CSE-3103: Microprocessor and Microcontroller

Dept. of Computer Science and Engineering
University of Dhaka

Prof. Sazzad M.S. Imran, PhD
Dept. of Electrical and Electronic Engineering
sazzadmsi.webnode.com

Basic Architecture of 8086

8086 microprocessor →

2 separate functional units.

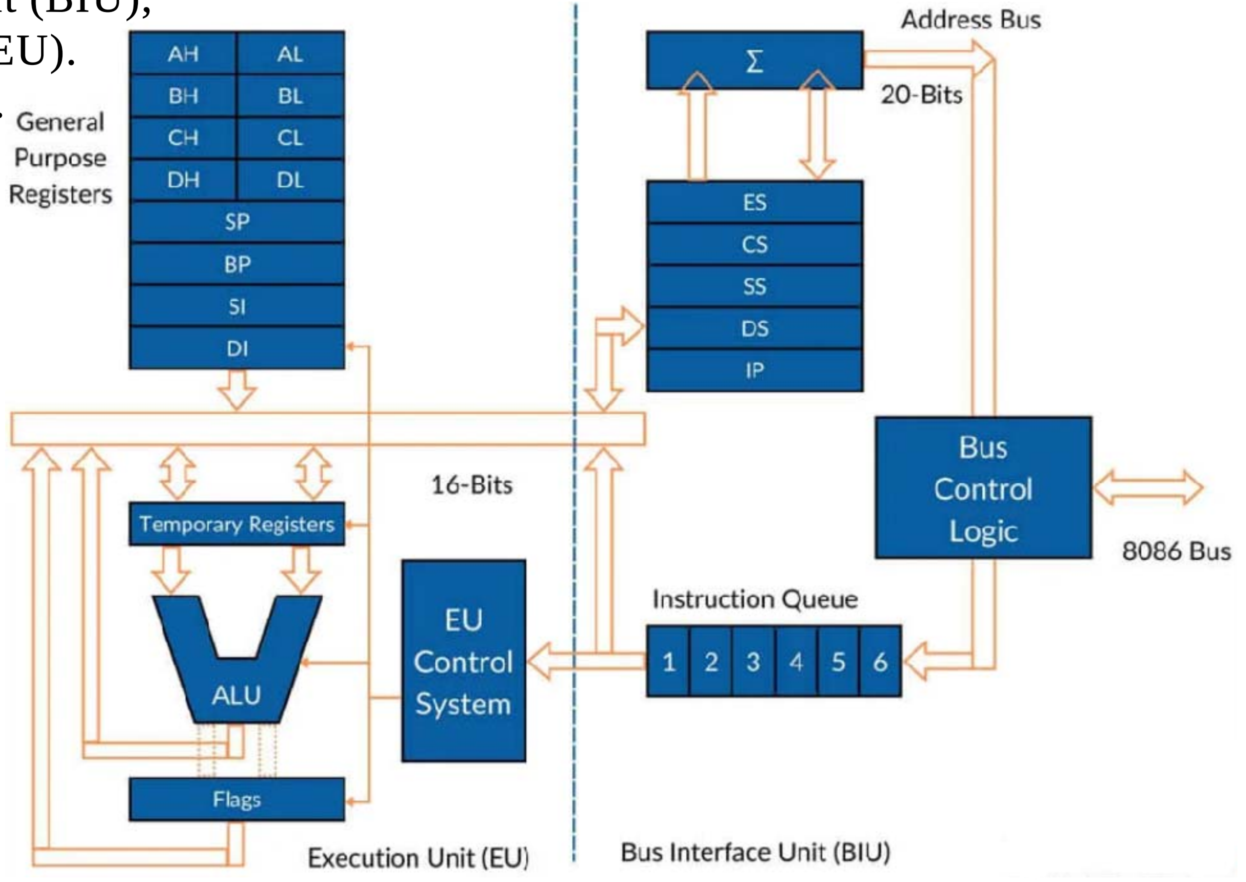
1) Bus Interface Unit (BIU),

2) Execution Unit (EU).

employs parallel processing.

BIU → segment registers,
instruction pointer,
address generation and
bus control logic block,
instruction queue.

EU → general purpose registers,
ALU,
control unit,
instruction register,
flag (or status) register.



Basic Architecture of 8086

Main jobs of BIU →

external bus operations.

instruction fetching,

reading/writing of data/operands for memory,

inputting/outputting of data for peripheral devices.

filling instruction queue.

address generation.

