

# Computer Structure and Language

Hamid Sarbazi-Azad  
 Department of Computer Engineering  
 Sharif University of Technology (SUT)  
 Tehran, Iran



(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

2

## Memory Word Processing: RX Format



### Load

Mnemonic: L r1,S2(X2)  
 L r1,S2  
 L r1,D2(X2,B2)  
 L r1,D2(X2)  
 L r1,D2(,B2)  
 L r1,D2

Operation:  $r1 \leftarrow (M_{D2+(B2)+(X2)});$

OPCODE: 58h

**Note 1:** Storage address  $S2 = D2 + (B2)$ . On top of it, we have (X2) indexed to S2 to generate the final address.

**Note 2:** If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

3

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Load

Example 1:

Assembly instruction: L 2,ARR(4)

Operation:  $R2 \leftarrow (M_{(R12)+(R4)+123h});$

Machine code: 5824C123

Symbol ARR has an address formed by base register 12 and displacement value 123h.

Example 2:

Assembly instruction: L 2,ARR

Operation:  $R2 \leftarrow (M_{(R12)+123h});$

Machine code: 5820C123

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

4

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Load

Example 3:

Assembly instruction: L 2,20(4,3)

Operation:  $R2 \leftarrow (M_{(R3)+(R4)+20});$

Machine code: 58243014

Example 4:

Assembly instruction: L 2,0(5)

Operation:  $R2 \leftarrow (M_{(R5)});$

Machine code: 58250000

Example 5:

Assembly instruction: L 2,10(,5)

Operation:  $R2 \leftarrow (M_{(R5)+10});$

Machine code: 5820500A

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

5

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Store

Mnemonic: ST r1,S2(X2)  
ST r1,S2  
ST r1,D2(X2,B2)  
ST r1,D2(X2)  
ST r1,D2(,B2)  
ST r1,D2

Operation:  $M_{D2+(B2)+(X2)} \leftarrow (r1);$   
OPCODE: 50h

**Note 1:** Storage address  $S2 = D2 + (B2)$ . On top of it, we have (X2) indexed to S2 to generate the final address.

**Note 2:** If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

6

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Store

Example 1:

Assembly instruction: ST 2,ARR(4)

Symbol ARR has an address formed by base register 12 and displacement value 100h.

Operation:  $M_{(R12)+(R4)+100h} \leftarrow (R2);$   
Machine code: 5024C100

Example 2:

Assembly instruction: ST 2,ARR

Operation:  $M_{(R12)+100h} \leftarrow (R2);$   
Machine code: 5020C100

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

7

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Store

Example 3:

Assembly instruction: ST 2,20(4,3)

Operation:  $M_{(R3)+(R4)+20} \leftarrow (R2);$

Machine code: 50243014

Example 4:

Assembly instruction: ST 6,0(5)

Operation:  $M_{(R5)} \leftarrow (R6);$

Machine code: 50650000

Example 5:

Assembly instruction: ST 3,10(,5)

Operation:  $M_{(R5)+10} \leftarrow (R3);$

Machine code: 5030500A

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

8

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Add

Mnemonic:

A r1,S2(X2)

A r1,S2

A r1,D2(X2,B2)

A r1,D2(X2)

A r1,D2(,B2)

A r1,D2

Operation:

$r1 \leftarrow (r1) + (M_{D2+(B2)+(X2)});$  and update CC with result.

OPCODE: 5Ah

Note 1:

Storage address  $S2 = D2 + (B2)$ . On top of it, we have (X2) indexed to S2 to generate the final address.

Note 2:

If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

9

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Add

Example 1:

Assembly instruction: A 7,ABS(6)

Symbol ABS has an address formed by base register 11 and displacement value 234h. Suppose the content of word  $M_{(R6)+(R11)+234h}$  contains -123 and  $(R7) = 200$ .

Operation: R7 ← 77; CC ← 10;

Machine code: 5A76B234

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

10

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Add

Example 2:

Assembly instruction: A 4,5(6,7)

Suppose the content of word  $M_{(R6)+(R7)+5}$  contains 333 and  $(R4) = -400$ .

