CSE 331 / L-08 / 07.03.2024 /

❀ Data Transfer Instruction:

⇒ XCHG ⇒ Exchanges contents of a register with any other register or memory location.

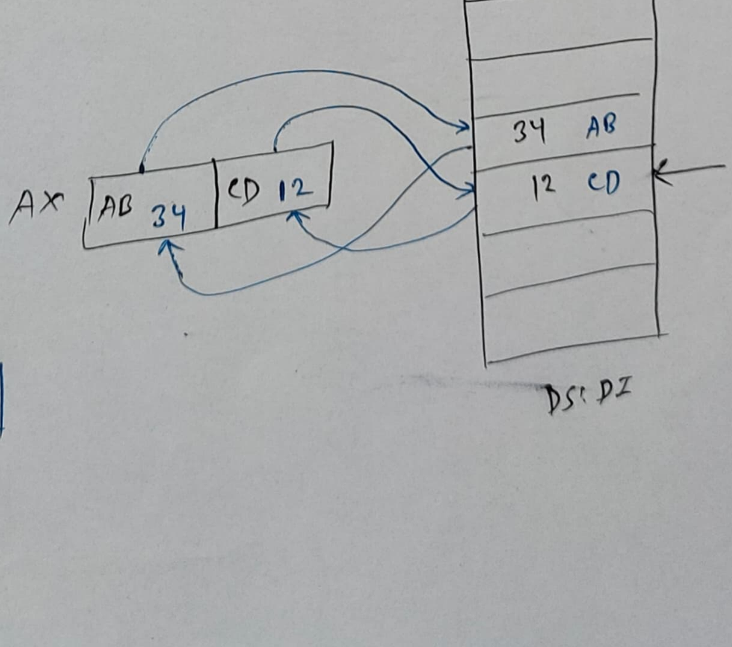
- can't exchange segment & memory to memory.
- size need to be same

❀ ⇒ XCHG AL, BL

AL = 01 H
BL = 2A H
⇒ AL = 2A H
BL = 01 H

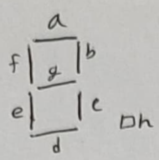
XCHG AL, [DI]
XCHG [DI], AL
Same, identical

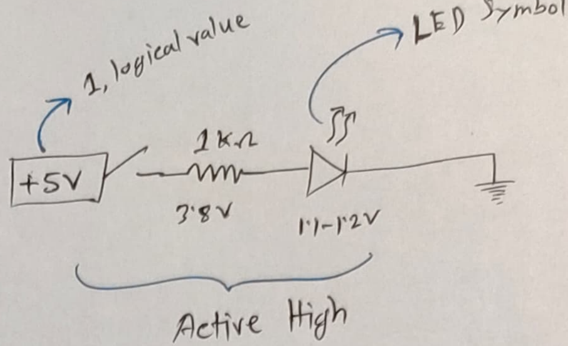
XCHG ~~AX~~, [DI]

