بسم الله الرحمن الرحيم

DEPARTMENT OF ELECTRICAL ENGINEERING

Batch (014)

UNIVERSITY OF GUJRAT

A WORLD CLASS UNIVERSITY

UNIVERSITY OF GUJRAT

UOG

A WORLD CLASS UNIVERSITY

GROUP MEMBER

MUHAMMAD ASIM	(14093122-007)
INZMAM TAHIR	(14093122-005)
YASIR ALI	(14093122-006)
ADNAN ASLAM	(14093122-003)
BILAL AMJAD	(14093122-004)
UMAR RASHEED	(14093122-002)

PRESENTATION

MICROPROCESSOR

MULTIPLICATION AND DIVISION IN ASSEMBLY LANGUAGE

INSTRUCTOR :
MAM YUMNA BILAL

OVERVIEW

- In chapter 7 ,we saw how to learn multiplication and division by shifting the bits in a byte or word.
- Left and right shift can be used for multiplying and dividing respectively by powers of 2.
- Process of multiplication and division is different for signed and unsigned numbers and there are different instructions used for signed and unsigned multiplication and division.
- One of the most useful applications of multiplication and division is to implement decimal input and output.

UNSIGNED MULTIPLICATION (MUL)

- In the case of unsigned multiplication, using instruction MUL .
- The syntax of this instruction is

  MUL   source
- Example

AL=128

1	0	0	0	0	0	0	0

BL=255

1	1	1	1	1	1	1	1

AX=32640

0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0

SIGNED MULTIPLICATION (IMUL)

- In the case of signed multiplication, using instruction IMUL .
- The syntax of this instruction is

  IMUL source

- Example

AL=128

1	0	0	0	0	0	0	0

BL=-1

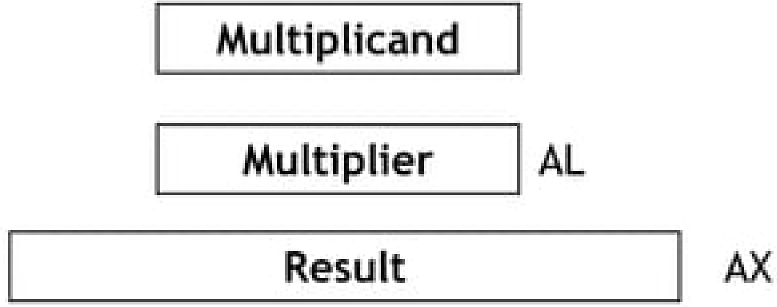
1	1	1	1	1	1	1	1

AX=32640

0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0

BYTE FORM

- In byte multiplication one number is contained in the source and the other is assumed to be in AL.
- The 16-bit product will be in AX.
- The source may be a byte register or memory byte but not a constant.



Multiplicand

Multiplier AL

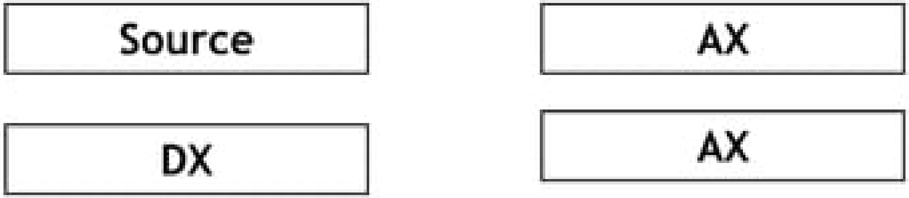
Result AX

WORD FORM

- In word multiplication one number is contained in the source and the other is assumed to be in AX.
- The most significant 16-bits of the double word product will in DX and the least significant 16-bits will be in AX.

DX:AX

- The source may a 16-bit register or memory word but not a constant.



Source

AX

DX

AX

EFFECT OF MUL/IMUL ON FLAGS

- SF,ZF,AF,PF undefined
- CF,OF:
- After MUL, CF/OF =0 if upper half of the result is zero.
  =1 otherwise
- After IMUL, CF/OF =0 the signed bits of upper and lower half are same.
  =1 otherwise

EXAMPLE (BYTE FORM)

MUL BL

F F h BL

8 0 h AL

7 F	8 0	1
AH	AL	CF/OF

IMUL BL

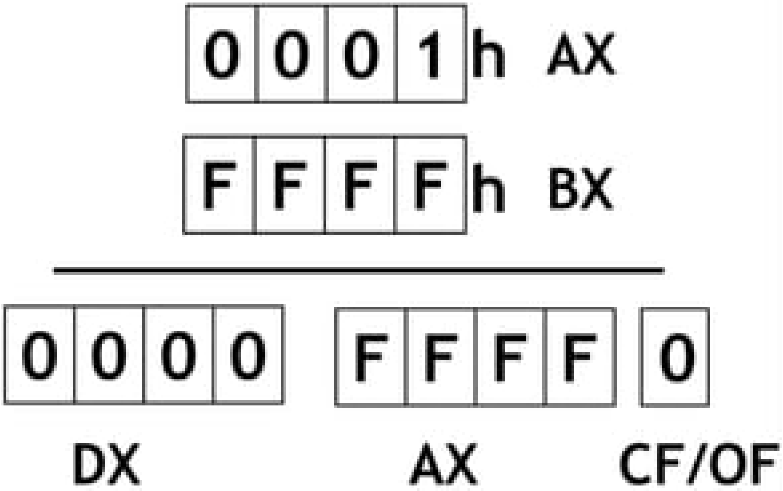
F F h BL

8 0 h AL

0 0	8 0	1
AH	AL	CF/OF

EXAMPLE (WORD FORM)