Operation: R4 ← -67; CC ← 01;

Machine code: 5A467005

Example 3:

Assembly instruction: A 6,0(10)

Suppose the content of word  $M_{(R10)}$  contains 25 and  $(R6) = -25$ .

Operation: R6 ← 0; CC ← 00;

Machine code: 5A6A0000

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

11

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Subtract

Mnemonic:

S r1,S2(X2)

S r1,S2

S r1,D2(X2,B2)

S r1,D2(X2)

S r1,D2(,B2)

S r1,D2

Operation:

$r1 \leftarrow (r1) - (M_{D2+(B2)+(X2)});$  and update CC with result.

OPCODE:

5Bh

Note 1:

Storage address  $S2 = D2 + (B2)$ . On top of it, we have (X2) indexed to S2 to generate the final address.

Note 2:

If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

12

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Subtract

Example 1:

Assembly instruction: S 3,AAA(6)

Symbol AAA has an address formed by base register 10 and displacement value 200. Suppose the content of word  $M_{(R6)+(R10)+200}$  contains -1 and  $(R3) = 7FFFFFFFh$ .

Operation:

$R3 \leftarrow 80000000h; CC \leftarrow 11;$

Machine code:

5B36A0C8

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

13

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Subtract

Example 2:

Assembly instruction: S 2,33(7)

Operation: R2 ← -30; CC ← 01;

Machine code: 5B270021

Suppose the content of word M<sub>(R7)+33</sub> contains 20 and (R2) = -10.

Example 3:

Assembly instruction: S 0,0(1)

Operation: R0 ← 5; CC ← 10;

Machine code: 5B010000

Suppose the content of word M<sub>(R1)</sub> contains 0 and (R0) = 5.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

14

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Multiply

Mnemonic: M r1,S2(X2)

M r1,S2

M r1,D2(X2,B2)

M r1,D2(X2)

M r1,D2(,B2)

M r1,D2

r1 must be an even numbered register

Operation: r1:r1+1 ← (r1+1) \* (M<sub>D2+(B2)+(X2)</sub>); and update CC with result.

OPCODE: 5Ch

Note 1: Storage address S2 = D2 + (B2). On top of it, we have (X2) indexed to S2 to generate the final address.

Note 2: If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

15

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Multiply

Example 1:

Assembly instruction:   M 2,Hamid(6)

Symbol Hamid has an address formed by base register 12 and displacement value 27Fh. Suppose the content of word  $M_{(R6)+(R12)+27Fh}$  contains -1 and  $(R3) = 7FFFFFFFh$ ,  $(R2 = 0)$ .

Operation:

Machine code:

R2:R3  $\leftarrow$  FFFFFFFF 80000001h; CC  $\leftarrow$  01;

5C26C27F

R2  $\leftarrow$  -1; R3  $\leftarrow$  80000001h;

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

16

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Multiply

Example 2:

Assembly instruction:   M 2,33(7)

Suppose the content of word  $M_{(R7)+33}$  contains 20 and  $(R3) = -10$ ,  $(R2) = 45F2h$ .

Operation:

Machine code:

R2:R3  $\leftarrow$  -200; CC  $\leftarrow$  01;

5C270021   R2  $\leftarrow$  FFFFFFFFh, R3  $\leftarrow$  -200.

Example 3:

Assembly instruction:   M 0,0(10)

Suppose the content of word  $M_{(R10)}$  contains 21 and  $(R0) = 5$ ,  $(R1)=2$ .

Operation:

Machine code:

R0:R1  $\leftarrow$  42; CC  $\leftarrow$  10;

5C0A0000   R0  $\leftarrow$  0, R1  $\leftarrow$  42.



(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

17

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Divide

Mnemonic:

D r1,S2(X2)

D r1,S2

D r1,D2(X2,B2)

D r1,D2(X2)

D r1,D2(,B2)

D r1,D2

r1 must be an even numbered register

Operation:

$$r1+1 \leftarrow \frac{(r1):(r1+1)}{(M_{D2+(B2)+(X2)})}; \quad r1 \leftarrow \text{Remainder}; \text{ and update CC.}$$

OPCODE:

5Dh

Note 1:

Storage address S2 = D2 + (B2). On top of it, we have (X2) indexed to S2 to generate the final address.