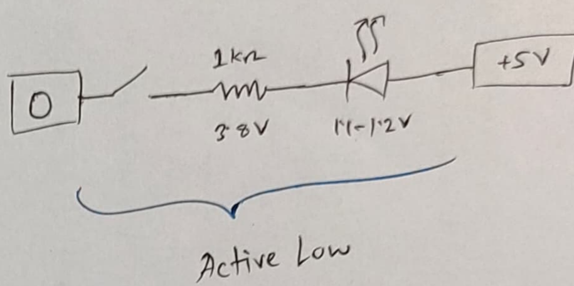


Slide-26

XLAT ⇒ Translate







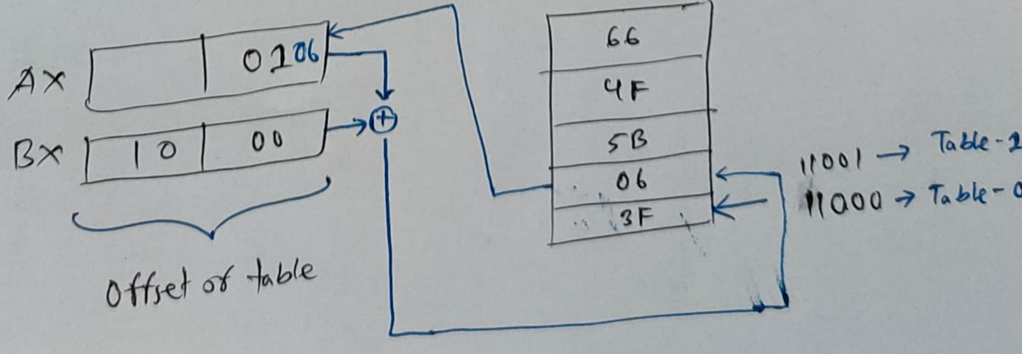
Dec	hgfe	deba	Hexa
0	0011	1111	3F
1	0000	0110	06
3	0100	1111	4F

⇒ Target: provide 1 to show 1 on the LED

⇒ XLAT
- → two register fixed for offset

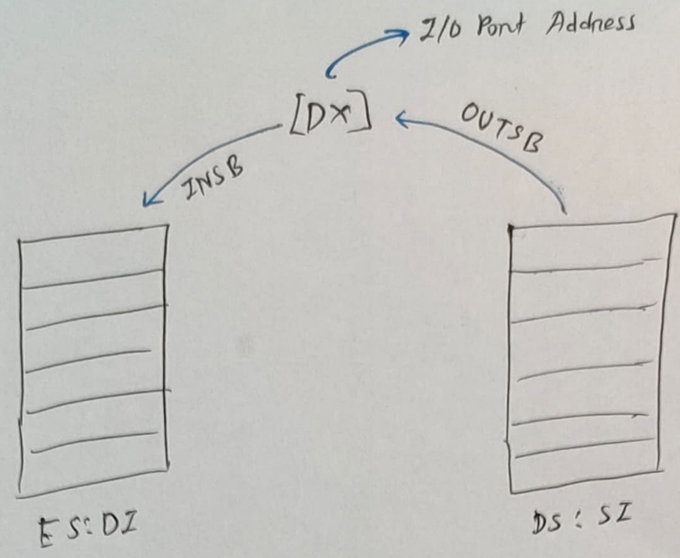
[AL+BX] ⇒ 8 bit + 16 bit = 16 bit offset

⇒ We can't display 0 by providing code 0, directly. if we can store the content at an index, and reffer the address as 0, then we can do it directly by providing 0.

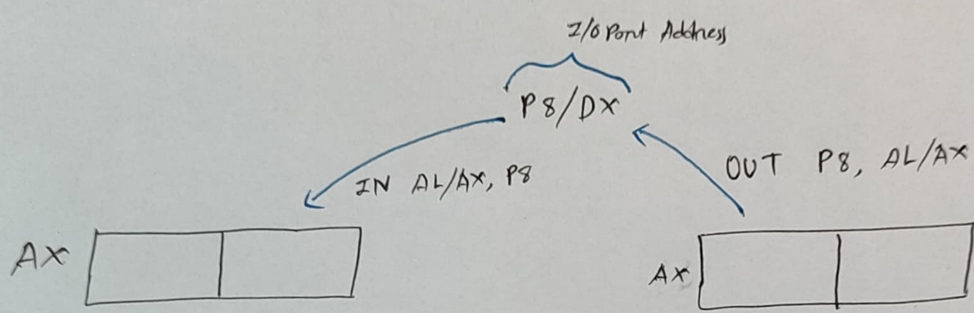


Slide-28,29

⊛ IN & OUT:



