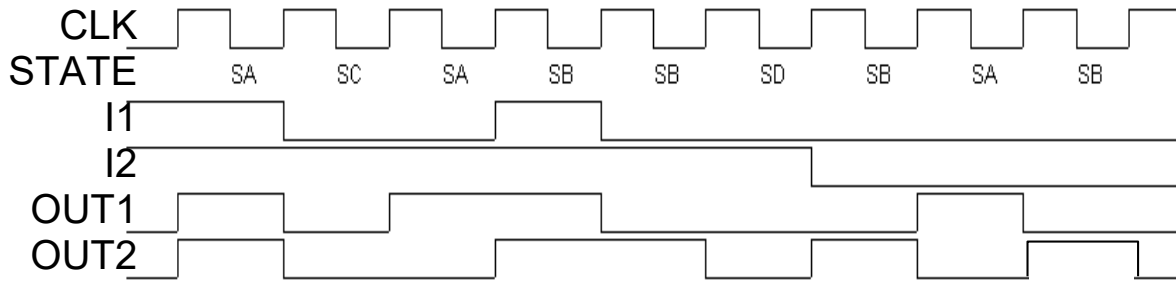
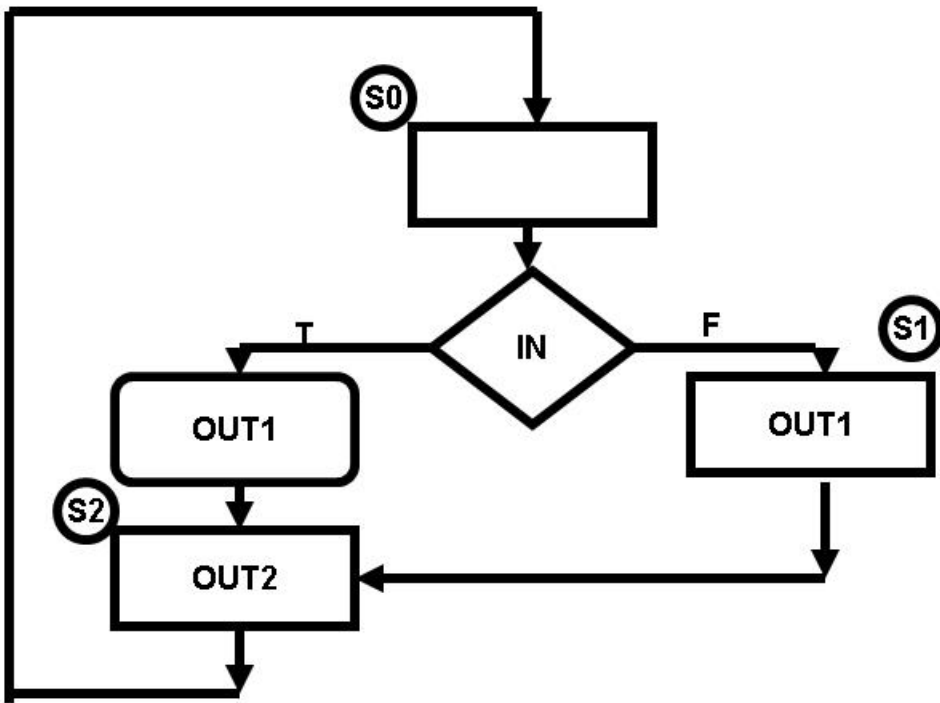


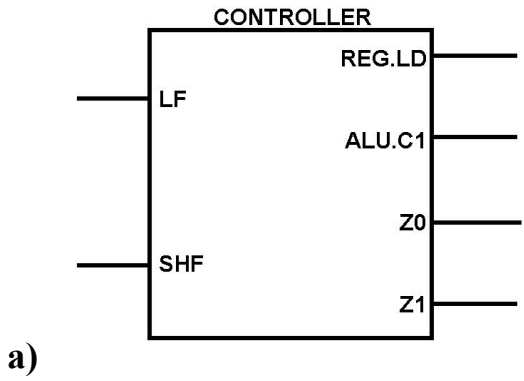
7.3)