MUL BX



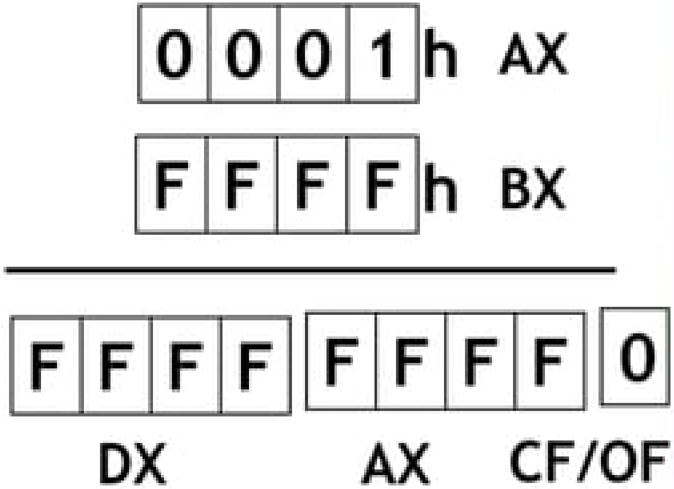
0 0 0 1 h AX

F F F F h BX

0 0 0 0 F F F F 0

DX AX CF/OF

IMUL BX



0 0 0 1 h AX

F F F F h BX

F F F F F F F F 0

DX AX CF/OF

APPLICATION EXAMPLE

- Translate the high level language assignment statement A= 5*A-12*B into assembly code remember A and B are word variable.
- sol:

```
MOV   AX = 5        ;AX =5
IMUL   A           ;AX=5*A
MOV   A , AX       ;A=5*A
MOV   AX, 12       ;AX=12
IMUL   B            ;AX=12*B
SUB     A , AX     ;A=5*A-12*B
```

DIVISION (DIV/IDIV)

- When division is performed we obtain two results the quotient and the remainder.
- In division there are separate instructions for signed and unsigned division.

SIGNED AND UNSIGNED DIVISION

- In the case of signed division IDIV (integer divide)
- The syntax used for signed division

  ```
  IDIV     divisor
  ```

- In the case of unsigned division DIV the syntax used for unsigned division

  ```
  DIV      divisor
  ```

- These instructions divide 8 (or 16) bits into 16(or 32) bits.
- The quotient and remainder have same size as the divisor

BYTE FORM

- In this case the divisor is an 8-bit register or memory byte.
- The 16-bit dividend is assume to be in AX.
- After division, 8bit quotient is in AL and 8-bit remainder is in AH.
- The divisor may not be a constant.

WORD FORM

- In this case divisor is a 16-bit register or memory word.
- The 32-bit dividend is assumed to be in DX:AX
- after division the 16-bit quotient is in AL and 16-bit remainder is in DX.
- The divisor may not be a constant.

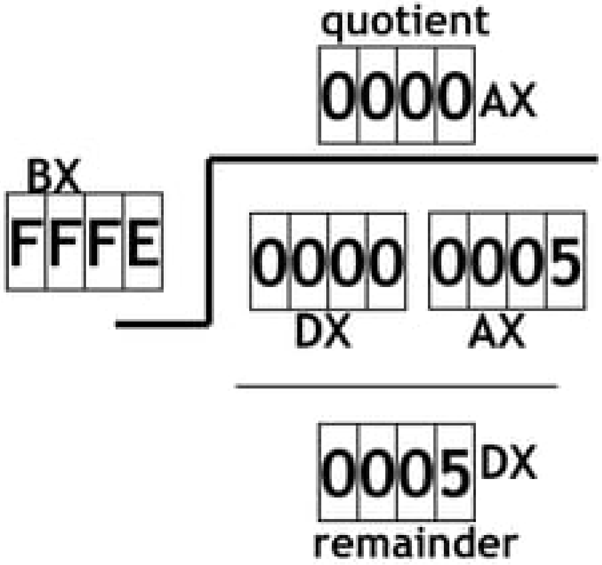
DIVIDE OVERFLOW

- It is possible that the quotient will be too big to fit in the specified destination (AL or AX).
- This can happen if the divisor is much smaller than the dividend.
- When this happens, the program terminates and the system displays the message “divide overflow”.