⇒



P8 ⇒ Fixed Port addressing, used for 8-bit port address

Dx ⇒ Variable port addressing, used for 16-bit port address

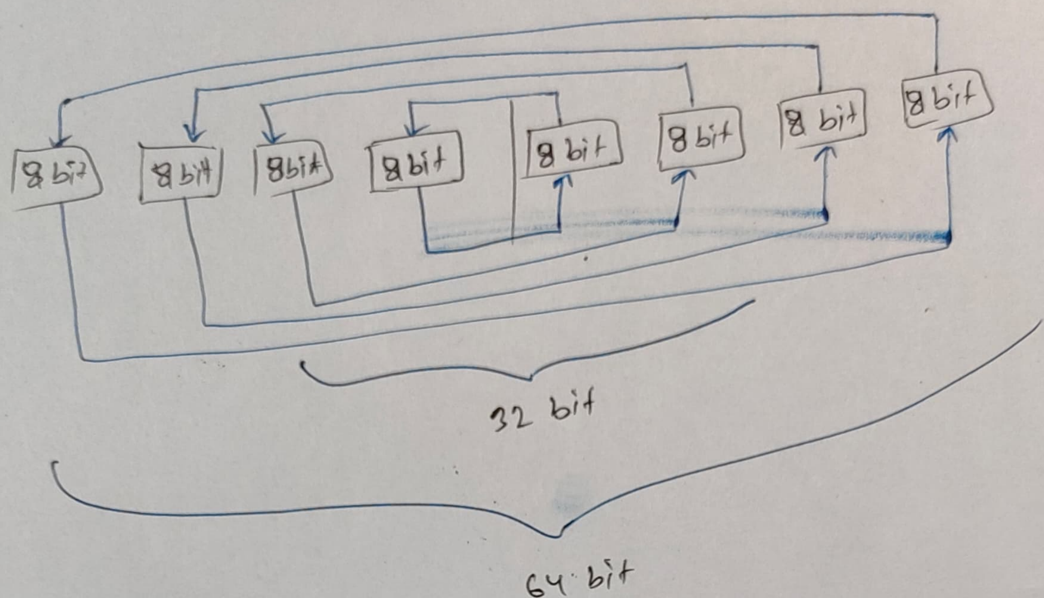
IN AX, P8

OUT P8, AX

→ Data bit size depends on source or destination register size

$\boxed{\text{Slide - 31}}$

* BSWAP ⇒ Byte Swap, minimum 32 bit architecture



* CMOV ⇒ conditional move

$\boxed{\text{slide - 37}}$

CMP AX, BX
CMOVB (if AX < BX)

} Right now, we don't to remember the flag used for the test/compare.

for unsigned data ⇒ below or above

for signed data ⇒ greater/less

↳ 0 = +
1 = -

* Segment Override: mov AX, DS: [BP]

$\boxed{\text{slide - 39}}$

$\boxed{\text{Quiz-1, 14.03.2024, Up to this}}$

Digital system principle and application
by tocci, 8th edition
Page - 282 - 297

6.1 - 6.6

6.1/ Binary Addition:

$0 + 0 = 0$
$0 + 1 = 1$
$1 + 0 = 1$
$1 + 1 = 0, 1 \text{ (carry)}$
$1 + 1 + 1 = 1, 1 \text{ (carry)}$

6.2/ Representing Signed Numbers:

- Sign-magnitude system
- 2's compliment system

0	1	1	0	1	0	0

$\Rightarrow +52_{10}$

Sign(+)

magnitude
52_{10}

1	1	1	0	1	0	0

$\Rightarrow (-)52_{10}$

Sign(-)

Not magnitude

⊛ 2's Compliment:

$$
\begin{array}{r l}
 & \overset{(+)}{0}\;\overbrace{101}^{5} \rightarrow +5 \\
\hline
 & 1010 \rightarrow \text{1's compliment/ not operation} \\
 & +1 \\
\hline
 & \underset{(-)}{1}011 \rightarrow -5, \text{ 2's Compliment}
\end{array}
$$

⇒ If signed bit is 0, then the magnitude is true binary, no need to convert.