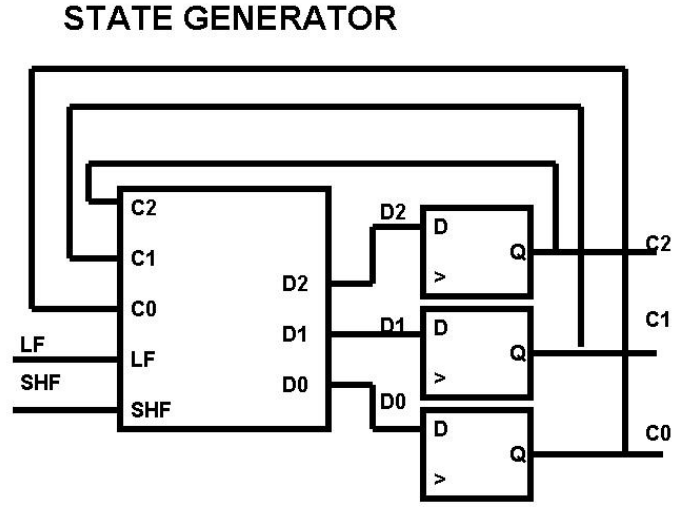
7.5)



7.8)



b)



c)

Q2	Q1	Q0			D2	D1	D0					
C2	C1	C0	LF	SHF	C2+	C1+	C0+	Reg.LD	ALU.C1	Z0	Z1	
0	0	0	0	0	0	0	1	1	0	0	0	
0	0	0	0	1	0	1	0	1	0	1	0	
0	0	0	1	0	0	1	1	1	0	0	0	
0	0	0	1	1	0	1	1	1	0	0	0	
0	0	1	*	*	0	1	0	0	1	0	0	
0	1	0	*	*	0	0	0	0	0	0	0	
0	1	1	*	0	1	0	0	0	1	0	0	
0	1	1	*	1	0	1	1	0	1	0	1	
1	0	0	*	*	0	0	0	0	0	0	0	
1	0	1	*	*	X	X	X	X	X	X	X	
1	1	0	*	*	X	X	X	X	X	X	X	
1	1	1	*	*	X	X	X	X	X	X	X	

d) LOGIC EQUATIONS:

$$\begin{aligned}
 D2 &= C1 C0 / SHF \\
 D1 &= /C2 /C1 LF + /C2 /C1 SHF + /C1 C0 + C0 SHF \\
 D0 &= /C2 /C1 /C0 /SHF + /C2 /C1 /C0 LF + C1 C0 SHF \\
 REG.LD &= /C2 /C1 /C0 \\
 ALU.C1 &= /C2 /C1 C0 + /C2 C1 C0 = /C2 C0 (/C1 + C1) = /C2 C0 \\
 Z0 &= /C2 /C1 /C0 /LF SHF \\
 Z1 &= /C2 C1 C0 SHF
 \end{aligned}$$

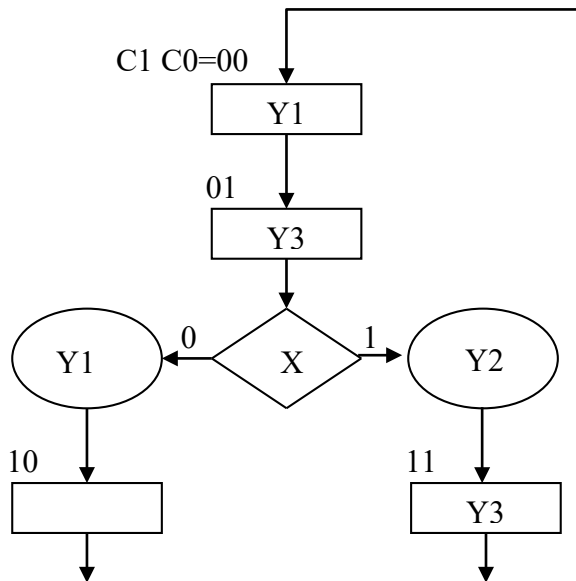
7.15)

WORKING BACKWARDS, WE OBTAIN THE FOLLOWING NEXT-STATE/OUTPUT TABLE:

