Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

MICROPROCESSOR & MICROCONTROLLERS

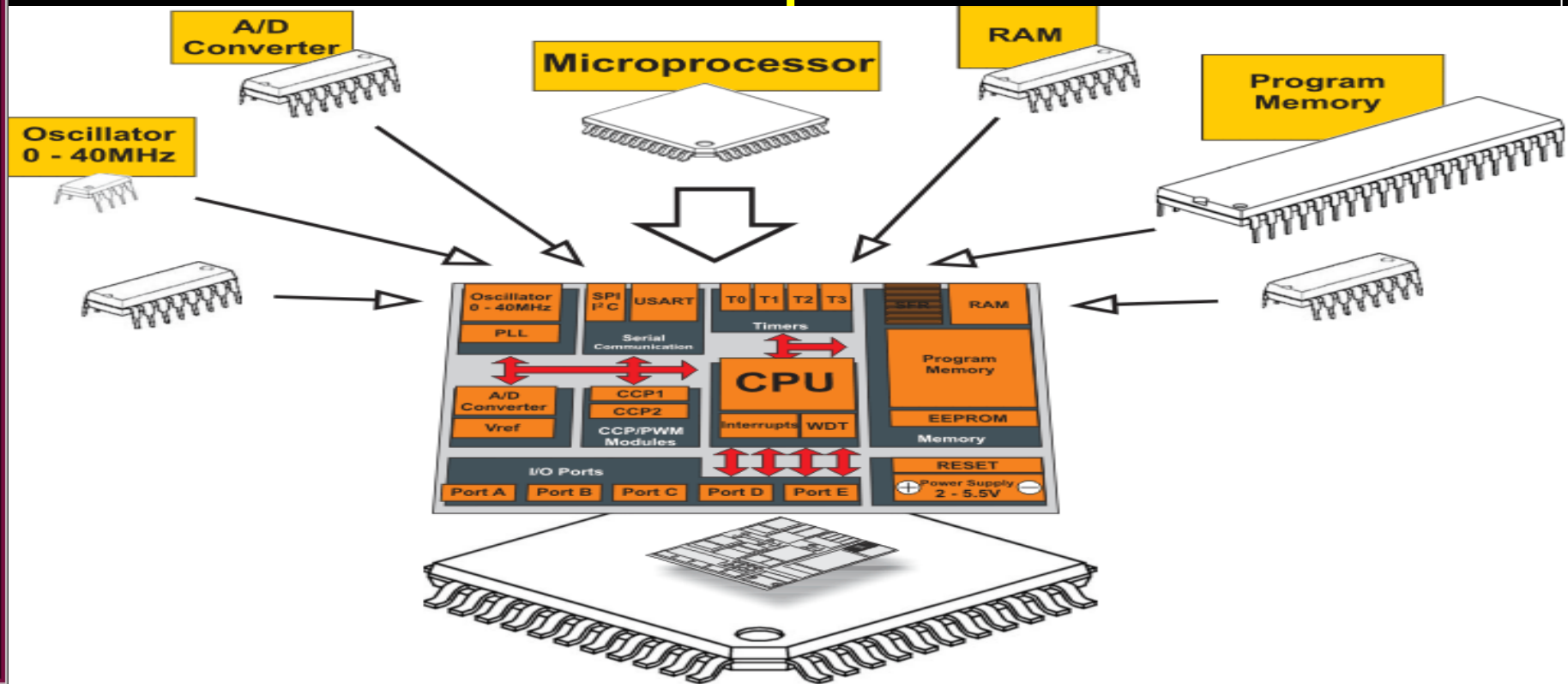
053 EC 65

Unit 1 – 8085 MICROPROCESSOR

Lecture 1-4 : 8085 Microprocessor Instruction Set

By DICKSON NKONGO, MSc.

EECE Department



Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

Lecture 1-4: Outline

1. 8085 Instruction Format
2. 8085 Instruction Set
3. Classification of Instruction Set

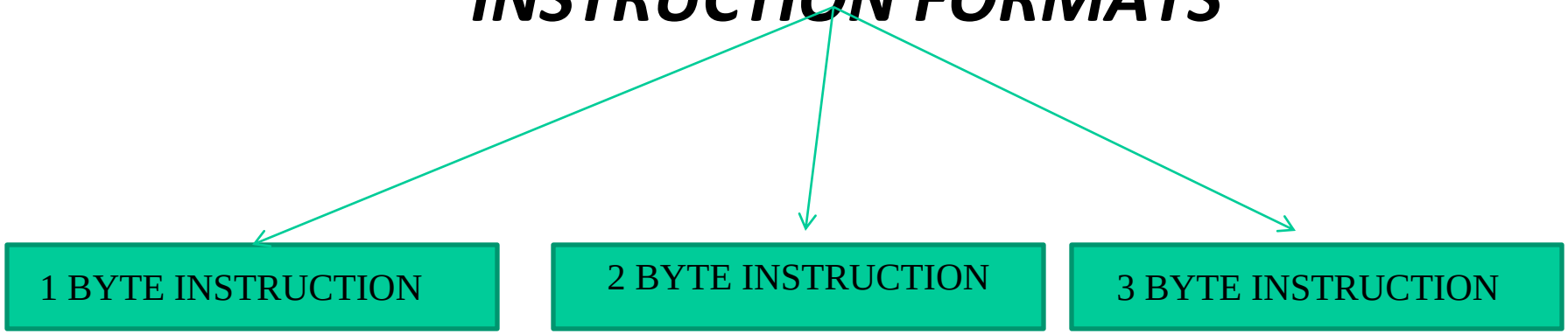
Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

8085 INSTRUCTION FORMAT

MP & MC

Lecture 4 – 8085 Microprocessor Instruction Set

INSTRUCTION FORMATS



MOV A,B --- 78H

MVI B, 02 --- 06H 02

JMP 6200H --- C3H 00 62

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

Instruction Formats

Instruction set of 8085A consists of one, two and three byte instructions.

- ***1 Byte Instruction***
- ***2Bytes Instruction***
- ***3 Bytes Instruction***

The first byte is always the opcode; in two byte instructions the second byte is usually data; in three byte instructions the last two byte present address or 16-bit data

MP & MC APPLICATIONS
Lecture 4 – 8085 Microprocessor Instruction Set

Instruction Formats

Instruction set of 8085A consists of one, two and three byte instructions.

- ***1.One Byte Instruction***

FORMAT | Opcode |

For example: MOV B, C whose opcode is 41H which is one byte. This instruction copies the contents of C register in B register

2. Two byte instruction:

FORMAT | Opcode | Operand |

For example: MVI B, 08H. The opcode for this instruction is 06H and is always followed by a byte data (08H in this case). This instruction is a two byte instruction which copies immediate data into B register

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

Instruction Formats

Instruction set of 8085A consists of one, two and three byte instructions.