If signed bit is 1, then magnitude is in 2's compliment form. Need to do 2's compliment again to see the inverse/ Positive number of that binary.

$$
\begin{array}{l l}
1011 & \rightarrow -5 \\
0100 & \rightarrow \text{1's compliment} \\
+1 & \\
\hline
0101 & \rightarrow +5, \text{ 2's compliment}
\end{array}
$$

Example - 6.1

b) -9

$$
\begin{array}{r l}
+9 = & 01001 \\
 & 10110 \\
 & +1 \\
\hline
 & 10111 \rightarrow -9
\end{array}
$$

d)

$$
\begin{array}{r l}
+2 = & 00010 \\
 & 11101 \\
 & +1 \\
\hline
 & 11110 = -2
\end{array}
$$

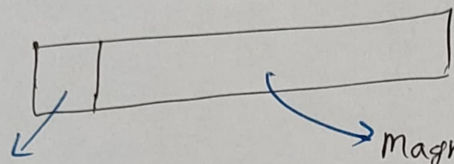
Example. 6.2

b)

$$
\begin{array}{l l}
11010 & \rightarrow -6 \\
00101 & \\
+1 & \\
\hline
00110 & \rightarrow +6
\end{array}
$$

L-9 / 09. 03. 2024 /

Special Case of 2's Compliment:



↳ 0 ⇒ magnitude is the main number

1 ⇒ need to convert in 2's compliment

sign bit

11011 → -5
00100
+1
00101 → +5

1000
0111
+1
1 000

⇒ Special Case

⇒ If sign bit is 1 and magnitude all zero, then its a special case.

bit count of magnitude

$1000 = -2^n = -2^3 = -8$

Lowest number in terms of negativity in the range of magnitude bit.

For, n-bit

minimum number

$\underbrace{-2^n \;.....\; 0 \;.....\; (2^n-1)}_{\text{total signed number } 2^{n+1}}$

maximum number

6.3/ Addition in the 2's compliment:

⇒ total - 5 cases

augend
+addend
sum

Case-1: two positive number

$$\begin{array}{rcc} +9 = & \boxed{0} & 1001 \\ +4 = & 0 & 0100 \\ \hline & 0 & 1101 \end{array} \Rightarrow +13$$

Case-2: Positive number and smaller negative number

$$\begin{array}{rcc} 4 = & 0 & 0100 \\ & 1 & 1011 \\ & & +1 \\ \hline & 1 & 1100 \end{array}$$

$$\begin{array}{rccc} +9 = & & \boxed{0} & 1001 \\ -4 = & & 1 & 1100 \\ \hline & \cancel{1} & 0 & 0101 \end{array} \rightarrow +5$$

↳ carry after signbit will be disregarded.

Case-3: Positive number and larger negative number

$$\begin{array}{rcc} -9 = & \boxed{1} & 0111 \\ +4 = & 0 & 0100 \\ \hline & 1 & 1011 \end{array} \rightarrow -5$$

case-5: Equal and Oposite number

$$\begin{array}{rccc} -9 = & & \boxed{1} & 0111 \\ +9 = & & 0 & 1001 \\ \hline & \cancel{1} & 0 & 0000 \end{array} \rightarrow +0$$

↳ Disregarded

Case-4: Tow Negative Numbers

$$\begin{array}{rr} -9 = & \boxed{1}\,0111 \\ -4 = & 1\,1100 \\ \hline & 1\,\boxed{1}\,0011 \end{array} \rightarrow -13$$

↳ disregarded.

✻ Arithmetic Overflow:

Add additional bit to get the correct number

6.4/ Substraction in 2's Compliment:

minuend
(-) subtrahend → make oposite sign by 2's compliment always, must
Result

$(+9) - (+4) \Rightarrow$

$+9 = 0\,1001$
$+4 = 00100 \Rightarrow 11100\ (-4)$

$$\begin{array}{rr} +9 = & \boxed{0}\,1001 \\ (-)\ +4 = & 1\,1100 \\ \hline & 1\,\boxed{0}\,0101 \end{array} \rightarrow +5$$