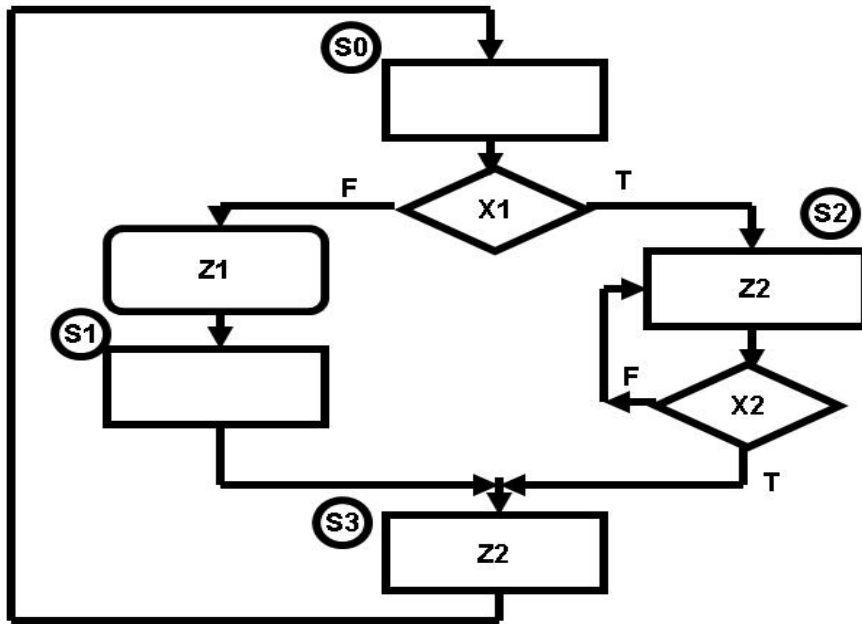
A2	A1	A0	Z4	Z3	Z2	Z1	Z0
C1	C0	X	Y1	Y2	Y3	C1+	C0+
0	0	0	1	0	0	0	1
0	0	1	1	0	0	0	1
0	1	0	1	0	1	1	0
0	1	1	0	1	1	1	1
1	0	0	0	0	0	0	0
1	0	1	0	0	0	0	0
1	1	0	0	0	1	0	0
1	1	1	0	0	1	0	0

FROM THE TABLE, WE CAN OBTAIN THE FOLLOWING ASM CHART:

b)