Main jobs of EU →

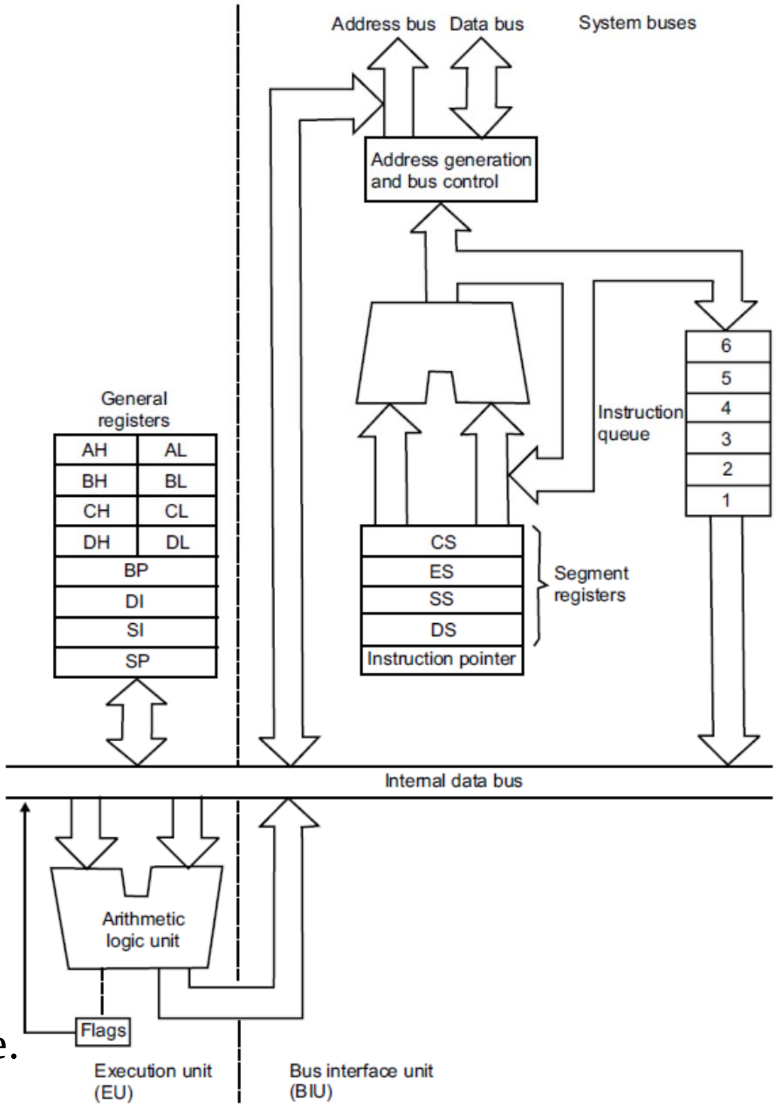
decoding/execution of instructions.

accepts instructions from instruction queue,
data from → general purpose registers,
or memory.

generates operand addresses, hands them to BIU.

tests and updates status of flags in control register when executing instructions.

waits for instructions from empty instruction queue.