- ***3. Three Byte Instruction***

FORMAT: | Opcode | Operand | Operand |

For example: JMP 8200H. The opcode for this instruction is C3H and is always followed by 16 bit address (8200H in this case).

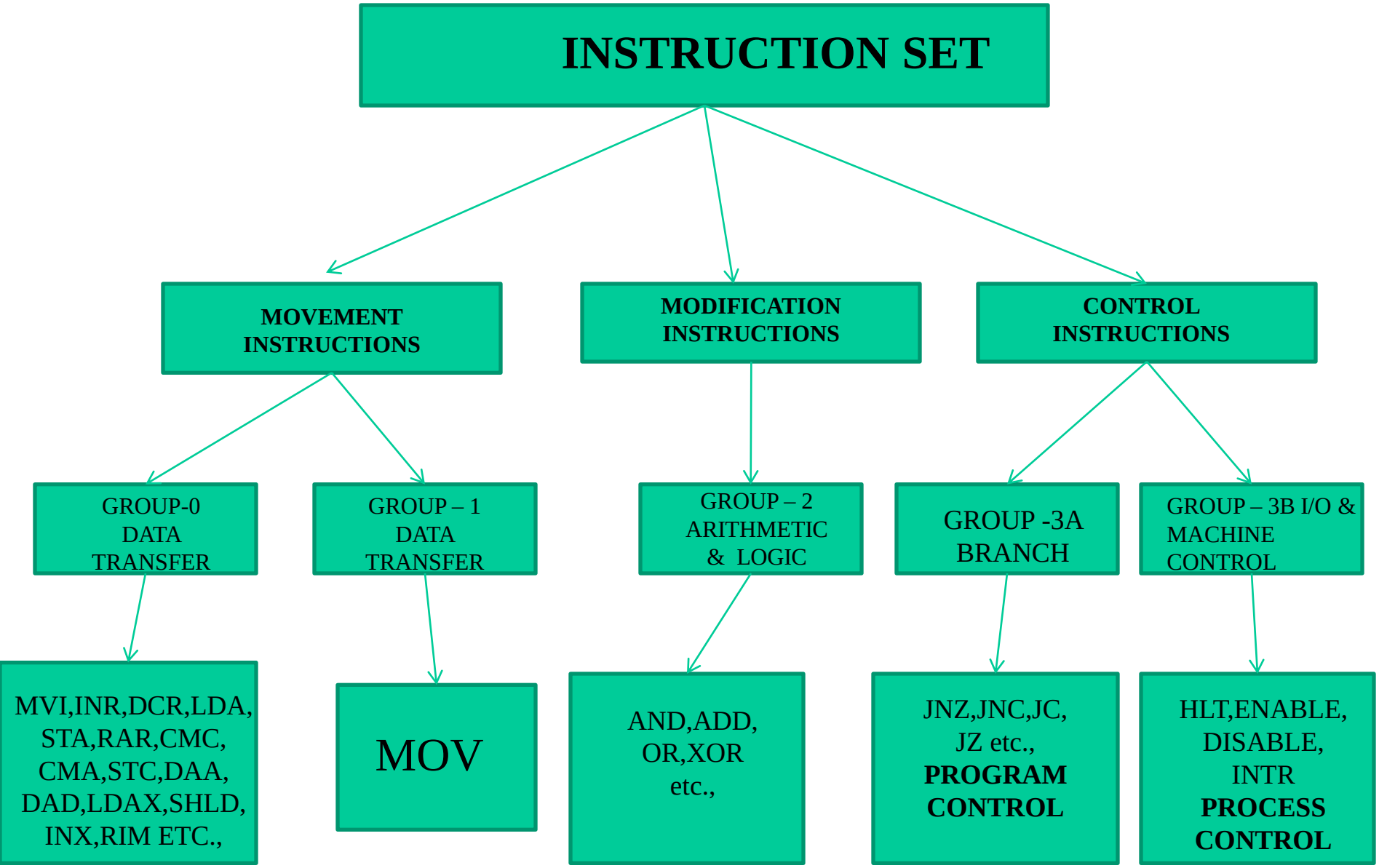
This instruction is a three byte which loads 16 bit address into program counter

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

8085 INSTRUCTION SET

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set



Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

BYTE ORGANIZATION

GROUP - 0

0	0	R	R	R	I0	I0	I0

GROUP - 1

0	1	R	R	R	S	S	S

GROUP - 2

1	0	A1	A1	A1	S	S	S

GROUP - 3

1	1	Cb	Cb	Cb	B0	B0	B0

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

Code For Receiving And Sending Registers/Pairs

REGISTERS	ADDRESS CODE	REGISTERS	ADDRESS CODE
B	000	B – C	00
C	001		
D	010	D – E	01
E	011		
H	100	H – L	10
L	101		
M	110	SP	11
A	111		

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

Information Operations (I0 I0 I0)

ADDRESS			OPERATION
I0	I0	I0	
0	0	0	NOT USED
0	0	1	IMMEDIATE OPERATION REGISTER PAIR
0	1	0	LOAD / STORE
0	1	1	INCREMENT/ DECREMENT REGISTER PAIR
1	0	0	INCREMENT SINGLE REGISTER
1	0	1	DECREMENT SINGLE REGISTER
1	1	0	IMMEDIATE OPERATION ON SINGLE REGISTER
1	1	1	REGISTER SHIFTING

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

Arithmetic And Logical Operations (A1 A1 A1)

ADDRESS			OPERATION
A1	A1	A1	
0	0	0	ADD
0	0	1	ADD WITH CARRY (ADC)
0	1	0	SUBTRACT (SUB)
0	1	1	SUBTRACT WITH BORROW (SBB)
1	0	0	LOGICAL AND
1	0	1	EXCLUSIVE OR (X-OR)
1	1	0	LOGICAL OR (OR)
1	1	1	COMPARE

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

Conditions Of Branch (Cb Cb Cb)

ADDRESS			OPERATION
Cb	Cb	Cb	
0	0	0	IF NOT ZERO (JNZ)
0	0	1	IF ZERO (JZ)
0	1	0	IF NO CARRY(JNC)
0	1	1	IF CARRY (JC)
1	0	0	IF ODD PARITY (JPO)
1	0	1	IF EVEN PARITY (JPE)
1	1	0	WAS IT POSITIVE (JP)
1	1	1	WAS IT NEGATIVE (JM)