↳ Disregarded

6.5/ Multiplication:

$$\begin{array}{r} \text{multiplicant} \\ \times\ \text{multiplier} \\ \hline \text{Result} \end{array}$$

$$\begin{array}{r} 15 \\ \times 25 \\ \hline 75 \\ 30\times \\ \hline 375 \end{array}$$

$$\begin{array}{r} 1001 \\ 1011 \\ \hline 1001 \\ 1001\times \\ 0000\times\times \\ 1001\times\times\times \\ \hline 1100011 \end{array}$$

⇒ if multiplier is 1
multiplication = multiplicant

⇒ if multiplier is 0
multiplication = 0000

6.6/ Division:

divison | dividend → Result/Quotient
⋮
Remainder

$$\begin{array}{r} 06 \\ 3\,\overline{)\,19} \\ 0\downarrow \\ \hline 19 \\ 18 \\ \hline 1 \end{array}$$

$$\begin{array}{r} 0011 \\ 11\,\overline{)\,1011} \\ 0 \\ \hline 10 \\ 00 \\ \hline 101 \\ 11 \\ \hline 0101 \\ 11 \\ \hline 10 \end{array}$$

0011 → 3

11 → 3

+10=2

10 → 2

Ytha Yu
Charles Marut } Chapter-9
Page-173

Multiplication & Division Instruction

Multiplication

- → unsigned (MUL)   BL | 8 bit |
- → signed (IMUL)   BL | S | 7 bit |
  - ↓ integer

MUL BL

→ multiplier

multiplicant is set by default, AL

Destination ⇒ 16 bit, AX

⇒ 32 bit, DX - AX

→ as BL is 8 bit, hence multiplicant will be 8 bit AL

$$\begin{array}{rl} AL & = 8 \text{ bit} \\ \times \; BL & = 8 \text{ bit} \\ \hline AX & = 16 \text{ bit} \end{array}$$

MUL BX

$$\Rightarrow \begin{array}{r} AX \\ \times \; BX \\ \hline DX - AX \end{array}$$

H   L

$$\text{MUL EBX} \Rightarrow \begin{array}{r} EAX \\ EBX \\ \hline EDX - EAX \end{array}$$

* IMUL BL ⇒

$$\begin{array}{r} AL \\ BL \\ \hline AX \end{array}$$

} 1 bit preserved for sign bit.

* MUL BX

$$\begin{array}{rl} AX = & 0001\ H \\ BX = & FFFF\ H \\ \hline & \underbrace{0000}_{DX}\ \underbrace{FFFF}_{AX}\ H \end{array}$$

* IMUL BX

⇒

0000 → S

$$\begin{array}{rll} AX = & 0001\ H & \Rightarrow +1 \\ BX = & FF\ FF\ H & \Rightarrow -1 \\ \hline & \underbrace{FFFF}_{DX}\ \underbrace{FFFF}_{AX} & \Rightarrow -1 \end{array}$$

1111 → S → need to convert in 2's compliment to see the value

* If first bit is 0, then calculator auto fill the front bit by 0. if first bit is 1, then we need to fill the front bit by F to complete 8 bit Hexa.

*

CF/OF ⇒ defined if the ~~result~~ sign bit of result extended or not.

CF/OF = 0, extended

= 1, not extended.

* MUL BX

$$
\begin{array}{rcl}
AX = FFFF\ H & = & 65535 \\
BX = FFFF\ H & = & 65535 \\
\hline
\underbrace{FFFE}_{DX}\,\underbrace{0001}_{AX} & = & 4294836225
\end{array}
$$

→ not filled by 0

CF/OF = 1

* IMUL, BX

$$
\begin{array}{rcl}
AX = FFFF\ H & = & -1 \\
BX = FFFF\ H & = & -1 \\
\hline
 & = & +1 \\
\underbrace{0000}_{DX}\,\underbrace{0001}_{AX} & &
\end{array}
$$

→ sign bit extended,

CF/OF = 0

H.W. ⇒ All Exercise & Example 9.1 - 9.5

Page - 175 Done