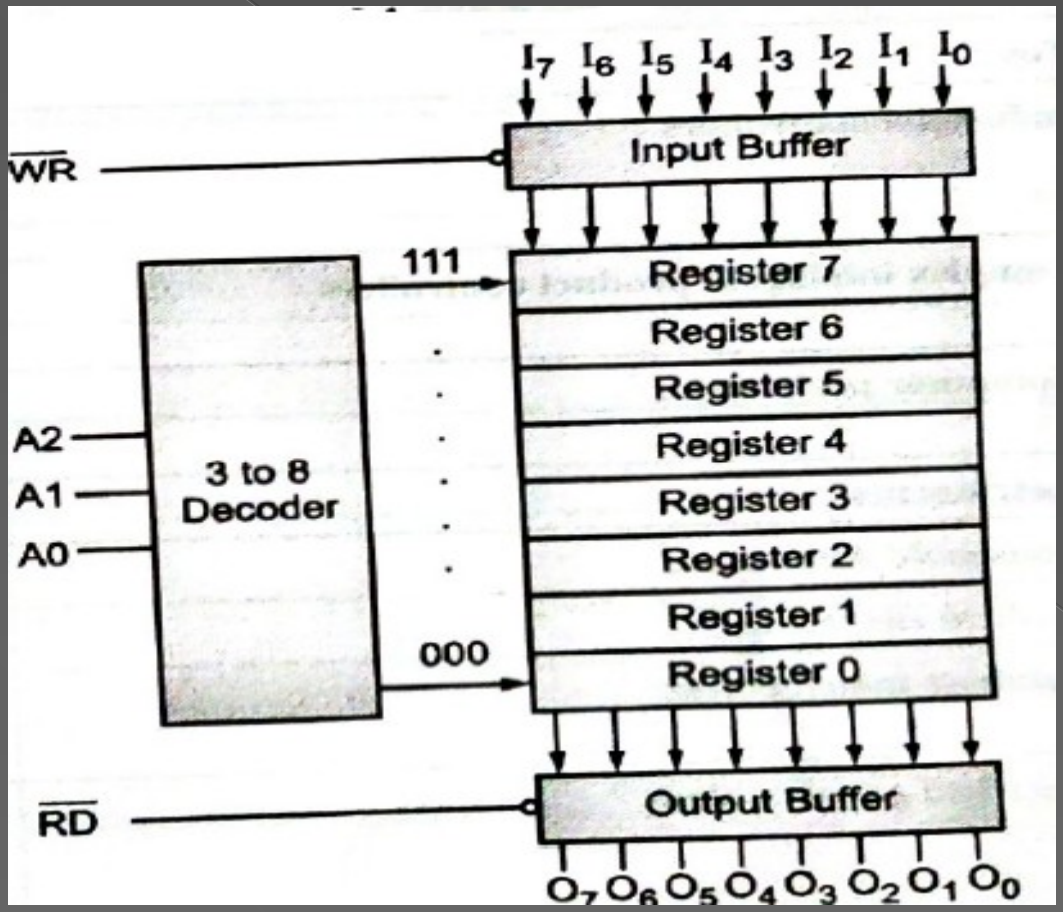
Cache memory



Digital circuit using decoder and registers for memory operations

- Fig shows the diagram of a 8x8 bit memeory.
- For storing 8 bytes ,it uses 8 registers

Digital circuit using decoder and registers for memory operations



- For performing read /write operation on aspecific memory location
- 3:8 decoder used selecting one of the eight memory location from 000 to 111.
- Uses address lines

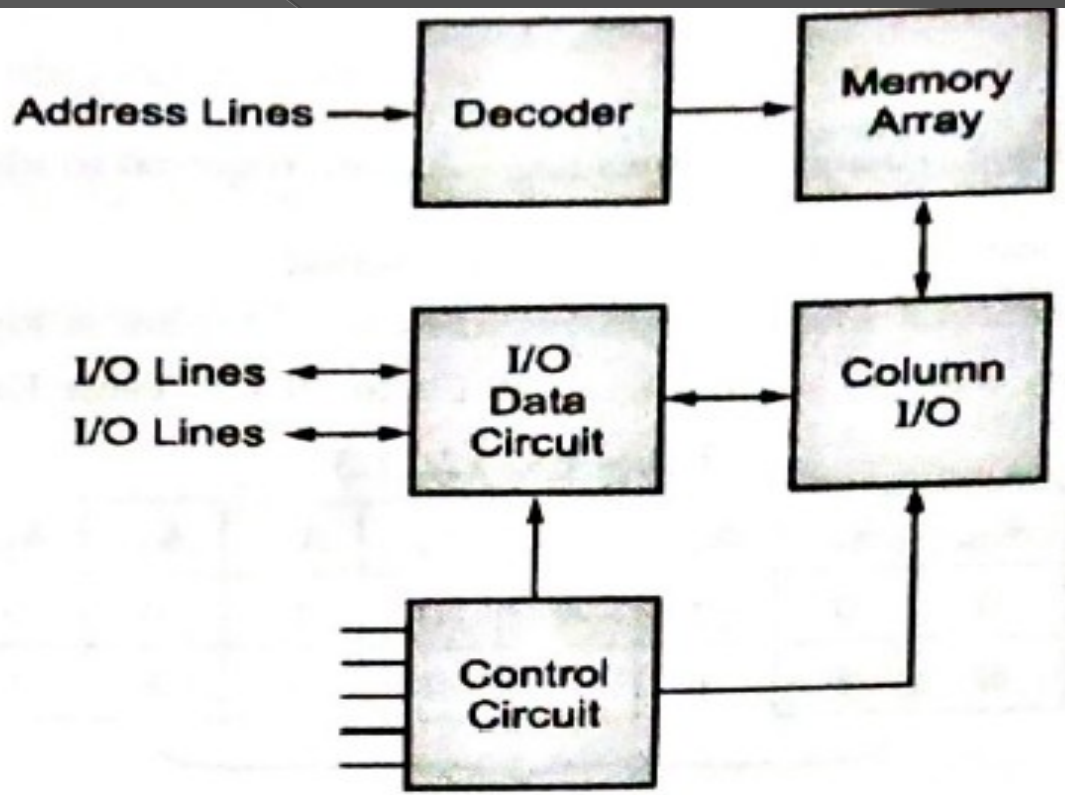


Fig. 12.12.2 : SRAM for memory operations

Applications

- Industrial Controller
- Computer Peripheral
- Instruments
- Household
- Medical
- Database Management
- Electrical Power