Operations of Instruction Queue

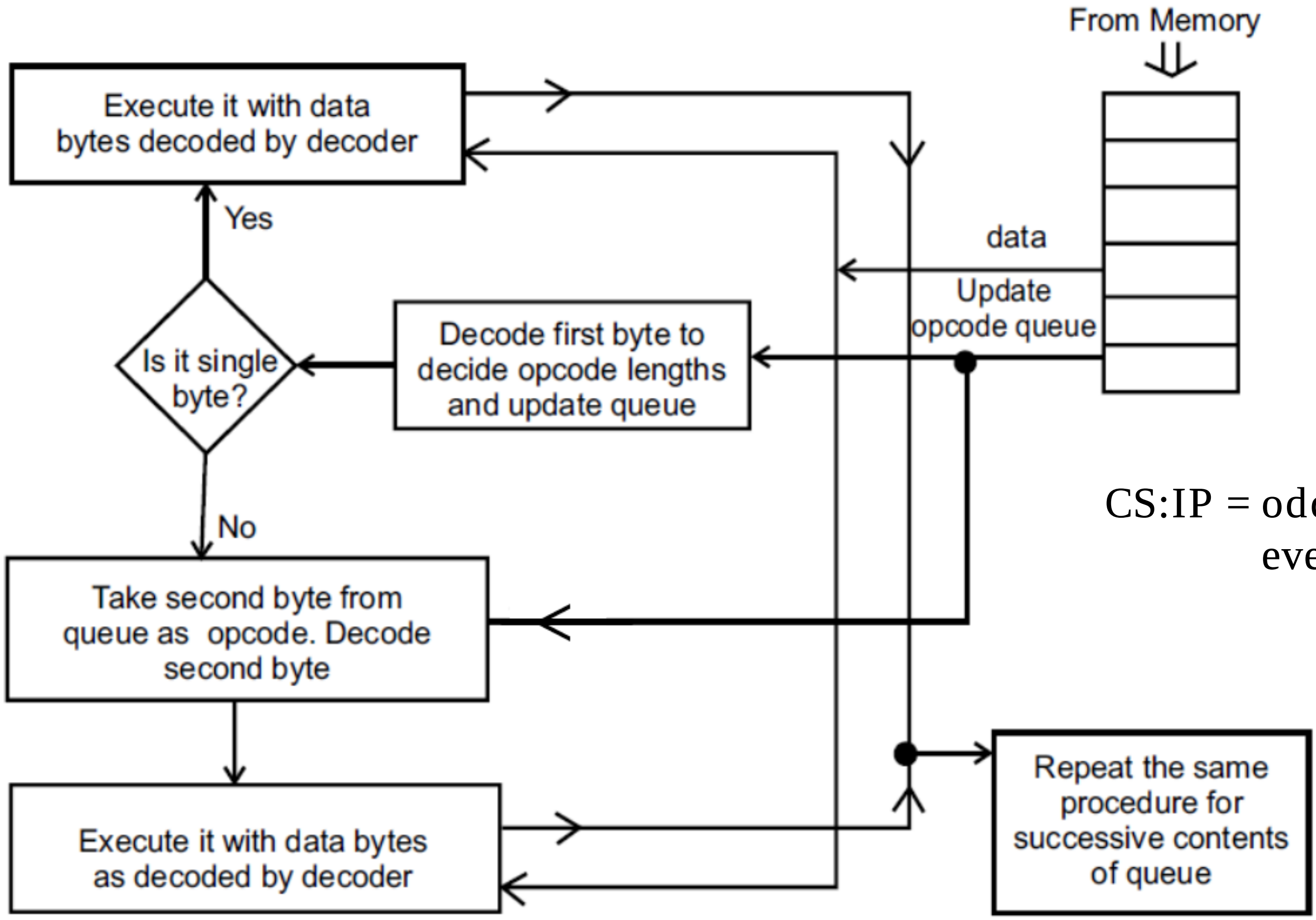
Pipelining procedure saves time →

Clock cycle	CC1	CC2	CC3	CC4	CC5	CC6	CC7	CC8	CC9
Without pipelining	IF1	ID1	IEX1	IF2	ID2	IEX2	IF3	ID3	IEX3
With pipelining	IF1	ID1	IEX1						
		IF2	ID2	IEX2					
			IF3	ID3	IEX3				
	Time (clock cycles) required for execution of three instructions without pipelining								
	Time (clock cycles) required for execution of three instructions with pipelining					Saved clock cycles (times)			

IF = Instruction Fetch
ID = Instruction Decode
IEX = Instruction Execution

Operations of Instruction Queue

Queue operation →


From Memory
Execute it with data bytes decoded by decoder
Yes
data
Update opcode queue
Is it single byte?
Decode first byte to decide opcode lengths and update queue
No
Take second byte from queue as opcode. Decode second byte
Execute it with data bytes as decoded by decoder
Repeat the same procedure for successive contents of queue

CS:IP = odd → 1 byte instruction
even → 2 bytes instruction

Registers of 8086

Fourteen 16-bit registers →

Register groups →
- data group →
  - AX, BX, CX, DX.
  - use bytewise or wordwise.
  - source or destination of operand.
- pointers and index group →
  - SP, BP, SI, DI, IP.
- segment group →
  - ES, CS, DS, SS.
- status and control flag group →
  - single 16-bit flag register.

registers for specific operations →
- CX = count register in string operations.
- DX = hold address of I/O port.
- AX = hold data for I/O operations.
- BX = hold offset address.

	16-bit		
	8-bit	8-bit	
AX: Accumulator	AH	AL	General purpose registers
BX: Base	BH	BL	
CX: Count	CH	CL	
DX: Data	DH	DL	
Stack Pointer	SP		Pointer registers
Base Pointer	BP		
Instruction Pointer	IP		
Source Index	SI		Index registers
Destination Index	DI		
Status and Control	FLAGS		Status registers
Code Segment	CS		Segment registers
Data Segment	DS		
Stack Segment	SS		
Extra Segment	ES		

CPU Registers

64-bit					
32-bit	32-bit				
	16-bit	16-bit			
		8-bit	8-bit		
RAX	EAX	AH	AL	AX: Accumulator
RBX	EBX	BH	BL	BX: Base
RCX	ECX	CH	CL	CX: Count
RDX	EDX	DH	DL	DX: Data
RSP	ESP	SP		Stack Pointer
RBP	EBP	BP		Base Pointer
RIP	EIP	IP		Instruction Pointer
RSI	ESI	SI		Source Index
RDI	EDI	DI		Destination Index
	EFLAGS	FLAGS		Status and Control
		CS		Code Segment
		DS		Data Segment
		SS		Stack Segment
		ES		Extra Segment

R: Register

E: Extended