Note 2:

If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

18

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Divide

Example 1:

Assembly instruction:

D 2,Hokm(2)

Symbol Hokm has an address formed by base register 12 and displacement value 15. Suppose the content of word  $M_{(R2)+(R12)+15}$  contains -17, (R2) = 0, (R3)= 173.

Operation:

R3 ← -10; R2 ← 3; CC ← 01;

Machine code:

5D22C00F

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

19

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Divide

Example 2:

Assembly instruction: D 2,33(7)

Suppose the content of word  $M_{(R7)+33}$  contains 10 and  $(R3) = -20$ ,  $(R2) = -1$ .

Operation:  $R3 \leftarrow -2$ ;  $R2 \leftarrow 0$ ;  $CC \leftarrow 01$ ;

Machine code: 5D270021

Example 3:

Assembly instruction: D 0,0(10)

Suppose the content of word  $M_{(R10)}$  contains 21 and  $(R0) = 0$ ,  $(R1)=2$ .

Operation:  $R1 \leftarrow 0$ ;  $R0 \leftarrow 2$ ;  $CC \leftarrow 00$ ;

Machine code: 5D0A0000

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

20

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Add Logical

Mnemonic: AL r1,S2(X2)  
AL r1,S2  
AL r1,D2(X2,B2)  
AL r1,D2(X2)  
AL r1,D2(,B2)  
AL r1,D2

Operation:  $r1 \leftarrow (r1) + (M_{D2+(B2)+(X2)})$ ; and update CC accordingly.

OPCODE: 5Eh

Note 1: Storage address  $S2 = D2 + (B2)$ . On top of it, we have  $(X2)$  indexed to  $S2$  to generate the final address.

Note 2: If  $R0$  is used as Base Register,  $B2$ , or Index Register,  $X2$ , it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

21

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Add Logical

Example 1:

Assembly instruction:   AL 8, SOS+2(7)

Symbol SOS has an address formed by base register 12 and displacement value 200h. Suppose the content of word  $M_{(R7)+(R12)+202h}$  contains 123 and  $(R8) = 200$ .

Operation:               R8 ← 323; CC ← 10;

Machine code:           5E87C202

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

22

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Add Logical

Example 2:

Assembly instruction:   AL 4,X'555'(6,7)

Suppose the content of word  $M_{(R6)+(R7)+555h}$  contains 333 and  $(R4) = 1400$ .

Operation:               R4 ← 1733; CC ← 10;

Machine code:           5E467555

Example 3:

Assembly instruction:   AL 2,0(1)

Suppose the content of word  $M_{(R1)}$  contains 25 and  $(R2) = -25$ .

Operation:               R2 ← 0; CC ← 11;

Machine code:           5E210000

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

23

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Subtract Logical

Mnemonic:

SL r1,S2(X2)

SL r1,S2

SL r1,D2(X2,B2)

SL r1,D2(X2)

SL r1,D2(,B2)

SL r1,D2

Operation:

$r1 \leftarrow (r1) - (M_{D2+(B2)+(X2)});$  and update CC accordingly.

OPCODE:

5Fh

Note 1:

Storage address  $S2 = D2 + (B2)$ . On top of it, we have (X2) indexed to S2 to generate the final address.

Note 2:

If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

24

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Subtract Logical

Example 1:

Assembly instruction:

SL 8, SIX-16(7)

Symbol SIX has an address formed by base register 12 and displacement value 200h. Suppose the content of word  $M_{(R7)+(R12)+1F0h}$  contains 123 and  $(R8) = 200$ .

Operation:

$R8 \leftarrow 77; CC \leftarrow 10;$

Machine code:

5F87C1F0

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

25

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Subtract Logical

Example 2:

Assembly instruction:SL 4,B'111'(6,7)

Suppose the content of word  $M_{(R6)+(R7)+7}$  contains 333 and  $(R4) = 1400$ .

