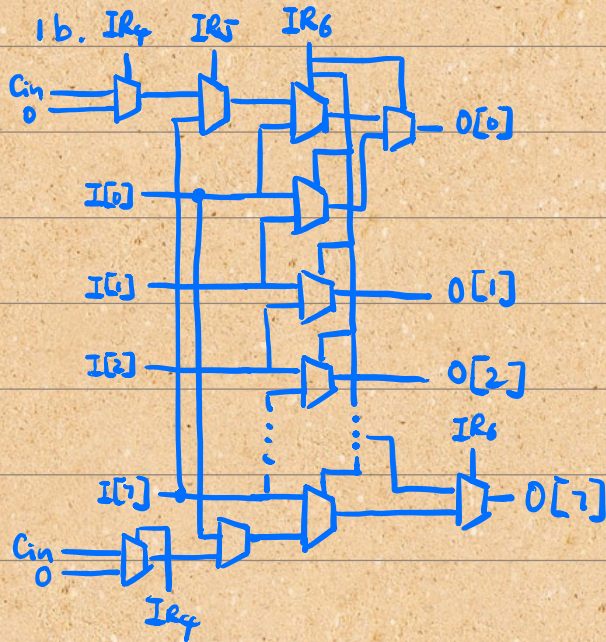


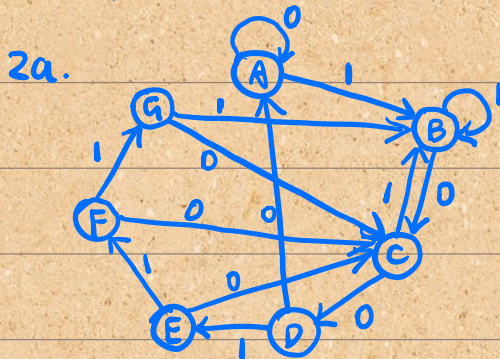
This is needed because when doing subtraction with two's complements, we need to invert the second number and add 1 to it.



1c.

State	Source	Destination	Purpose
1	Data In	IR	The first instruction byte is loaded
2	Data In	A	The first number is loaded into register A
3	Data In, ALU	B, C	Second # \rightarrow B, C is set to 0
4	Data In	IR	The second instruction byte is loaded
5	ALU	RES, C	Result becomes A plus B, carry-out computed
1	Data In	IR	The first instruction byte is loaded
2	RES	A	Result from last cycle loaded into A
3	ALU	C	C is set to 0
4	Data In	IR	second instruction loaded to IR
5	Shifter	RES	Result becomes A shift right by 1

1d. Cycle 1 state 2: $IR^7 \quad IR^0$
 Cycle 1 state 5: $IR^7 \quad IR^0$
 Cycle 2 state 2: $IR^7 \quad IR^0$
 Cycle 2 state 5: $IR^7 \quad IR^0$



2b.

Input	This State	Q2	Q1	Q0	Next State	D2	D1	D0
0	A	0	0	0	A	0	0	0
0	B	0	0	1	C	0	1	0
0	C	0	1	0	D	0	1	1
0	D	0	1	1	A	0	0	0
0	E	1	0	0	C	0	1	0
0	F	1	0	1	C	0	1	0
0	G	1	1	0	C	0	1	0
0	X	1	1	1	X	X	X	X
1	A	0	0	0	B	0	0	1
1	B	0	0	1	B	0	0	1
1	C	0	1	0	B	0	0	1
1	D	0	1	1	E	1	0	0
1	E	1	0	0	F	1	0	1
1	F	1	0	1	G	1	1	0
1	G	1	1	0	B	0	0	1
1	X	1	1	1	X	X	X	X

2c. K-map for D_2 :

Input, Q_2	Q_1, Q_0			
	00	01	11	10
00	0	0	0	0
01	0	0	X	0
11	1	1	X	0
10	0	0	1	0

$$D_2 = \text{Input} \cdot Q_2 \cdot Q_1' + \text{Input} \cdot Q_1 \cdot Q_0 = \text{Input} \cdot (Q_2 \cdot Q_1' + Q_1 \cdot Q_0)$$

K-map for D_1 :

Input, Q_2		Q_1, Q_0			
		00	01	11	10
00	0	0	1	0	1
01	1	1	1	X	1
11	0	0	1	X	0
10	0	0	0	0	0

$$D_1 = I' \cdot Q_1 \cdot Q_0' + I' \cdot Q_2 \cdot Q_1' + I' \cdot Q_1' \cdot Q_0 + Q_2 \cdot Q_0$$

K-map for D_0 :

Input, Q_2		Q_1, Q_0			
		00	01	11	10
00	0	0	0	0	1
01	0	0	0	X	0
11	1	1	1	0	1
10	1	1	0	X	1

$$D_0 = I \cdot Q_2 \cdot Q_1' + I \cdot Q_1' \cdot Q_0' + I \cdot Q_1 \cdot Q_0' + Q_2' \cdot Q_1 \cdot Q_0'$$

2d. K-map for output

Q_2	Q_1, Q_0			
	00	01	11	10
0	0	0	0	0
1	0	0	0	1

$$\text{Output} = Q_2 \cdot Q_1 \cdot Q_0'$$

2e. When designing state sequencing logic, we decide when input=0, unused state goes to state 010, which is state C.

When input=1, unused state goes to state 110, which is state G.