

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 #10: IBM360 Machine

2

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Load Half-Word

Mnemonic:

LH r1,S2(X2)

LH r1,S2

LH r1,D2(X2,B2)

LH r1,D2(X2)

LH r1,D2(,B2)

LH r1,D2

Operation:

$r1 \leftarrow (M_{D2+(B2)+(X2)})_{HW}$

The 16-bit half-word is sign-extended to 32-bit word

OPCODE:

48h

Note 1:

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

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 #10: IBM360 Machine

3

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Load Half-Word

Example 1:

Assembly instruction: LH 2,ARRAY(4)

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

Machine code: 4824C123

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

Example 2:

Assembly instruction: LH 2,ARRAY

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

Machine code: 4820C123

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #10: IBM360 Machine

4

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Load Half-Word

Example 3:

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

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

Machine code: 48243014

Example 4:

Assembly instruction: LH 2,0(5)

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

Machine code: 48250000

Example 5:

Assembly instruction: LH 2,10(,5)

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

Machine code: 4820500A

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #10: IBM360 Machine

5

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Store Half-Word

Mnemonic:

STH r1,S2(X2)

STH r1,S2

STH r1,D2(X2,B2)

STH r1,D2(X2)

STH r1,D2(,B2)

STH r1,D2

Operation:

$M_{D2+(B2)+(X2)} \leftarrow (r1)_{15-0};$

The least-significant half-word in r1 is used

OPCODE:

40h

Note 1:

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

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 #10: IBM360 Machine

6

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Store Half-Word

Example 1:

Assembly instruction:

STH 2,ARR(4)

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

Operation:

$M_{(R12)+(R4)+100h} \leftarrow (R2)_{15-0};$

Machine code:

4024C100

Example 2:

Assembly instruction:

STH 2,ARR

Operation:

$M_{(R12)+100h} \leftarrow (R2)_{15-0};$

Machine code:

4020C100

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #10: IBM360 Machine

7

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Store Half-Word

Example 3:

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

Operation: $M_{(R3)+(R4)+20} \leftarrow (R2)_{15-0};$

Machine code: 40243014

Example 4:

Assembly instruction: STH 6,0(5)

Operation: $M_{(R5)} \leftarrow (R6)_{15-0};$

Machine code: 40650000

Example 5:

Assembly instruction: STH 3,10(,5)

Operation: $M_{(R5)+10} \leftarrow (R3)_{15-0};$

Machine code: 4030500A

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #10: IBM360 Machine

8

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Add Half-Word

Mnemonic: AH r1,S2(X2)

AH r1,S2

AH r1,D2(X2,B2)

AH r1,D2(X2)

AH r1,D2(,B2)

AH r1,D2

Operation: $r1 \leftarrow (r1) + (M_{D2+(B2)+(X2)})_{HW}$ and update CC with result.

OPCODE: 4Ah

The 16-bit half-word is sign-extended to 32-bit word before addition

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

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 #10: IBM360 Machine

9

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Add Half-Word

Example 1:

Assembly instruction: AH 7,ABS(6)

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

Operation: R7 \leftarrow 77; CC \leftarrow 10;

Machine code: 4A76B234

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #10: IBM360 Machine

10

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Add Half-Word

Example 2:

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

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

Operation: R4 \leftarrow -67; CC \leftarrow 01;

Machine code: 4A467005

Example 3:

Assembly instruction: AH 6,0(10)

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

Operation: R6 \leftarrow 0; CC \leftarrow 00;

Machine code: 4A6A0000

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #10: IBM360 Machine

11

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Subtract Half-Word

Mnemonic:

SH r1,S2(X2)

SH r1,S2

SH r1,D2(X2,B2)

SH r1,D2(X2)

SH r1,D2(,B2)

SH r1,D2

Operation:

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

OPCODE:

4Bh

The 16-bit half-word is sign-extended to 32-bit word before subtract

Note 1:

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

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 #10: IBM360 Machine

12

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Subtract Half-Word

Example 1:

Assembly instruction:

SH 3,AAA(6)

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

Operation:

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

Machine code:

4B36A0C8

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #10: IBM360 Machine

13

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Subtract Half-Word

Example 2:

Assembly instruction:SH 2,33(7)

Operation:R2 ← -30; CC ← 01;

Machine code:4B270021

Suppose the content of half-word $M_{(R7)+33}$ is 20 and $(R2) = -10$.

Example 3:

Assembly instruction:SH 0,0(1)

Operation:R0 ← 5; CC ← 10;

Machine code:4B010000

Suppose the content of half-word $M_{(R1)}$ is 0 and $(R0) = 5$.