Operation: $R4 \leftarrow 1067; CC \leftarrow 10;$

Machine code:5F467007

Example 3:

Assembly instruction:SL 2,0(1)

Suppose the content of word  $M_{(R1)}$  contains 35 and  $(R2) = 35$ .

Operation: $R2 \leftarrow 0; CC \leftarrow 00;$

Machine code:5F210000

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

26

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Compare

Mnemonic:

C r1,S2(X2)

C r1,S2

C r1,D2(X2,B2)

C r1,D2(X2)

C r1,D2(,B2)

C r1,D2

Operation:

Realize  $(r1) - (M_{D2+(B2)+(X2)})$  and update CC accordingly.

OPCODE:59h

Note 1:

Storage address  $S2 = D2 + (B2)$ . On top of it, we have  $(X2)$  indexed to  $S2$  to generate the final address.

Note 2:

If  $R0$  is used as Base Register,  $B2$ , or Index Register,  $X2$ , it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

27

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Compare

Example 1:

Assembly instruction: C 2, HEY(11)

Symbol HEY has an address formed by base register 12 and displacement value 70h. Suppose the content of word  $M_{(R12)+(R11)+70h}$  contains -100 and  $(R2) = -120$ .

Operation:

Machine code:

$CC \leftarrow 01;$

592BC070

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

28

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Compare

Example 2:

Assembly instruction: C 4,B'11'(B'10',X'A')

Suppose the content of word  $M_{(R2)+(R10)+3}$  contains 33 and  $(R4) = 140$ .

Operation:

Machine code:

$CC \leftarrow 10;$

5942A003

Example 3:

Assembly instruction: C 2,0(1)

Suppose the content of word  $M_{(R1)}$  contains 35 and  $(R2) = 35$ .

Operation:

Machine code:

$CC \leftarrow 00;$

59210000

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

29

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Compare Logical

Mnemonic:

CL r1,S2(X2)

CL r1,S2

CL r1,D2(X2,B2)

CL r1,D2(X2)

CL r1,D2(,B2)

CL r1,D2

Operation:

Realize  $(r1) - (M_{D2+(B2)+(X2)})$  and update CC accordingly.

OPCODE:

55h

Note 1:

Storage address  $S2 = D2 + (B2)$ . On top of it, we have (X2) indexed to S2 to generate the final address.

Note 2:

If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

30

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Compare Logical

Example 1:

Assembly instruction:

CL 2, HEY(11)

Symbol HEY has an address formed by base register 12 and displacement value 70h. Suppose the content of word  $M_{(R12)+(R11)+70h}$  contains 100 and  $(R2) = -120$ .

Operation:

$CC \leftarrow 10;$

Machine code:

552BC070

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

31

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Compare Logical

Example 2:

Assembly instruction:CL 4,B'11'(B'10',X'A')

Operation:CC ← 01;

Machine code:5542A003

Suppose the content of word  $M_{(R2)+(R10)+3}$  contains 3 and  $(R4) = 1$ .

Example 3:

Assembly instruction:CL 2,0(1)

Operation:CC ← 00;

Machine code:55210000

Suppose the content of word  $M_{(R1)}$  contains -5 and  $(R2) = -5$ .

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

32

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

And

Mnemonic:

N r1,S2(X2)

N r1,S2

N r1,D2(X2,B2)

N r1,D2(X2)

N r1,D2(,B2)

N r1,D2

Operation:r1 ← (r1) ∧ ( $M_{D2+(B2)+(X2)}$ ); and update CC accordingly.

OPCODE:54h

Note 1:

Storage address  $S2 = D2 + (B2)$ . On top of it, we have  $(X2)$  indexed to  $S2$  to generate the final address.

Note 2:

If  $R0$  is used as Base Register,  $B2$ , or Index Register,  $X2$ , it is ignored and NOT accounted for address generation.



(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

33

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

And

Example 1:

Assembly instruction: N 0, ME(11)

Symbol ME has an address formed by base register 12 and displacement value 70h. Suppose the content of word  $M_{(R12)+(R11)+70h}$  contains 77h and  $(R0) = -4$ .