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

Branch Operations (BO BO BO)

ADDRESS			OPERATION
BO	BO	BO	
0	0	0	CONDITIONAL RETURN
0	0	1	SIMPLE RETURN
0	1	0	CONDITIONAL JUMP
0	1	1	UNCONDITIONAL JUMP
1	0	0	CONDITIONAL CALL
1	0	1	SIMPLE CALL
1	1	0	SPECIAL A/L OPERATIONS
1	1	1	SPECIAL UNCONDITIONAL JUMPS

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

MVI B, BYTE

0	0	R	R	R	I0	I0	I0
0	0	0	0	0	1	1	0

MOV B,C

0	1	R	R	R	S	S	S
0	1	0	0	0	0	0	1

ADD B

1	0	A1	A1	A1	S	S	S
1	0	0	0	0	0	0	0

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

CLASSIFICATION OF INSTRUCTION SET

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

Classification Of Instruction Set

There are 5 categories:

- **(1) Data Transfer Instruction,**
- **(2) Arithmetic Instructions,**
- **(3) Logical Instructions,**
- **(4) Branching Instructions,**
- **(5) Control Instructions,**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

(1) Data Transfer Instructions

- ***MOV Rd, Rs***
- ***MOV M, Rs***
- ***MOV Rd, M***
- **This instruction copies the contents of the source register into the destination register.**
- **The contents of the source register are not altered.**
- **Example: MOV B,A or MOV M,B or MOV C,M**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

A	20	B	

MOV B, A

AFTER EXECUTION

A	20	B	20

A		F	
B	30	C	
D		E	
H	20	L	50

MOV M, B

A		F	
B	30	C	
D		E	
H	20	L	50

30

MEMORY

A		F	
B		C	
D		E	
H	20	L	50

40

MOV C, M

MEMORY

A		F	
B		C	40
D		E	
H	20	L	50

40

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

(2) Data Transfer Instructions

- ***MVI R, Data(8-bit)***
- ***MVI M, Data(8-bit)***

- ***The 8-bit immediate data is stored in the destination register (R) or memory (M), R is general purpose 8 bit register such as A, B, C, D, E, H and L.***

- ***Example: MVI B, 60H or MVI M, 40H***

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

A		F	
B		C	
D		E	
H		L	

MVI B,60H

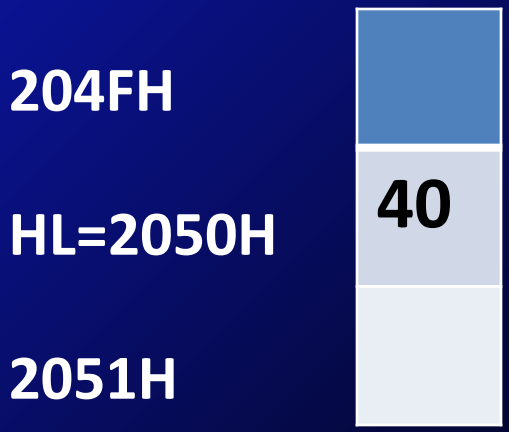
AFTER EXECUTION

A		F	
B	60	C	
D		E	
H		L	

BEFORE EXECUTION

MVI M,40H

AFTER EXECUTION



Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

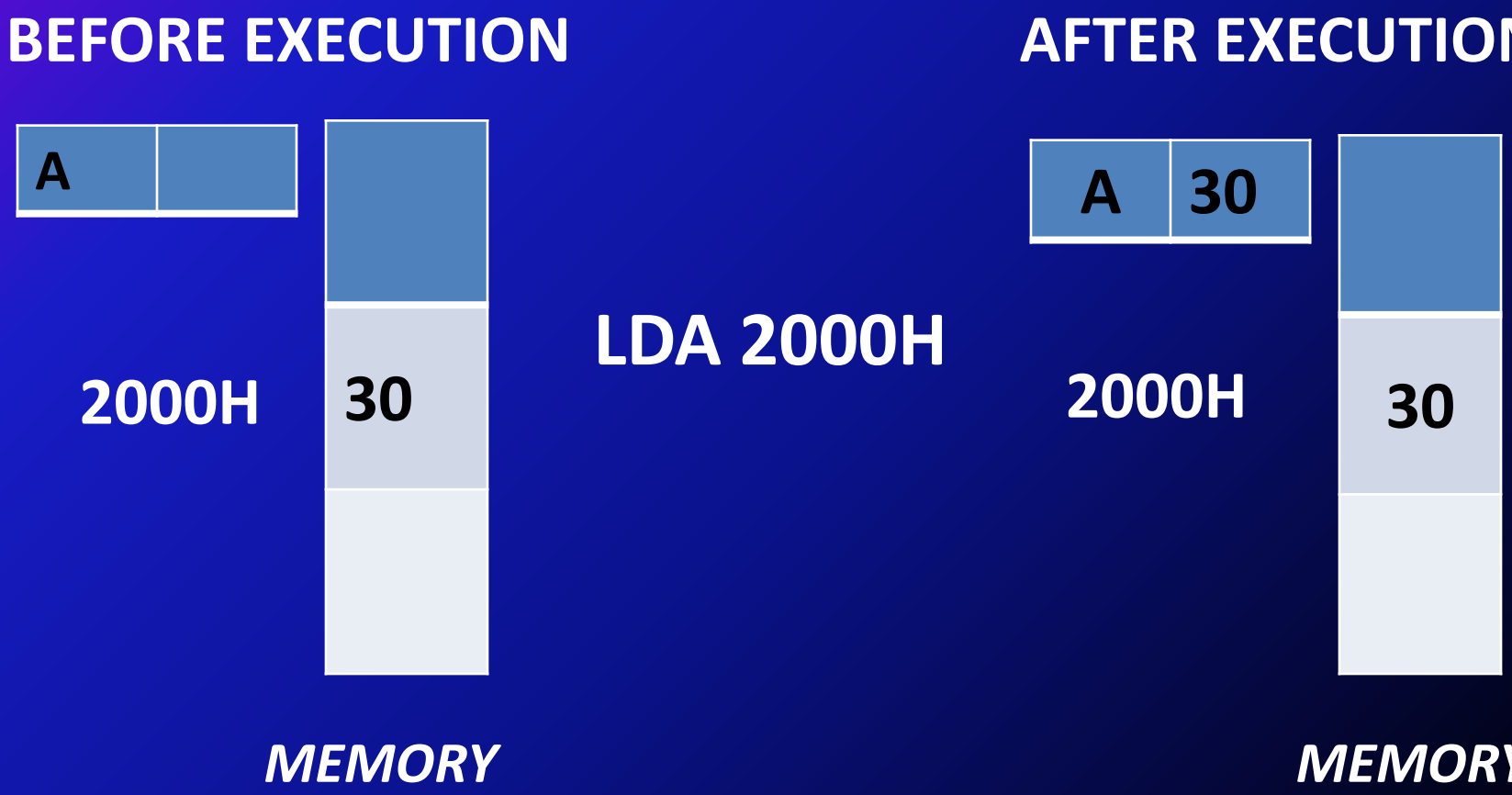
(3) Data Transfer Instructions

- **LDA 16-bit address**
- **The contents of a memory location, specified by a 16-bit address in the operand, are copied to the accumulator (A).**
- **The contents of the source are not altered.**