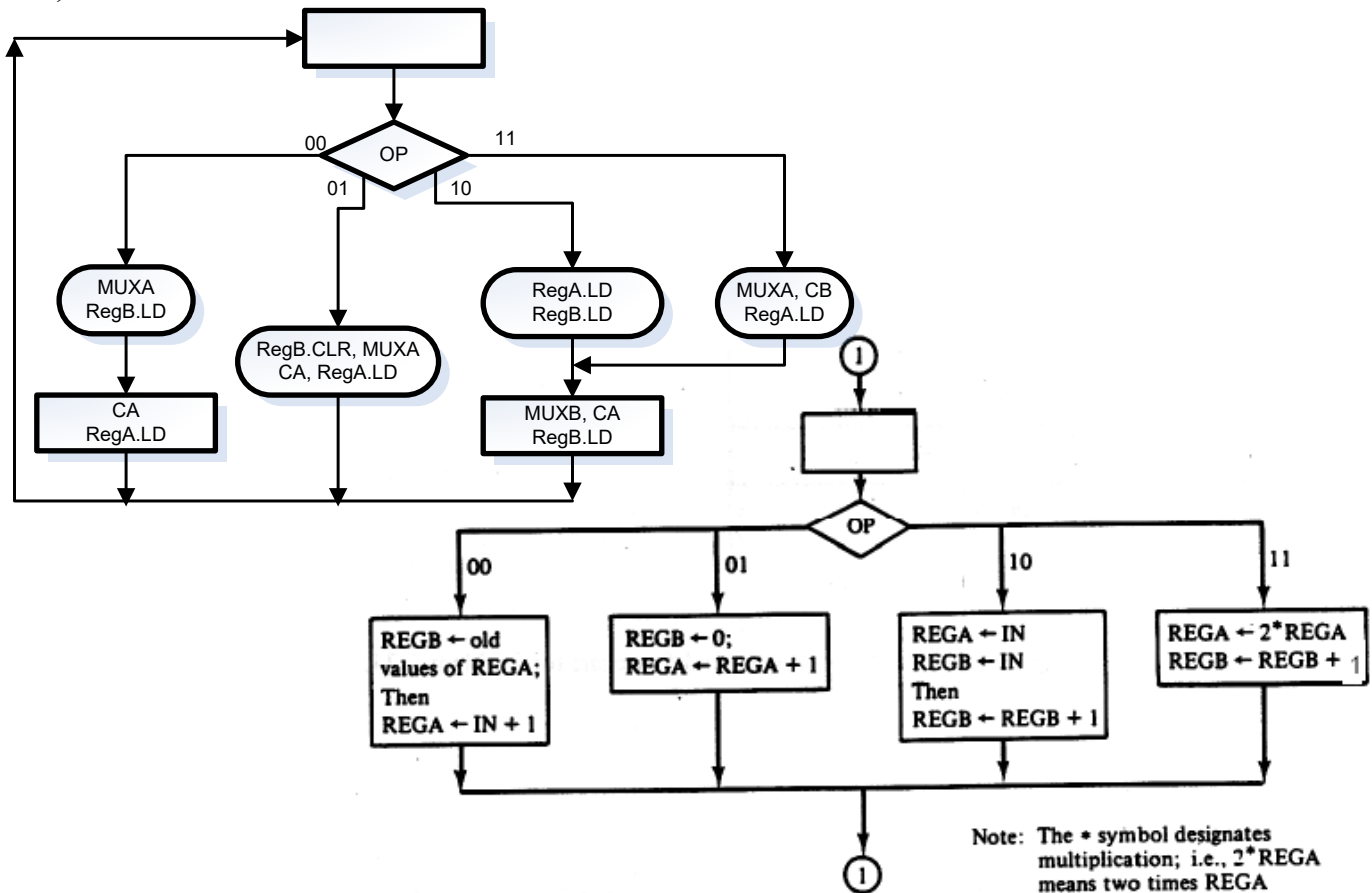


7.18)



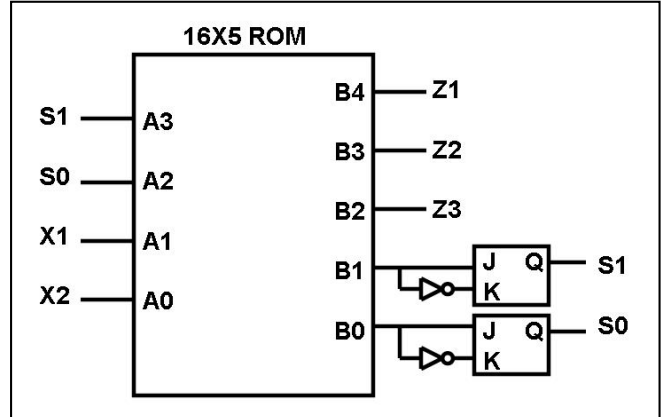
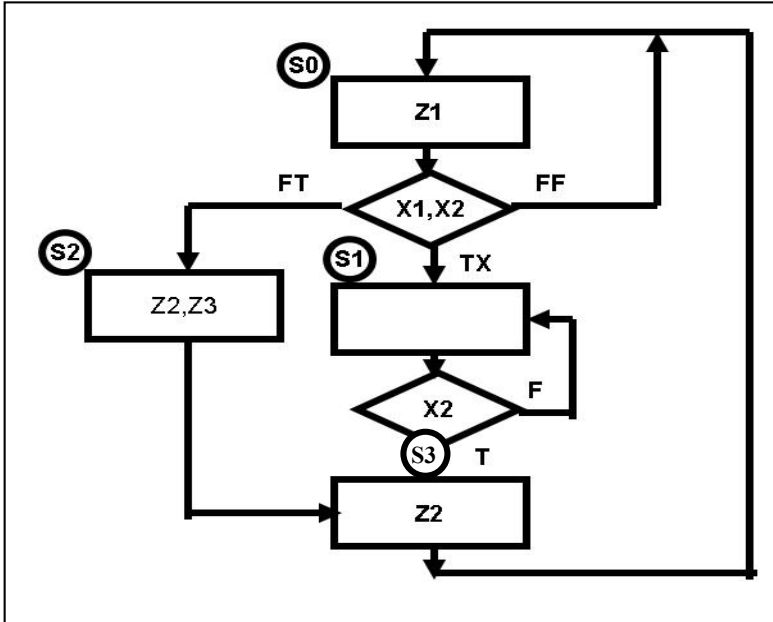
7.19) See next page

7.23)



Homework 10 Solutions
 Lam Chapter 7 --- Revision 0

7.19)



LOCATION:			CONTENTS			
S1	S0	X1	Z1	Z2	Z3	J1 J0
A3	A2	A1	B4	B3	B2	B1 B0
00	00		1	0	0	00
00	01		1	0	0	10
00	10		1	0	0	01
00	11		1	0	0	01
01	00		0	0	0	01
01	01		0	0	0	11
01	10		0	0	0	01
01	11		0	0	0	11
10	00		0	1	1	11
10	01		0	1	1	11
10	10		0	1	1	11
10	11		0	1	1	11
11	00		0	1	0	00
11	01		0	1	0	00
11	10		0	1	0	00
11	11		0	1	0	00

In a general state machine, the J's and K's **should be done separately**, possibly eliminating the need for the inverters. But the above solution would work. Doing the J's and K's separately would require two more ROM data bits (B5 and B6). But since most available ROMs are 8-bits wide, this solution would cost less (eliminating the two inverters). Therefore, the below solution is better, i.e., worth more points!!!

