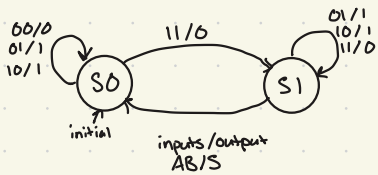
HW 1

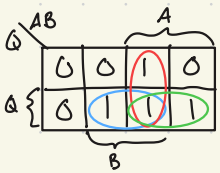


A	B	PS	S	C^+	NS
0	0	0	0	0	S0
0	1	0	1	0	S0
1	0	0	1	0	S0
1	1	0	0	1	S1
0	0	1	1	0	S0
0	1	1	0	1	S1
1	0	1	0	1	S1
1	1	1	0	1	S1

State 0: carry = 0 (initial state)
State 1: carry = 1

inputs: A + B
output: S

Cin Q	A	B	S	Cout Q'
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



$Q' = AB + AQ + BQ$

$= A \oplus B \oplus Q$

$= \Sigma(3, 5, 6, 7)$

← simplest AND/OR/NOT implementation

make verification table using example

Clk Cycle | A | B | Q | S(SG) | S(Diagram) | Q^+

A_2 Q	A_1 A	A_0 B	O_1 Q^+	O_0 S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	0	1
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

ROM Implementation

input: A+B

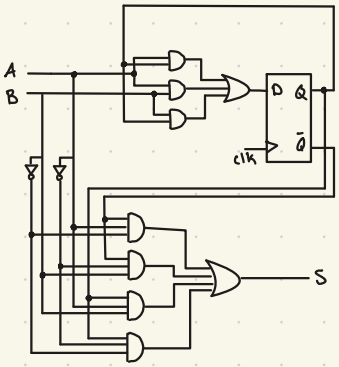
state: Q (1 FF)

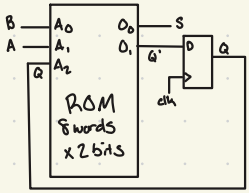
address bits: QAB

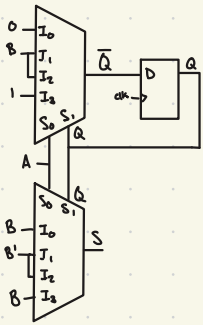
8 addresses

(outputs) data bits: Q^+ + S

ROM size: 8x2





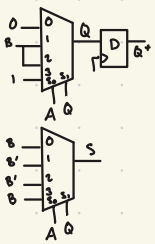


Mux Implementation

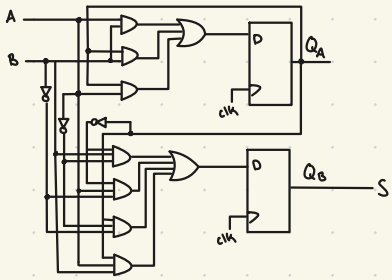
state = carry stored in D_ff

inputs: A B

output: S



select lines							
Q	A	B	S		Q^{+1}		
0	0	0	0	B	0	0	0
0	0	1	1		0		
0	1	0	1	B'	0	B	1
0	1	1	0		1		
1	0	0	1	B'	0	B	2
1	0	1	0		1		
1	1	0	0	B	1	1	3
1	1	1	1		1		



Moore Implementation

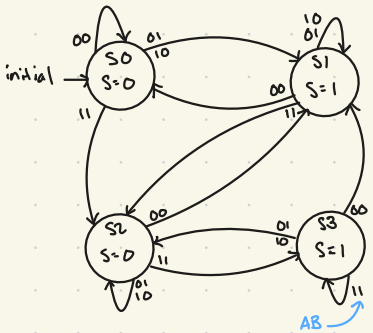
State Transition Table

	Current Q_A	Current Q_B	A	B	Next Q_A^+	Next Q_B^+	S
S0	0	0	0	0	0	0	0
	0	0	0	1	0	1	
	0	0	1	0	0	1	
	0	0	1	1	1	0	
S1	0	1	0	0	0	0	1
	0	1	0	1	0	1	
	0	1	1	0	0	1	
	0	1	1	1	1	0	
S2	1	0	0	0	0	1	0
	1	0	0	1	1	0	
	1	0	1	0	1	0	
	1	0	1	1	1	1	
S3	1	1	0	0	0	1	1
	1	1	0	1	1	0	
	1	1	1	0	1	0	
	1	1	1	1	1	1	

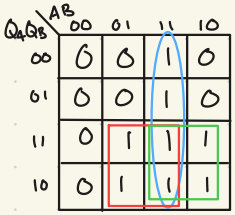
Simplified State Table

Current	Next AB 00	01	10	11	Output S
S0	S0	S1	S1	S2	0
S1	S0	S1	S1	S2	1
S2	S1	S2	S2	S3	0
S3	S1	S2	S2	S3	1

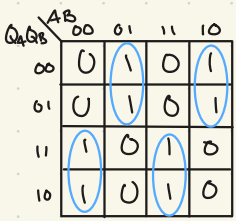
C	S	State
0	0	S0
0	1	S1
1	0	S2
1	1	S3



$Q_A^+ = AB + Q_AB + Q_AA$



$Q_B^+ = Q_AAB + Q_AA'B' + Q_AA'B + Q_A'AB'$



$S = Q_B$