

ТЕСТ 1

1. Занести $A=1$ в регистр AX , $B=-22$ в регистр BX .
2. Установить в «1» 8 бит регистра AX .
3. Инвертировать младший байт регистра BX .
4. Инвертировать «10» бит регистра BX .
5. Записать старший байт регистра BX в старший байт регистра AX .
6. Выполнить линейный арифметический сдвиг AX на один бит вправо.
7. Инвертировать 1, 2, 4-15-ые биты регистра AX .
8. Выполнить линейный логический сдвиг регистра AX на один бит влево.
9. Выполнить циклический арифметический сдвиг регистра BX на один бит вправо.
10. Выполнить линейный логический сдвиг регистра AX на один бит влево.
11. Установить в 0 «1» бит регистра BX .
12. Выполнить линейный сдвиг регистра BX на один бит вправо.
13. Выполнить линейный арифметический сдвиг регистра AX на один бит влево.
14. Найти $C = -8 + (-22)$ в дополнительном коде. Результат перевести в прямой код, выполнить проверку.

Example 1. Part 1.

XOR AX,AX	Reset AX	AX: 00 00	
XOR BX,BX	Reset BX	BX: 00 00	
MOV AX, 1	Moving $A=1_{10}$ into register AX	AX: 00 01	AX:0000 0000 0000 0001 0 0 0 1
MOV BX, -22	Moving number $B= -22_{10}$ into register BX	BX: 80 16	BX: 1000 0000 0001 0110 8 0 1 6
OR AX,0100	Setting 8-th bit at AX into 1	AX:01 01	AX:0000 0000 0000 0001 OR <u>0000 0001 0000 0000</u> <u>0000 0001 0000 0001</u> 0 1 0 1

Example 1. Part 2.

NOT BL	Inversion the low byte of BX	BX:80 16 BX:80 E9	BX: <u>1000 0000 0001 0110</u> 1000 0000 1110 1001 8 0 E 9
XOR BX, 04 00_h	Inversion 10-th bit of BX	BX: 84 E9	BX: 1000 0000 1110 1001 XOR 0000 0100 0000 0000 <hr/> 1000 0100 1110 1001 8 4 E 9
MOV AH, BH	Moving the high byte of BX into high byte of AX	AX:84 01	AX: 0000 0001 0000 0001 1000 0100 0000 0001 8 4 0 1

Example 1. Part 3.

SAR BX, 1	Linear arithmetic shift right BX one digit position	CF=0 BX:84 E9 CF=1 ← BX:C274	BX: 1000 0100 1110 1001 <div style="text-align: center;"> SAR </div> 1100 0010 0111 0100 <div style="text-align: center;"> C 2 7 4 </div>
XOR AX,FFF6	Inversion 1, 2, 4-15-th bit AX	AX:84 01 AX: 7BF7	AX: 1000 0100 0000 0001 <div style="text-align: center;"> XOR </div> 1111 1111 1111 0110 <hr/> 0111 1011 1111 0111 <div style="text-align: center;"> 7 B F 7 </div>
SHL AX, 1	Linear logical shift right AX one digit position	CF=1 AX:7BF7 CF=0 ← AX:F7EE	AX: 0111 1011 1111 0111 <div style="text-align: center;"> SHL AX,1 </div> 1111 0111 1110 1110 <div style="text-align: center;"> F 7 E E </div>
RCR BX, 1	Rotate arithmetic right BX one digit position	CF=0 BX:C274 CF=0 ← BX=A13A	BX: 1100 0010 0111 0100 <div style="text-align: center;"> RCR BX,1 </div> 1010 0001 0011 1010 <div style="text-align: center;"> A 1 3 A </div>

Example 1. Part 4.

ROL AX, 1	Rotate left AX one digit position	CF=0 AX:F7EE CF=1 ← AX:EFDD	AX: 1111 0111 1110 1110 ROL AX,1 1110 1111 1101 1101 E F D D
AND BX,FF FD	Reset 1-th bit of BX	BX: A1 38	BX: 1010 0001 0011 1010 AND <u>1111 1111 1111 1101</u> 1010 0001 0011 1000 A 1 3 8
SHR BX, 1	Linear logic shift right BX one digit position	CF=1 BX: A1 38 CF=0 ← BX: 50 9C	BX: 1010 0001 0011 1000 SHR BX,1 0101 0000 1001 1100 5 0 9 C

Example 1. Part 5.

RCL AX, 1	Rotate arithmetic left AX one digit position	CF=0 AX:EFDD CF=1 AX:DFDA	<p> AX: 1110 1111 1101 1101 RCL 1101 1111 1011 1010 D F B A </p>
XOR AX,AX	Reset AX	AX: 00 00	
XOR BX,BX	Reset BX	BX: 00 00	
MOV AX, 0x0008	Moving $A=8_{10}$ into register AX	AX: 00 08	AX:0000 0000 0000 1000 0 0 0 8
MOV BX, 0x0016	Moving number $B=22_{10}$ into register BX	BX: 00 16	BX: 0000 0000 0001 0110 0 0 1 6
NOT AX	Inversion AX	AX: FF F7	AX: 1111 1111 1111 0111 F F F 7
ADD AX, 0x0001	Add 1d to AX	AX: FF F8	AX: 1111 1111 1111 1000 F F F 8

Example 1. Part 6.

XOR BX, FFFF	Inversion BX	BX:FFE9	BX: 0000 0000 0001 0110 XOR 1111 1111 1111 1111 <hr/> 1111 1111 1110 1001 F F E 9
ADD BX, 0x0001	Add 1d to BX	BX:FF EA	BX:1111 1111 1110 1010 F F E A
ADD AX, BX	Add AX, BX	AX: FFE2	AX: 1111 1111 1111 1000 + BX: 1111 1111 1110 1010 <hr/> AX: 1111 1111 1110 0010 F F E 2
XOR AX, 7FFF	Inversion AX without high byte	AX:80 1E	AX:1111 1111 1110 0010 XOR <hr/> 0111 1111 1111 1111 <hr/> 1000 0000 0001 1101
ADD AX,0001		$11110_2 =$ $= 30_{10}$ $C_{10} = 30$	\oplus <hr/> 1000 0000 0001 1101 0000 0000 0000 0001 1000 0000 0001 1110

ПРИКЛАД 2.

Визначте вміст регістрів та значення змінної С

Команда	Стан регістру	Примітка
MOV AX, 0x005A	AX:005A	<p>0x???? - змінна в шістнадцятиричній системі числення.</p> <p>005Ah = 0000 0000 0101 1010 = 64+16+8+2 = 90₁₀ =</p>
MOV BX, 00017	BX: 000F	<p>0???? - змінна в восьмиричній системі числення.</p> <p>0017₈ = 8+7=15₁₀</p> <p>0000 0000 0000 1111₂ = 000F_h</p>
MOV CX, C	CX: ????	Визначте значення змінної С
ADD AX, CX	AX: 006E	<p>006E_h = 0000 0000 0110 1110₂ =</p> <p>= 64+32+8+4+2 = 110₁₀</p>
Відповідь:	CX:0014	C = 110 - 90 = 20
ADD CX, BX	CX: 00 23	<p>0000 0000 0001 0100 = 20₁₀</p> <p>0000 0000 0000 1111 = 15₁₀</p> <hr/> <p>0000 0000 0010 0011</p>