- **Example: LDA 2000H**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set



Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(4) Data Transfer Instructions

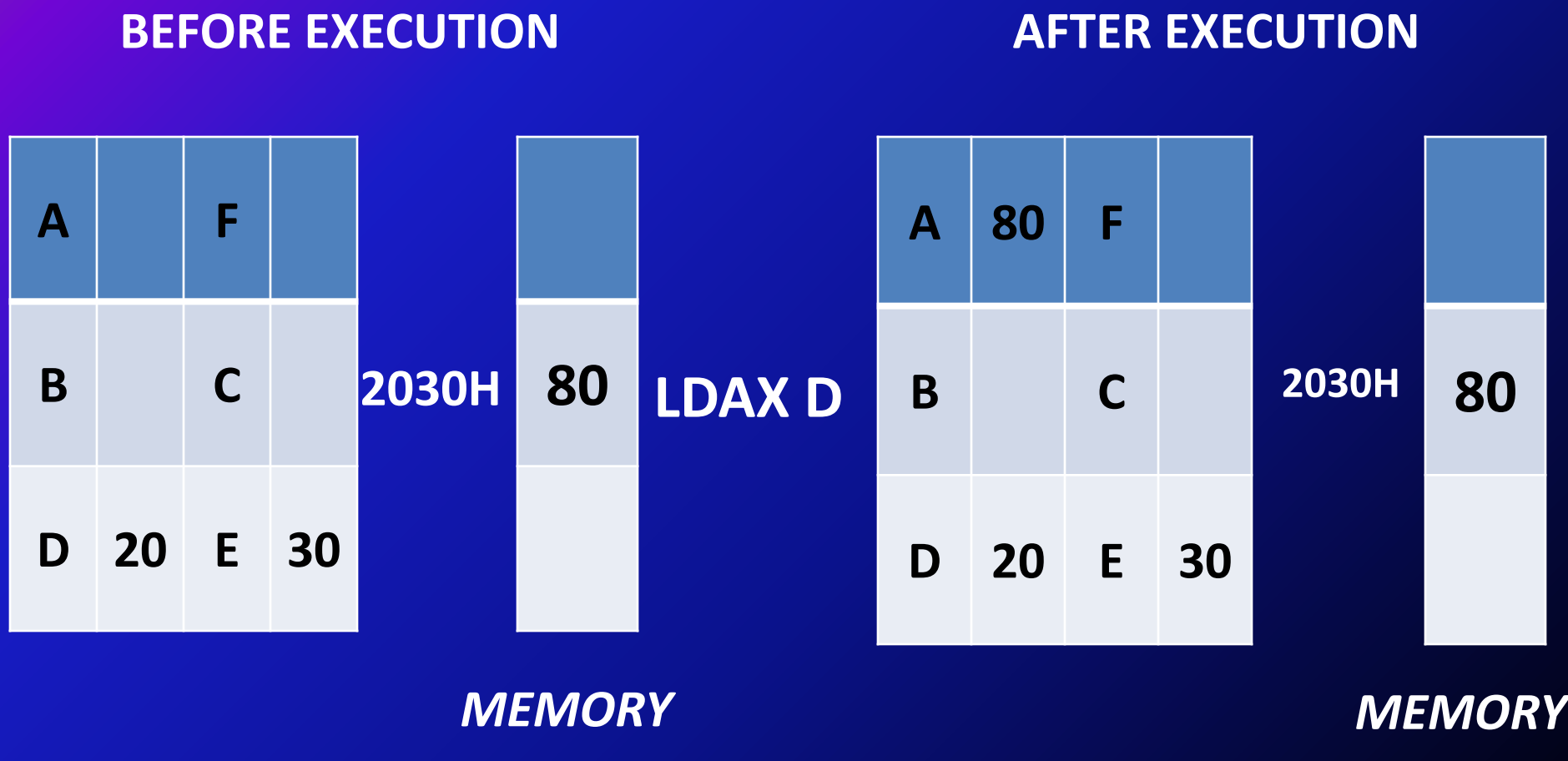
LDAX Register Pair

- Load accumulator (A) with the contents of memory location whose address is specified by BC or DE or register pair.
- The contents of either the register pair or the memory location are not altered.

- Example: LDAX D

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set



Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(5) Data Transfer Instructions