Operation:  $R0 \leftarrow 74h; CC \leftarrow 10;$

Machine code: 540BC070

$FFFFFFFC \wedge 00000077 = 00000074h$

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

34

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

And

Example 2:

Assembly instruction: N 3,4(5,6)

Suppose the content of word  $M_{(R5)+(R6)+4}$  contains -13 and  $(R3) = 1$ .

Operation:  $R3 \leftarrow 1; CC \leftarrow 10;$

Machine code: 54356004

Example 3:

Assembly instruction: N 2,0(1)

Suppose the content of word  $M_{(R1)}$  contains -5 and  $(R2) = -5$ .

Operation:  $R2 \leftarrow -5; CC \leftarrow 01;$

Machine code: 54210000

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

35

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Or

Mnemonic:

O r1,S2(X2)

O r1,S2

O r1,D2(X2,B2)

O r1,D2(X2)

O r1,D2(,B2)

O r1,D2

Operation:

$r1 \leftarrow (r1) \vee (M_{D2+(B2)+(X2)});$  and update CC accordingly.

OPCODE:

56h

Note 1:

Storage address  $S2 = D2 + (B2)$ . On top of it, we have (X2) indexed to S2 to generate the final address.

Note 2:

If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

36

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Or

Example 1:

Assembly instruction:

O 0, ME(11)

Symbol ME has an address formed by base register 12 and displacement value 70h. Suppose the content of word  $M_{(R12)+(R11)+70h}$  contains 77h and  $(R0) = -4$ .

Operation:

$R0 \leftarrow -1; CC \leftarrow 01;$

Machine code:

560BC070

$FFFFFFFC \vee 00000077 = FFFFFFFFh$

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

37

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Or

Example 2:

Assembly instruction: O 3,4(5,6)

Operation: R3 ← -13; CC ← 01;

Machine code: 56356004

Suppose the content of word  $M_{(R5)+(R6)+4}$  contains -13 and (R3) = 1.

Example 3:

Assembly instruction: O 2,0(1)

Operation: R2 ← -5; CC ← 01;

Machine code: 56210000

Suppose the content of word  $M_{(R1)}$  contains -5 and (R2) = -5.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

38

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Exclusive-Or

Mnemonic: X r1,S2(X2)

X r1,S2

X r1,D2(X2,B2)

X r1,D2(X2)

X r1,D2(,B2)

X r1,D2

Operation: r1 ← (r1) xor ( $M_{D2+(B2)+(X2)}$ ); and update CC accordingly.

OPCODE: 57h

Note 1:

Storage address  $S2 = D2 + (B2)$ . On top of it, we have (X2) indexed to S2 to generate the final address.

Note 2:

If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

39

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Exclusive-Or

Example 1:

Assembly instruction: X 0, ME(11)

Symbol ME has an address formed by base register 12 and displacement value 70h. Suppose the content of word  $M_{(R12)+(R11)+70h}$  contains 77h and  $(R0) = -4$ .

Operation:  $R0 \leftarrow \text{FFFFFF8Bh}; CC \leftarrow 01;$

Machine code: 570BC070

$\text{FFFFFFFC} \text{ xor } 00000077 = \text{FFFFFF8Bh}$

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

40

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Exclusive-Or

Example 2:

Assembly instruction: X 3,4(5,6)

Suppose the content of word  $M_{(R5)+(R6)+4}$  contains -13 and  $(R3) = 1$ .

Operation:  $R3 \leftarrow -14; CC \leftarrow 01;$

Machine code: 57356004

$\text{FFFFFFF3h} \text{ xor } 00000001 = \text{FFFFFFF2h}$

Example 3:

Assembly instruction: X 2,0(1)

Suppose the content of word  $M_{(R1)}$  contains -5 and  $(R2) = -5$ .