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #10: IBM360 Machine

14

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Multiply Half-Word

Mnemonic:

MH r1,S2(X2)

MH r1,S2

MH r1,D2(X2,B2)

MH r1,D2(X2)

MH r1,D2(B2)

MH r1,D2

r1 can be any of 16 registers

Operation:

OPCODE:

$r1 \leftarrow (r1)_{15-0} * (M_{D2+(B2)+(X2)})_{HW}$ and update CC with result.

4Ch

Note 1:

Note 2:

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

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 #10: IBM360 Machine

15

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Multiply Half-Word

Example 1:

Assembly instruction:MH 2,HAM(2)

Symbol HAM has an address formed by base register 12 and displacement value 27Fh. Suppose the content of half-word $M_{(R2)+(R12)+27Fh}$ is -1 and (R2 = 2).

Operation: $R2 \leftarrow FFFFFFFEh; CC \leftarrow 01;$

Machine code:4C22C27Fh

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #10: IBM360 Machine

16

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Multiply Half-Word

Example 2:

Assembly instruction:MH 3,33(7)

Suppose the content of half-word $M_{(R7)+33}$ is 20 and (R3) = -10.

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

Machine code:4C370021

Example 3:

Assembly instruction:MH 0,0(10)

Suppose the content of half-word $M_{(R10)}$ is 21 and (R0) = 5.

Operation: $R0 \leftarrow 105; CC \leftarrow 10;$

Machine code:4C0A0000

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #10: IBM360 Machine

17

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Compare Half-Word

Mnemonic:

CH r1,S2(X2)

CH r1,S2

CH r1,D2(X2,B2)

CH r1,D2(X2)

CH r1,D2(B2)

CH r1,D2

Operation:

Realize $(r1) - (M_{D2+(B2)+(X2)})_{HW}$ and update CC with result.

OPCODE:

49h

The 16-bit half-word is sign-extended to 32-bit word before subtract

Note 1:

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

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 #10: IBM360 Machine

18

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Compare half-Word

Example 1:

Assembly instruction:

CH 2, HEY(11)

Symbol HEY has an address formed by base register 12 and displacement value 7. Suppose the content of half-word $M_{(R12)+(R11)+7}$ is -100 and (R2)=-120.

Operation:

$CC \leftarrow 01;$

Machine code:

492BC007

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #10: IBM360 Machine

19

Half-Word Processing: RX Format

OPCODE

r1

X2

B2

D2

Compare Half-Word

Example 2:

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

Operation:CC ← 10;

Machine code:4942A003

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

Example 3:

Assembly instruction:CH 2,0(1)

Operation:CC ← 00;

Machine code:49210000

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

(c) Hamid Sarbazi-Azad

Computer Structure & Language -- Lecture #10: IBM360 Machine

20

Example: Write an assembly program to sort a100-element half-Word array ARR in ascending order (Selection Sort).

SORTH START 0

Defining R12 as base register & initialize it to 6 → (R12) =6.

LA 2,99

XR 6,6 outer index

LOP2 LA 7,2(6) inner index

LR 3,2

LOP1 LH 4,ARR(6)

CH 5,ARR(7)

BL OUT

LH 5,ARR(7)

STH 4,ARR(7)

STH 5,ARR(6)

OUT LA 7,2(7)

BCT 3,LOP1

LA 6,2(6)

BCT 2,LOP2

Returning to OS

ARR DS 100H

END SORTH

(c) Hamid Sarbazi-AzadComputer Structure & Language -- Lecture #10: IBM360 Machine21

Example: Write an assembly program to sort a 100-element half-word array ARR in ascending order (Selection Sort).

Address	Machine Code	Assembly Code
000000		SORTH START 0
		Defining R12 as base register & initialize it to 6 → (R12) = 6. Six Bytes
000006	41200063	LA 2,99
00000A	1766	XR 6,6 outer index
00000C	41760002	LOP2 LA 7,2(6) inner index
000010	1832	LR 3,2
000012	4846C03A	LOP1 LH 4,ARR(6)
000016	4947C03A	CH 4,ARR(7)
00001A	4740C024	BL OUT
00001E	4857C03A	LH 5,ARR(7)
000022	4047C03A	STH 4,ARR(7)
000026	4056C03A	STH 5,ARR(6)
00002A	41770002	OUT LA 7,2(7)
00002E	4630C00C	BCT 3,LOP1
000032	41660002	LA 6,2(6)
000036	4620C006	BCT 2,LOP2
00003A		Returning to OS Six Bytes
000040		ARR DS 100H
		END SORTH

Symbol Table

Symbol	B	Disp.
LOP2	C	006h
LOP1	C	00Ch
OUT	C	024h
ARR	C	03Ah

(c) Hamid Sarbazi-AzadComputer Structure & Language -- Lecture #10: IBM360 Machine22

End of Slides