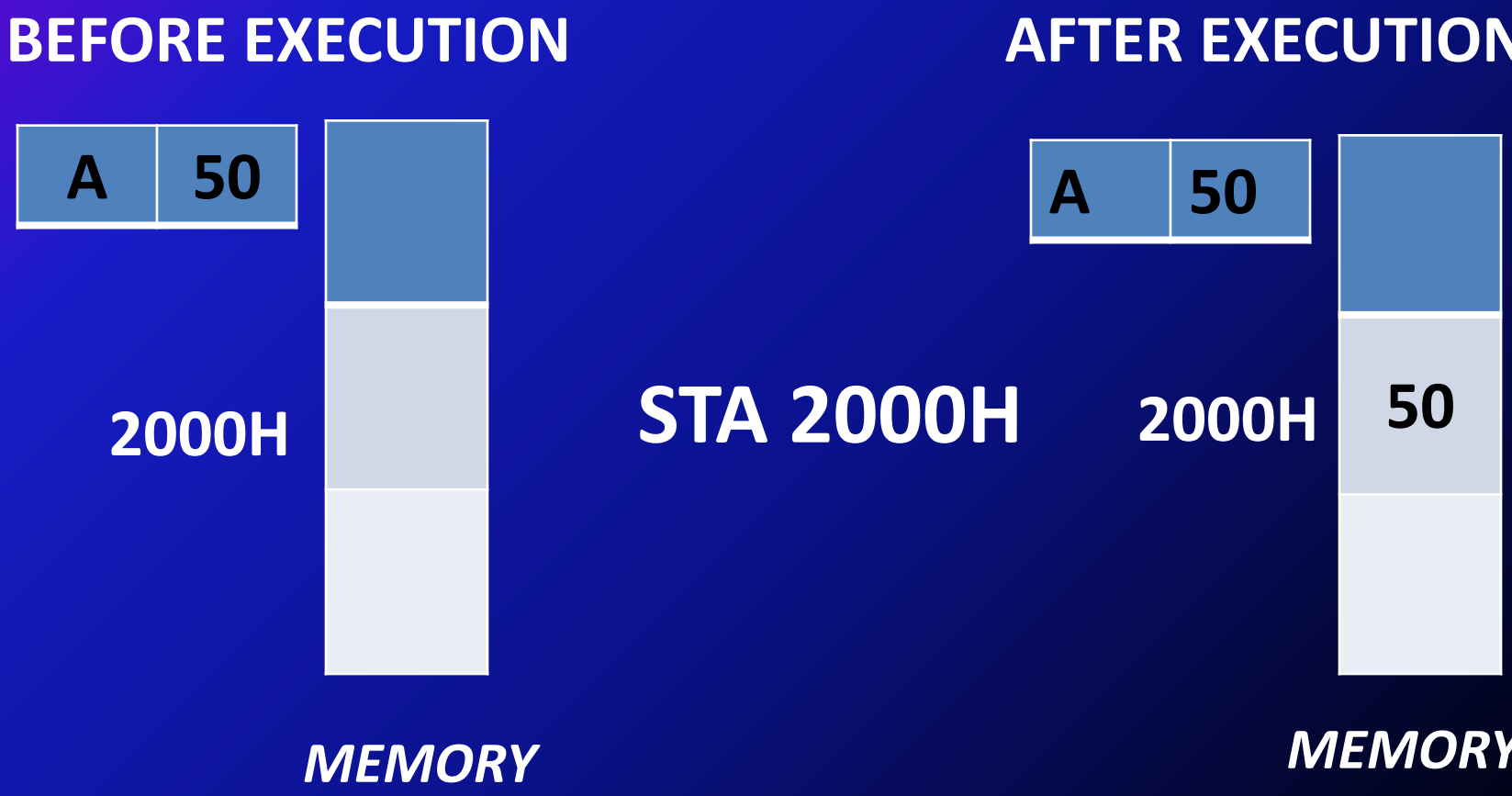
STA 16-bit address

- **The contents of accumulator are copied into the memory location i.e. address specified by the operand in the instruction.**

- **Example: STA 2000 H**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set



Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(6) Data Transfer Instructions

STAX Register Pair

- **Store the contents of accumulator (A) into the memory location whose address is specified by BC Or DE register pair.**

- **Example: STAX B**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

A	50	F	
B	10	C	20
D		E	

1020H

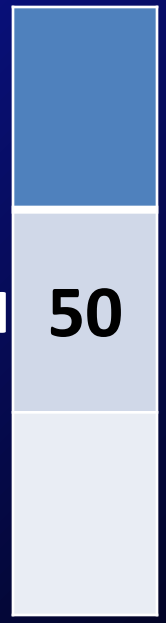
MEMORY

STAX B

AFTER EXECUTION

A	50	F	
B	10	C	20
D		E	

1020H



MEMORY

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(8) Data Transfer Instructions

XCHG

- The contents of register H are exchanged with the contents of register D.
- The contents of register L are exchanged with the contents of register E.

- Example: XCHG

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

D	20	E	40
H	70	L	80

XCHG

AFTER EXECUTION

D	70	E	80
H	20	L	40

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(12) Data Transfer Instructions

IN 8-bit port address

- ***Copy data to accumulator from a port with 8-bit address.***
- **The contents of I/O port are copied into accumulator.**

- **Example: IN 80 H**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

PORT 80H		10

A	

IN 80H

AFTER EXECUTION

PORT 80H		10

A	10

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(13) Data Transfer Instructions

- ***OUT 8-bit port address***

- ***Copy data from accumulator to a port with 8-bit address***
- **The contents of accumulator are copied into the I/O port.**

- **Example: OUT 50 H**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

PORT 50H	
	10

A	40

OUT 50H

AFTER EXECUTION

PORT 50H	
	40

A	40

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

B. Arithmetic Instructions

These instructions perform the operations like:

- ***Addition***
- ***Subtraction***
- ***Increment***
- ***Decrement***

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(1) Arithmetic Instructions

- ***ADD R***
- ***ADD M***

- **The contents of register or memory are added to the contents of accumulator.**
- **The result is stored in accumulator.**
- **If the operand is memory location, its address is specified by H-L pair.**

- **Example: ADD C or ADD M**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

A	20		
B		C	30
D		E	
H		L	

ADD C

A=A+R

AFTER EXECUTION

A	50		
B		C	30
D		E	
H		L	

BEFORE EXECUTION

A	20		
B		C	
D		E	
H	20	L	50

2050

ADD M

A=A+M

10

AFTER EXECUTION

A	30		
B		C	
D		E	
H	20	L	50

2050

10

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(2) Arithmetic Instructions

- *ADC R*
- *ADC M*
- *The contents of register or memory and Carry Flag (CY) are added to the contents of accumulator.*
- **The result is stored in accumulator.**
- *If the operand is memory location, its address is specified by H-L pair*. **All flags are modified to reflect the result of the addition.**

- **Example: ADC C or ADC M**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

CY	1

A	50

B		C	20
D		E	
H		L	

ADC C

A=A+R+CY

AFTER EXECUTION

CY	0

A	71

B		C	20
D		E	
H		L	

BEFORE EXECUTION

CY	1
A	20

2050H

30

H	20	L	50

ADC M

A=A+M+CY

AFTER EXECUTION

CY	0
A	51

2050H

30

H	20	L	50

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(3) Arithmetic Instructions

- ***ADI 8-bit data***

- ***The 8-bit data is added to the contents of accumulator.***
- **The result is stored in accumulator.**

- **Example: ADI 10 H**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

A	50

ADI 10H

A=A+DATA(8)

AFTER EXECUTION

A	60

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(4) Arithmetic Instructions

- ***ACI 8-bit data***

- ***The 8-bit data and the Carry Flag (CY) are added to the contents of accumulator.***
- **The result is stored in accumulator.**

- **Example: ACI 20 H**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

CY	1
A	30

ACI 20H

A=A+DATA(8)+CY

AFTER EXECUTION

CY	0
A	51

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(6) Arithmetic Instructions

- ***SUB R***
- ***SUB M***

- ***The contents of the register or memory location are subtracted from the contents of the accumulator.***
- **The result is stored in accumulator.**
- **If the operand is memory location, its address is specified by H-L pair.**

- **Example: SUB B or SUB M**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