Operation:  $R2 \leftarrow 0; CC \leftarrow 00;$

Machine code: 57210000

## Memory Word Processing: RX Format



### Load Address

Mnemonic:     LA r1,S2(X2)  
                   LA r1,S2  
                   LA r1,D2(X2,B2)  
                   LA r1,D2(X2)  
                   LA r1,D2(,B2)  
                   LA r1,D2

Operation:      $r1 \leftarrow D2 + (B2) + (X2)$ ;

OPCODE:       41h

**Note 1:** Storage address  $S2 = D2 + (B2)$ . On top of it, we have (X2) indexed to S2 to generate the final address.

**Note 2:** If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

## Memory Word Processing: RX Format



**Load Address** is a useful instruction and can be used to:

- Load the address of a symbol (say PUT) into a register (say R2):  
 $LA\ 2,PUT$
- Initialize a register (say R5) with a positive 12-bit number (0-4095):  
 $LA\ 5,8$
- To add two registers (say R5, R7) with a positive 12-bit number (0-4095) and store the result into a register (say R9):  
 $LA\ 9,77(5,7)$
- To add two registers (say R5, R7) and store the result into a register (say R9):  
 $LA\ 9,0(5,7)$
- To add a register (say R5) with a positive 12-bit number (0-4095) and store the result into a register (say R9):  
 $LA\ 9,800(5)$

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

43

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Load Address

Example 1:

Assembly instruction: LA 3,4(5,6)     Suppose (R5)+(R6)+4 = 001AF234h.

Operation:  $R3 \leftarrow 001AF234h;$

Machine code: 41356004

Example 2:

Assembly instruction: LA 1,1(1)     Suppose (R1)=77

Operation:  $R1 \leftarrow 78;$

Machine code: 41110001

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

44

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Branch on Count

Mnemonic: BCT r1,S2(X2)

BCT r1,S2

BCT r1,D2(X2,B2)

BCT r1,D2(X2)

BCT r1,D2(,B2)

BCT r1,D2

Operation:  $r1 \leftarrow (r1) - 1; \text{ if } (r1) \neq 0 \text{ then } PC \leftarrow D2 + (B2) + (X2);$

OPCODE: 46h

Note 1: Storage address  $S2 = D2 + (B2)$ . On top of it, we have (X2) indexed to S2 to generate the final address.

Note 2: If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

45

Memory Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Branch on Count

Example 1:

Assembly instruction:    BCT 3,LOOP    Suppose symbol LOOP is formed by base register 12 and displacement 1FFh and (R3)=2.

Operation:                 $R3 \leftarrow 1; PC \leftarrow (R12) + 1FFh;$

Machine code:            4630C1FFh

Example 2:

Assembly instruction:    BCT 1,2(3,4)    Suppose (R1)=1

Operation:                 $R1 \leftarrow 0;$

Machine code:            46134002

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

46

Branch Instructions:

OPCODE

M1

X2

B2

D2

Branch on Condition

Mnemonic:            BC M1,S2(X2)

BC M1,S2

BC M1,D2(X2,B2)

BC M1,D2(X2)

BC M1,D2(B2)

