

Name: Shoaib Akhtar
Roll No : 20P-0147
Report : 7

Extended Operation :

I have done multiplication of two numbers using shr operators which is also known as extended operation.

```
[org 0x0100]

mov cl , 10
mov fx , [multiplier]

checkbit:
    shr dx,1
    jnc skip

mov ax , [multiplicand]
add [result],ax
mov ax , [multiplicand+2]
add [result + 2],ax

skip:
    shl word [multiplicand+2]
    rcl word [multiplicand+2],1
    dec cl
    jnz checkbit

mov ax, 0x4c00
int 0x21

multiplicand: dd 1000
multiplier: dw 500
result: dd 0
```

The screenshot shows the DOSBox 0.74-3 interface. At the top, it displays 'DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: AFD'. Below this, there's a table of registers and their values: AX: 4C00, SI: 0000, CS: F000, IP: 14A0, Stack: +0 42BD, Flags: 7044; BX: 0000, DI: 0000, DS: 19F5, +2: 06C5; CX: 0000, BP: 0000, ES: 19F5, HS: 19F5, +4: 7044, OF: 0, DF: 0, IF: 0, SF: 0, ZF: 1, AF: 0, PF: 0, CF: 0; DX: 0000, SP: FFF2, SS: 19F5, FS: 19F5, +6: 0129. Below the registers, there's a command line 'CMD >' and a list of assembly instructions with their addresses: 0127 CD21 INT 21; 14A0 FB STI; 14A1 FE DB FE; 14A2 3B25 CMP DI, AH; 14A4 06CF ADD BH, CL; 14A6 CB RET Far; 14A7 51 PUSH CX; 14A8 B94001 MOV CX, 0140; 14AB E2FE LOOP 14AB. To the right of the instructions, there's a memory dump showing the contents of memory locations from DS:0000 to DS:004B. At the bottom, there's a status bar with buttons: 1 Step, 2 ProcStep, 3 Retrieve, 4 Help ON, 5 BRK Menu, 6, 7 up, 8 dn, 9 le, 10 ri.

Bitwise operations:

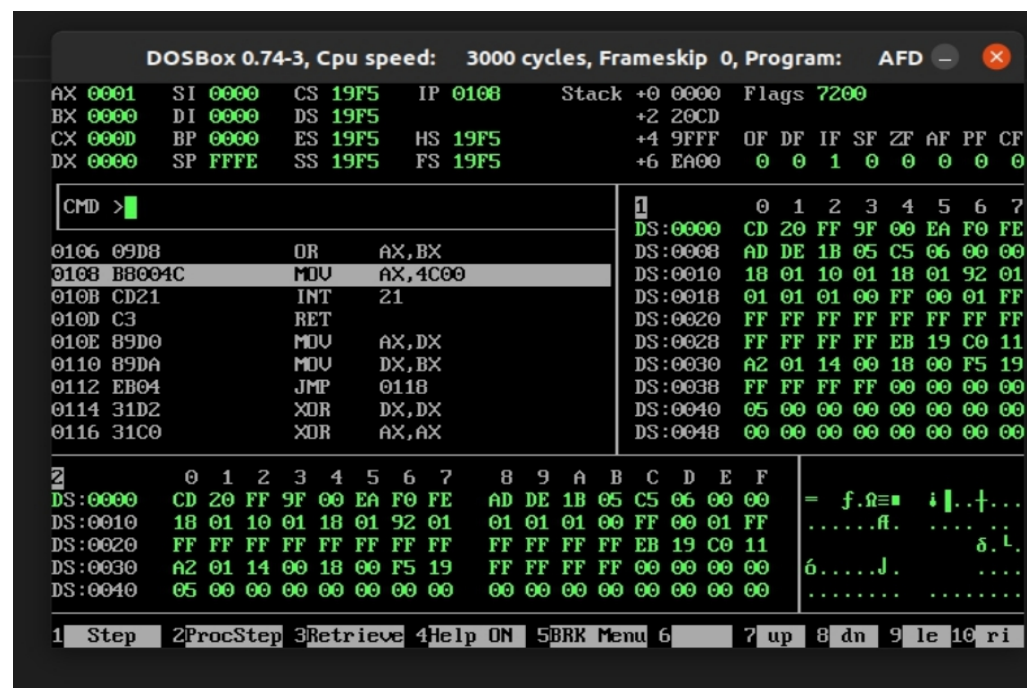
Operation which we can do on bits are called bitwise operations

```
[org 0x0100]
```

```
mov ax , 1  
mov bx , 0
```

```
or ax,bx
```

```
mov ax , 0x4c00  
int 0x21
```



Masking :

```
[org 0x0100]
```

```
mov ax , 6
mov bx , 5
```

and ax, bx

```
mov ax , 0x4c00
int 0x21
```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: AFD

AX 0004	SI 0000	CS 19F5	IP 0108	Stack +0 0000	Flags 7200
BX 0005	DI 0000	DS 19F5		+2 20CD	
CX 000D	BP 0000	ES 19F5	HS 19F5	+4 9FFF	OF DF IF SF ZF AF PF CF
DX 0000	SP FFFE	SS 19F5	FS 19F5	+6 EA00	0 0 1 0 0 0 0 0

CMD >

0106 21D8	AND	AX,BX	DS:0000	CD 20 FF 9F 00 EA F0 FE
0108 B8004C	MOV	AX,4C00	DS:0008	AD DE 1B 05 C5 06 00 00
010B CD21	INT	21	DS:0010	18 01 10 01 18 01 92 01
010D C3	RET		DS:0018	01 01 01 00 FF 00 01 FF
010E 89D0	MOV	AX,DX	DS:0020	FF FF FF FF FF FF FF FF
0110 89DA	MOV	DX,BX	DS:0028	FF FF FF FF EB 19 C0 11
0112 EB04	JMP	0118	DS:0030	A2 01 14 00 18 00 F5 19
0114 31D2	XOR	DX,DX	DS:0038	FF FF FF FF 00 00 00 00
0116 31C0	XOR	AX,AX	DS:0040	05 00 00 00 00 00 00 00
			DS:0048	00 00 00 00 00 00 00 00

2 0 1 2 3 4 5 6 7 8 9 A B C D E F

DS:0000	CD 20 FF 9F 00 EA F0 FE	AD DE 1B 05 C5 06 00 00	= f.α. i.+....
DS:0010	18 01 10 01 18 01 92 01	01 01 01 00 FF 00 01 FFα.δ.L.
DS:0020	FF FF FF FF FF FF FF FF	FF FF FF FF EB 19 C0 11J.
DS:0030	A2 01 14 00 18 00 F5 19	FF FF FF FF 00 00 00 00
DS:0040	05 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00

1 Step 2ProcStep 3Retrieve 4Help ON 5BRK Menu 6 7 up 8 dn 9 le 10 ri