A	50		
B	30	C	
D		E	
H		L	

SUB B

A=A-R

AFTER EXECUTION

A	20		
B	30	C	
D		E	
H		L	

BEFORE EXECUTION

A	50

H	10	L	20

1020H

10

SUB M

A=A-M

AFTER EXECUTION

A	40

H	10	L	20

1020H

10

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(7) Arithmetic Instructions

- *SBB R*
- *SBB M*

- *The contents of the register or memory location and Borrow Flag (i.e.CY) are subtracted from the contents of the accumulator.*
- **The result is stored in accumulator.**
- **If the operand is memory location, its address is specified by H-L pair.**

- **Example: SBB C or SBB M**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

CY	1

A	40		
B		C	20
D		E	
H		L	

SBB C

A=A-R-CY

AFTER EXECUTION

CY	0

A	19		
B		C	20
D		E	
H		L	

BEFORE EXECUTION

CY	1
A	50

2050H

H	20	L	50

10

SBB M

A=A-M-CY

AFTER EXECUTION

CY	0
A	39

2050H

H	20	L	50

10

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(8) Arithmetic Instructions

- *SUI 8-bit data*
- *OPERATION: A=A-DATA(8)*
- *The 8-bit immediate data is subtracted from the contents of the accumulator.*
- **The result is stored in accumulator.**
- **Example: SUI 45 H**

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(9) Arithmetic Instructions

- **SBI 8-bit data**

- ***The 8-bit data and the Borrow Flag (i.e. CY) is subtracted from the contents of the accumulator.***
- **The result is stored in accumulator.**

- **Example: SBI 20 H**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

CY	1
A	50

SBI 20H

A=A-DATA(8)-CY

AFTER EXECUTION

CY	0
A	29

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(10) Arithmetic Instructions

- **INR R**
- **INR M**

- ***The contents of register or memory location are incremented by 1*.**
- **The result is stored in the same place.**
- **If the operand is a memory location, its address is specified by the contents of H-L pair.**

- **Example: INR B or INR M**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

A	

B	10	C	
D		E	
H		L	

INR B

R=R+1

AFTER EXECUTION

A	

B	11	C	
D		E	
H		L	

BEFORE EXECUTION

H		L	
	20		50

2050H

30

INR M

M=M+1

AFTER EXECUTION

H		L	
	20		50

31

2050H

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(11) Arithmetic Instructions

- **INX Rp**
- ***This instruction increments the contents of register pair by 1.***
- **The result is stored in the same place.**
- **Example: INX H**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

SP			
B		C	
D		E	
H	10	L	20

INX H

RP=RP+1

AFTER EXECUTION

SP			
B		C	
D		E	
H	10	L	21

QN: What is the next address after 11FF?

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(12) Arithmetic Instructions

- **DCR R**
- **DCR M**

- ***The contents of register or memory location are decremented by 1*.**
- **The result is stored in the same place.**
- **If the operand is a memory location, its address is specified by the contents of H-L pair.**
- **Example: DCR E or DCR M**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

A			
B		C	
D		E	20
H		L	

DCR E

R=R-1

AFTER EXECUTION

A			
B		C	
D		E	1F
H		L	

BEFORE EXECUTION

H	20	L	50

2050H

21

DCR M

M=M-1

H	20	L	50

AFTER EXECUTION

2050H

20

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(13) Arithmetic Instructions

DCX Rp

- *This instruction decrements the contents of register pair by 1*.
- **The result is stored in the same place.**

- **Example: DCX D**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

SP			
B		C	
D	10	E	20
H		L	

DCX D

RP=RP-1

AFTER EXECUTION

SP			
B		C	
D	10	E	1F
H		L	

QN: What is the address before 10F0?

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(1) Logical Instructions

- **ANA R**
- **ANA M**
- ***AND specified data in register or memory with accumulator*.**
- **Store the result in accumulator (A).**
- Example: **ANA B, ANA M**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

A	AA

B	0F	C	
D		E	
H		L	

1010 1010=AAH
0000 1111=0FH

0000 1010=0AH

ANA B
A=A and R

AFTER EXECUTION

A	0A

B	0F	C	
D		E	
H		L	

BEFORE EXECUTION

A	55

2050H

H	20	L	50

B3

0101 0101=55H
1011 0011=B3H

0001 0001=11H

ANA M
A=A and M

A	11

2050H

H	20	L	50

B3

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(2) Logical Instructions

- **ANI 8-bit data**
- **AND 8-bit data with accumulator (A).**
- **Store the result in accumulator (A)**

- Example: **ANI 3FH**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

AFTER EXECUTION

1011 0011=B3H

0011 1111=3FH

0011 0011=33H

ANI 3FH

A=A and DATA(8)

A	B3

A	33

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(3) Logical Instructions

- **XRA Register (8-bit)**

- **XOR specified register with accumulator.**
- **Store the result in accumulator.**

- Example: **XRA C**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

CY		AC	

A	AA		
B		C	2D
D		E	
H		L	

1010 1010=AAH
0010 1101=2DH

1000 0111=87H

XRA C
A=A xor R

AFTER EXECUTION

CY	0	AC	0

A	87		
B		C	2D
D		E	
H		L	

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(4) Logical Instructions

- **XRA M**
- ***XOR data in memory (memory location pointed by H-L pair) with Accumulator.***
- **Store the result in Accumulator.**
- **Example: XRA M**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

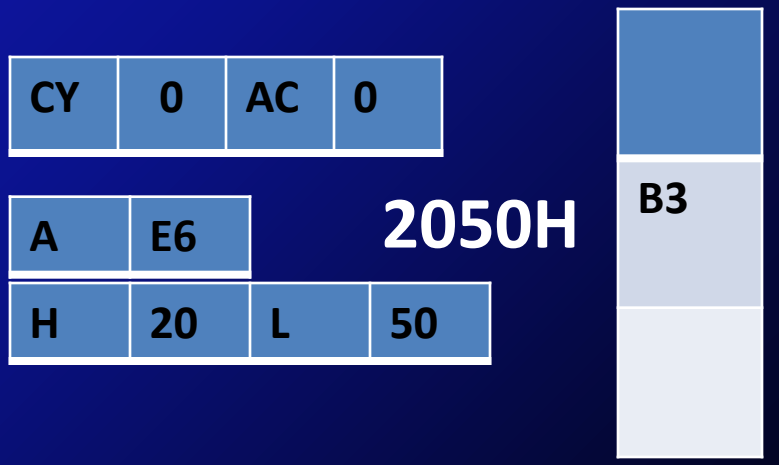
BEFORE EXECUTION

0101 0101=55H
1011 0011=B3H

1110 0110=E6H

AFTER EXECUTION

XRA M
A=A xor M



Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(5) Logical Instructions

XRI 8-bit data

- *XOR 8-bit immediate data with accumulator (A).*
- **Store the result in accumulator.**