BC M1,D2

Operation:            If M1 conforms CC then  $PC \leftarrow D2 + (B2) + (X2);$

OPCODE:              47h

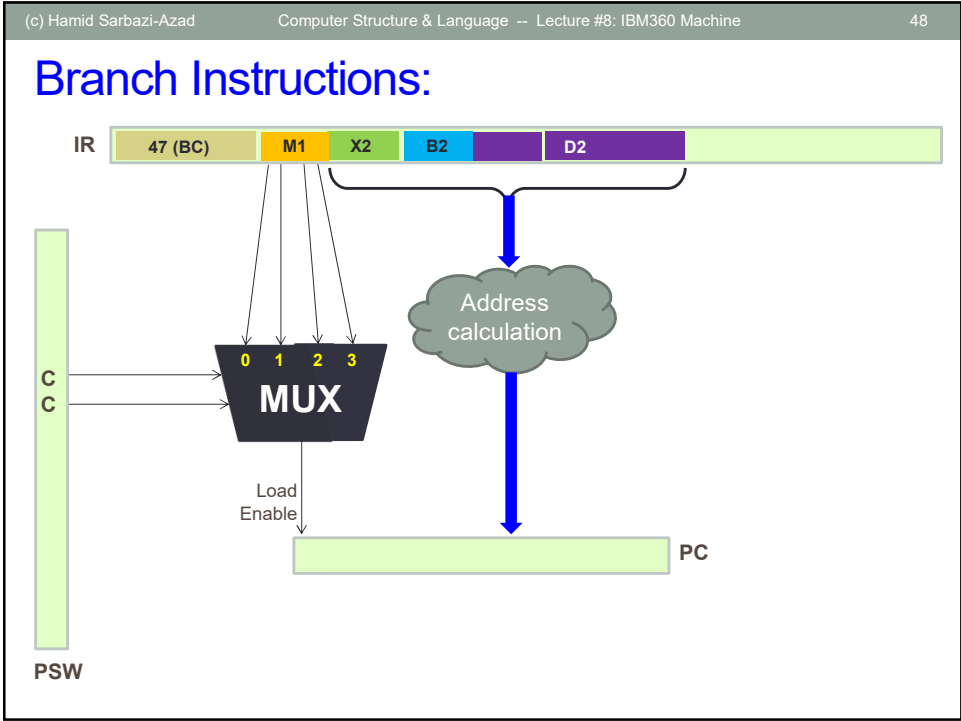
Note 1: Storage address  $S2 = D2 + (B2)$ . On top of it, we have (X2) indexed to S2 to generate the final address.

Note 2: If R0 is used as Base Register, B2, or Index Register, X2, it is ignored and NOT accounted for address generation.

(c) Hamid Sarbazi-AzadComputer Structure & Language -- Lecture #8: IBM360 Machine47

Branch Instructions:

BC Instruction & Mask (M1)	Meaning	Extended Mnemonic	
BC 0,S2(X2)	No operation	NOP	
BC 1,S2(X2)	Branch on overflow	BO	S2(X2)
BC 2,S2(X2)	Branch on plus/high	BP/BH	S2(X2)
BC 3,S2(X2)	Branch on plus/high or overflow	-	
BC 4,S2(X2)	Branch on minus/low	BM/BL	S2(X2)
BC 5,S2(X2)	Branch on minus/low or overflow	-	
BC 6,S2(X2)	Branch on plus or high/minus or low	-	
BC 7,S2(X2)	Branch on not zero/not equal	BNZ/BNE	S2(X2)
BC 8,S2(X2)	Branch on zero/equal	BZ/BE	S2(X2)
BC 9,S2(X2)	Branch on zero/equal or overflow	-	
BC 10,S2(X2)	Branch on zero/equal or plus/high	-	
BC 11,S2(X2)	Branch on not minus/not low	BNM/BNL	S2(X2)
BC 12,S2(X2)	Branch on zero/equal or minus/low	-	
BC 13,S2(X2)	Branch on not plus/not high	BNP/BNH	S2(X2)
BC 14,S2(X2)	Branch on not overflow	BNO	S2(X2)
BC 15,S2(X2)	Branch always (unconditional)	B	S2(X2)





(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

49

Branch Instructions:

OPCODE

M1

X2

B2

D2

Branch on Condition

Example 1:

Assembly instruction:BC 3,LOOP

Suppose symbol LOOP is formed by base register 12 and displacement 1FFh and CC=11.

Operation: $PC \leftarrow (R12)+1FFh;$

Machine code:4730C1FFh

Example 2:

Assembly instruction:BC B'1011',2(3,4)

Suppose CC=10

Operation: $PC \leftarrow (R3)+(R4)+2;$

Machine code:47B34002

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

50

Branch Instructions:

OPCODE

M1

X2

B2

D2

Branch on Condition

Example 3:

Assembly instruction:BNE LOP1

Suppose symbol LOP1 is formed by base register 12 and displacement 100h and C=00.