EXAMPLE (UNSIGNED)

Suppose DX=0000h and AX=0005h and BX=FFFEh

DIV BX



quotient

0000 AX

BX

FFFE

0000 0005

DX AX

0005 DX

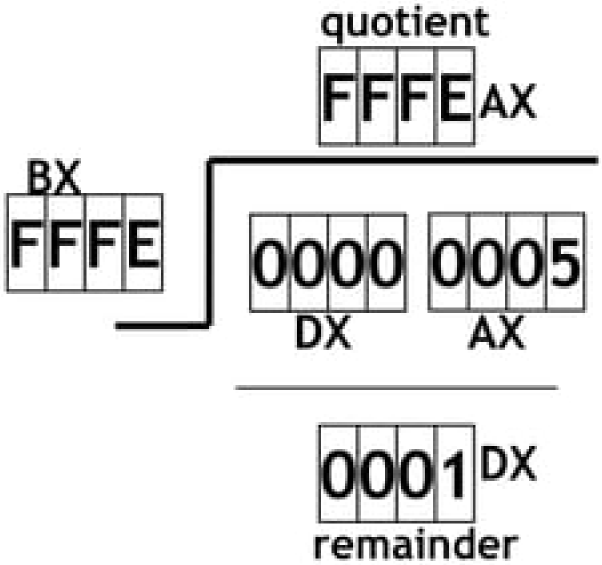
remainder

Here dividend is 5 and divisor =FFFE=65534

After division
quotient =0
and remainder =5

SIGNED

IDIV BX



quotient

FFFE AX

BX

FFFE

0000 0005

DX AX

0001 DX

remainder

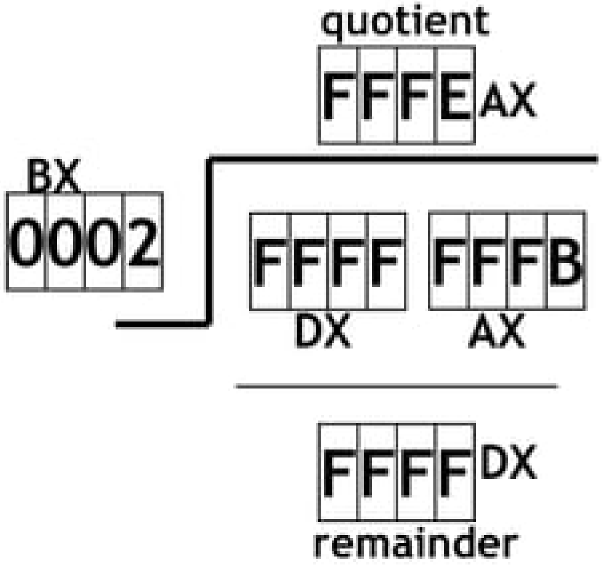
Here in the case of signed divisor = FFFE = -2 and dividend =5

After division quotient=-2 and remainder=1

EXAMPLE (SIGNED)

suppose DX=FFFFh and AX=FFFBh and BX=0002h

IDIV DX



quotient

FFFE AX

BX

0002

FFFF FFFB

DX AX

FFFF DX

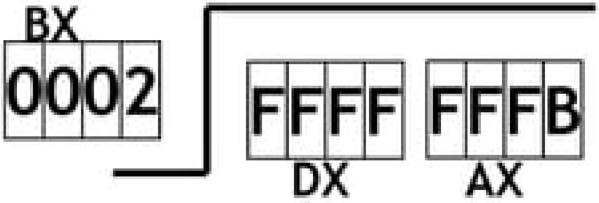
remainder

DX:AX = FFFFFFFBh = -5,
BX = 2.

-5 divided by 2 gives a quotient of -2 = FFFEh and a remainder of -1 = FFFFh.

UNSIGNED

```
DIV    BX
```



DIVIDE OVERFLOW

For DIV, the dividend DX:AX = FFFFFFFBh = 4294967291 and the divisor= 2.

The actual quotient is 2147483646 = 7FFFFFFEh. This is too big
to fit in AX.

SIGN EXTENSION OF DIVIDEND

- Word Division

  the dividend is in DX:AX even if the actual dividend will fit in AX. In this case DX should be prepared as

1. For DIV ,DX should be cleared.
2. For IDIV,DX should me made the sign extension of AX. The instruction CWD will do the extension.

EXAMPLE

- Divide -1250 by 7

Sol:-

```
MOV   AX,-1250             ;AX gets dividend
CWD                        ;extend sign to DX
MOV   BX,7                 ;BX has divisor
IDIV    BX                 ;AX get quot. DX has rem.
```

BYTE DIVISION

- The dividend is in AX. If the actual dividend is a byte then AH should be prepared as

1. For DIV,AH should be cleared.
2. For IDIV,AH should the sign extension of AL. the instruction CWB will do the extension.

EXAMPLE

- Divide the signed value of the byte variable XBYTE by -7
- Sol:-

```
MOV  AL,XBYTE            ;AL has dividend
CBW                      ;extend sign to AH
MOV BL,-7                ;BL has divisor
IDIV   BL                ;AL has quot. AH has rem.
```

- There is no effect of CBW and CWD on the flags.