- **Example: XRI 39H**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

1011 0011=B3H
0011 1001=39H
1000 1010=8AH

BEFORE EXECUTION

CY		AC	

A	B3

XRI 39H

A=A xor DATA(8)

AFTER EXECUTION

CY	0	AC	0

A	8A

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(6) Logical Instructions

- **ORA Register**
- **OR specified register with accumulator (A).**
- **Store the result in accumulator.**

- **Example: ORA B**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

1010 1010=AAH
0001 0010=12H

1011 1010=BAH

BEFORE EXECUTION

CY		AC	

A	AA

B	12	C	
D		E	
H		L	

ORA B
A=A or R

AFTER EXECUTION

CY	0	AC	0

A	BA

B	12	C	
D		E	
H		L	

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(7) Logical Instructions

- **ORA M**

- *OR H-L pair (i.e. M) with accumulator (A).*
- **Store the result in accumulator.**

- **Example: ORA M**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

0101 0101=55H

1011 0011=B3H

1111 0111=F7H

BEFORE EXECUTION

CY		AC	

AFTER EXECUTION

CY	0	AC	0

ORA M

A=A or M

A	55

2050H

B3

H	20	L	50

A	F7

2050H

B3

H	20	L	50

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(8) Logical Instructions

ORI 8-bit data

- OR 8-bit data with accumulator (A).
- Store the result in accumulator.

- Example: ORI 08H

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

1011 0011=B3H
0000 1000=08H

1011 1011=BBH

ORI 08H
A=A or DATA(8)

BEFORE EXECUTION

CY		AC	

A	B3

AFTER EXECUTION

CY	0	AC	0

A	BB

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(9) Logical Instructions

- **CMP Register**
- **CMP M**

- ***Compare specified data in register or memory with accumulator (A).***
- **Store the result in accumulator.**

- **Example: CMP D or CMP M**

BEFORE EXECUTION

CY		Z	

A	B8

B		C	
D	B9	E	
H		L	

A>R: CY=0,Z=0
A=R: CY=0,Z=1
A<R: CY=1,Z=0

CMP D
A-R

AFTER EXECUTION

CY	0	Z	0

A	B8

B		C	
D	B9	E	
H		L	

BEFORE EXECUTION

CY		Z	

A	B8

2050H

H	20	L	50

B8

A>M: CY=0,Z=0
A=M: CY=0,Z=1
A<M: CY=1,Z=0

CMP M
A-M

AFTER EXECUTION

CY	0	Z	1

A	B8

2050H

H	20	L	50

B8

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(10) Logical Instructions

- **CPI 8-bit data**
- ***Compare 8-bit immediate data with accumulator (A).***
- **Store the result in accumulator.**

- **Example: CPI 30H**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

A>DATA: CY=0, Z=0

A=DATA: CY=0, Z=1

A<DATA: CY=1, Z=0

BEFORE EXECUTION

CY		Z	

A	BA

CPI 30H

A-Data

AFTER EXECUTION

CY	0	Z	0

A	BA

1011 1010=BAH

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(11) Logical Instructions

- **STC**

- ***It sets the carry flag to 1*.**
- **Example: STC**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