Operation:Nothing

Machine code:4770C100h

Example 4:

Assembly instruction:BNO 12(3,4)

Suppose CC=10

Operation: $PC \leftarrow (R3)+(R4)+12;$

Machine code:47E3400C

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

51

Branch Instructions in RR Format:

OPCODE

M1

r2

Branch on Condition Register

Mnemonic:

BCR M1,R2

Operation:

If M1 conforms CC then  $PC \leftarrow (R2)$ ;

OPCODE:

07h

Example 1:

Assembly instruction:

BCR 8,7 CC = 00

Operation:

$PC \leftarrow (R7)$ ;

Machine code:

0787

Example 2:

Assembly instruction:

BOR 5 CC = 01

Operation:

Nothing

Machine code:

0715

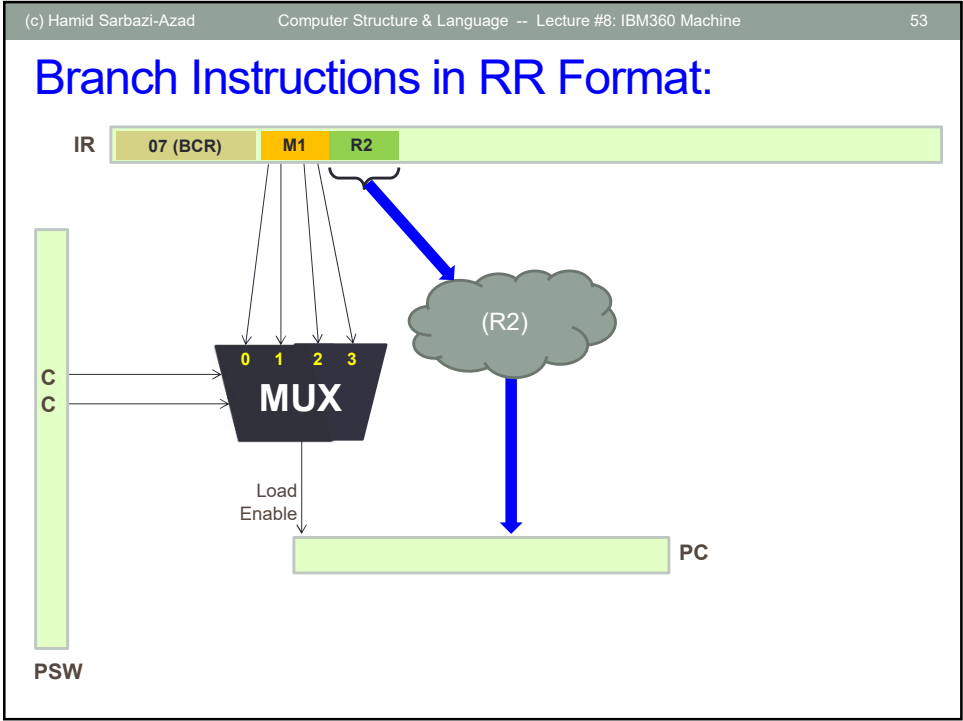
(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

52

Branch Instructions in RR Format:

BCR Instruction & Mask (M1)	Meaning	Extended Mnemonic
BCR 0,R2	No operation	-
BCR 1,R2	Branch on overflow register	BOR R2
BCR 2,R2	Branch on plus/high register	BPR/BHR R2
BCR 3,R2	Branch on plus/high or overflow register	-
BCR 4,R2	Branch on minus/low register	BMR/BLR R2
BCR 5,R2	Branch on minus/low or overflow register	-
BCR 6,R2	Branch on plus or high/minus or low reg.	-
BCR 7,R2	Branch on not zero/not equal register	BNZR/BNER R2
BCR 8,R2	Branch zero/equal register	BZR/BER R2
BCR 9,R2	Branch on zero/equal or overflow reg.	-
BCR 10,R2	Branch on zero/equal or plus/high reg.	-
BCR 11,R2	Branch on not minus/not plus register	BNMR/BNLR R2
BCR 12,R2	Branch on zero/equal or minus/low reg.	-
BCR 13,R2	Branch on not plus/not high register	BNPR/BNHR R2
BCR 14,R2	Branch on not overflow register	BNOR R2
BCR 15,R2	Branch always (unconditional) register	BR R2



(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #8: IBM360 Machine

54

End of Slides