CY	0

STC

CY=1

AFTER EXECUTION

CY	1

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(12) Logical Instructions

- **CMC**

- ***It complements the carry flag.***
- **Example: CMC**

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

CY	1

CMC

AFTER EXECUTION

CY	0

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(13) Logical Instructions

- **CMA**

- ***It complements each bit of the accumulator.***
- **Example: CMA**

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(14) Logical Instructions

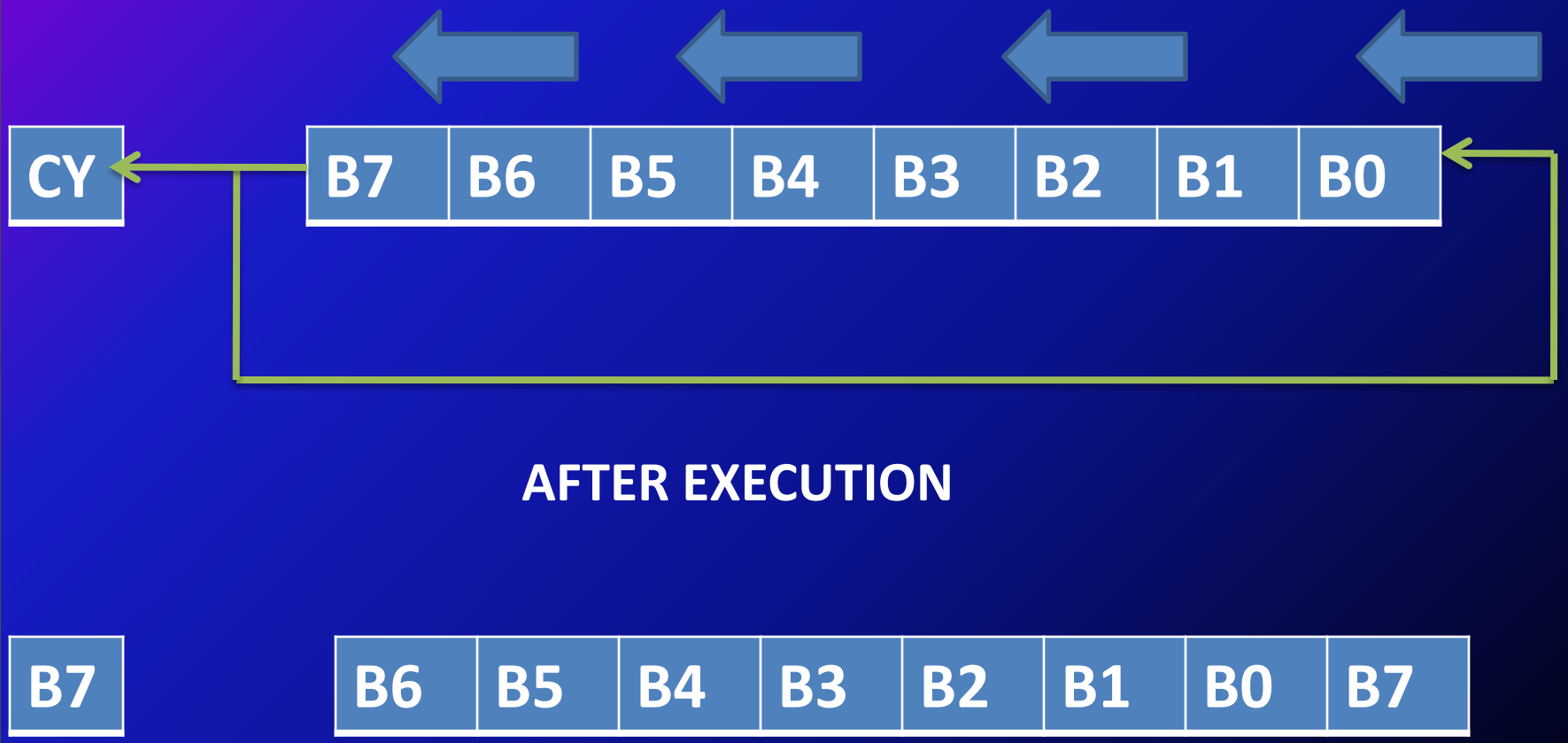
- ***RLC***

- *Rotate accumulator left*
- *Each binary bit of the accumulator is rotated left by one position.*
- ***Bit D7 is placed in the position of D0 as well as in the Carry flag.***
- *CY is modified according to bit D7.*

- *Example: RLC.*

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set



Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(15) Logical Instructions

RRC

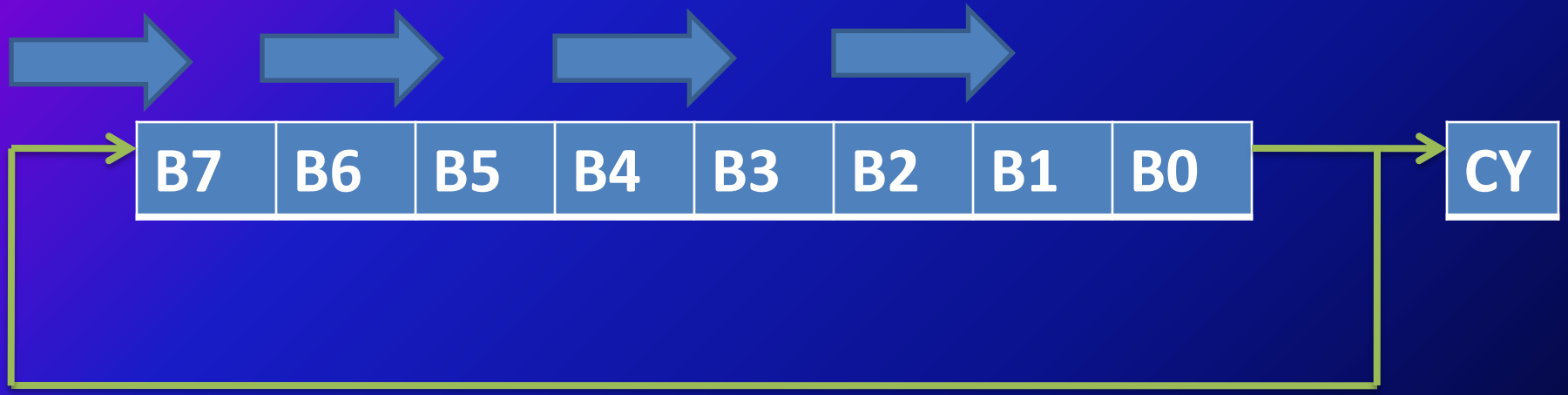
- *Rotate accumulator right*
- *Each binary bit of the accumulator is rotated right by one position*.
- Bit D0 is placed in the position of D7 as well as in the Carry flag.
- CY is modified according to bit D0.

- Example: RRC.

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION



AFTER EXECUTION

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

Branching Instructions

- *The branch group instructions allows the microprocessor to change the sequence of program either conditionally or under certain test conditions*. The group includes,
- (1) Jump instructions,
- (2) Call and Return instructions,
- (3) Restart instructions,

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(1) Branching Instructions

JUMP address

- BEFORE EXECUTION

PC	

JMP 2000H

AFTER EXECUTION

PC	2000

- *Jump unconditionally to the address.*
- *The instruction loads the PC with the address given within the instruction and resumes the program execution from specified location.*
- Example: JMP 2000H

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

Conditional Jumps

Instruction Code	Description	Condition For Jump
JC	Jump on carry	CY=1
JNC	Jump on not carry	CY=0
JP	Jump on positive	S=0
JM	Jump on minus	S=1
JPE	Jump on parity even	P=1
JPO	Jump on parity odd	P=0
JZ	Jump on zero	Z=1
JNZ	Jump on not zero	Z=0

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(2) Branching Instructions

CALL address

- ***Call unconditionally a subroutine whose starting address given within the instruction and used to transfer program control to a subprogram or subroutine*.**

- **Example: CALL 2000H**

Conditional Calls

Instruction Code	Description	Condition for CALL
CC	Call on carry	CY=1
CNC	Call on not carry	CY=0
CP	Call on positive	S=0
CM	Call on minus	S=1
CPE	Call on parity even	P=1
CPO	Call on parity odd	P=0
CZ	Call on zero	Z=1
CNZ	Call on not zero	Z=0

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(4) Branching Instructions

RST n

- Restart n (0 to 7)
- *This instruction transfers the program control to a specific memory address. The processor multiplies the RST number by 8 to calculate the vector address (in hexadecimal).*

- Example: RST 6

MP & MC APPLICATIONS

Lecture 4 – 8085 Microprocessor Instruction Set

BEFORE EXECUTION

SP	3000
PC	2000

2FFEH	
2FFFH	
3000H	

SP-1

RST 6

SP	2999
PC	0030

AFTER EXECUTION

2FFEH	01
2FFFH	20
3000H	

ADDRESS OF THE NEXT INSTRUCTION IS 2001H

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

Vector Address For Return Instructions

Instruction Code	Vector Address
RST 0	0*8=0000H
RST 1	1*8=0008H
RST 2	2*8=0010H
RST 3	3*8=0018H
RST 4	4*8=0020H
RST 5	5*8=0028H
RST 6	6*8=0030H
RST 7	7*8=0038H

Mr. Dickson Nkongo, MSc.: SJUCET – EECE Dept.

(1) Control Instructions

NOP

- *No operation*
- *No operation is performed.*
- The instruction is fetched and decoded but no operation is executed.

- Example: NOP

Mr. Dickson Nkongo, MSc.: SJUCET – ECE Dept.

(2) Control Instructions

- **HLT**

- **Halt**
- ***The CPU finishes executing the current instruction and halts any further execution*.**
- **An interrupt or reset is necessary to exit from the halt state.**

- **Example: HLT**