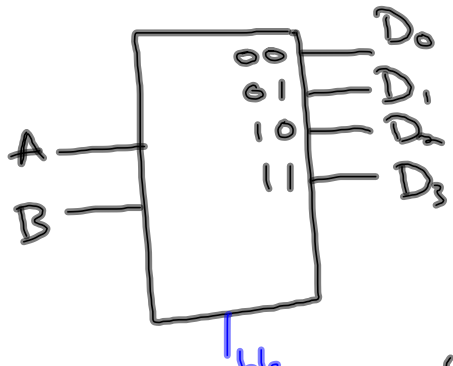
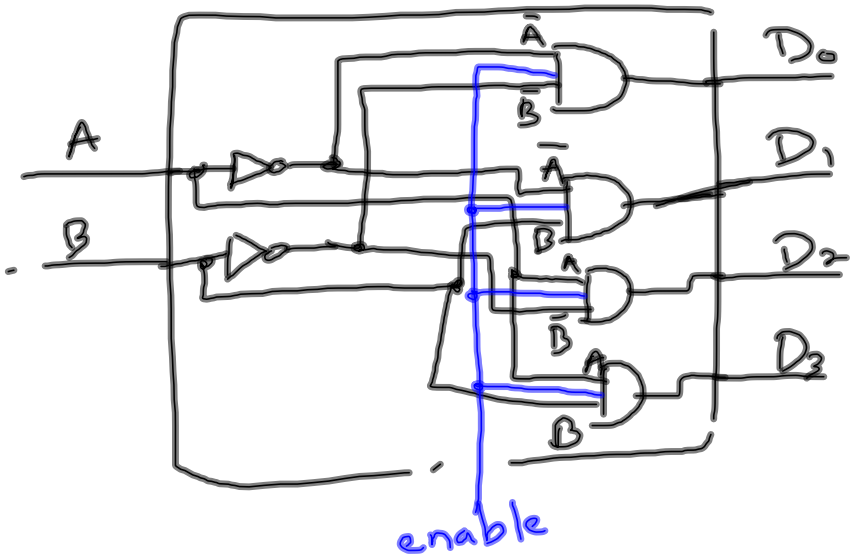
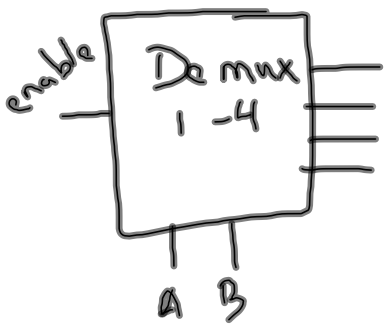
Decoder

2-to-4

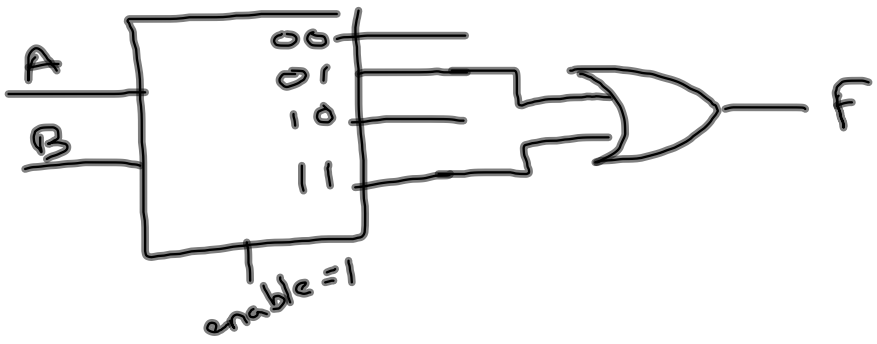


enable e	A	B	D_0	D_1	D_2	D_3
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1



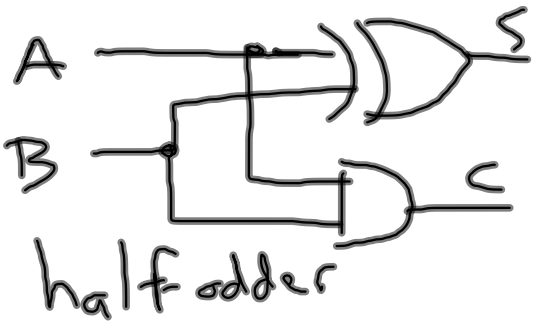


A	B	F
0	0	0
0	1	1
1	0	0
1	1	1



$$\begin{array}{r} 0 \\ +\,0 \\ \hline 0\,0 \end{array} \qquad \begin{array}{r} 0 \\ +\,1 \\ \hline 0\,1 \end{array} \qquad \begin{array}{r} 1 \\ +\,0 \\ \hline 0\,1 \end{array} \qquad \begin{array}{r} 1 \\ +\,1 \\ \hline 1\,0 \end{array}$$

A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

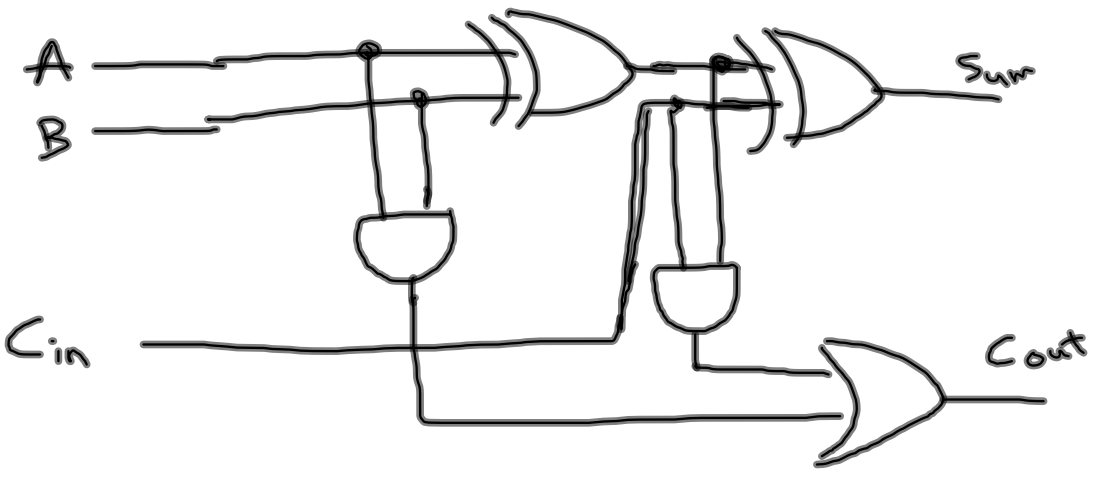


halfadder

```
  1 1 1
  0 1 0 1      5
+ 0 0 1 1      3
---------
  1 0 0 0      8
```

A	B	C_{in}	C_{out}	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

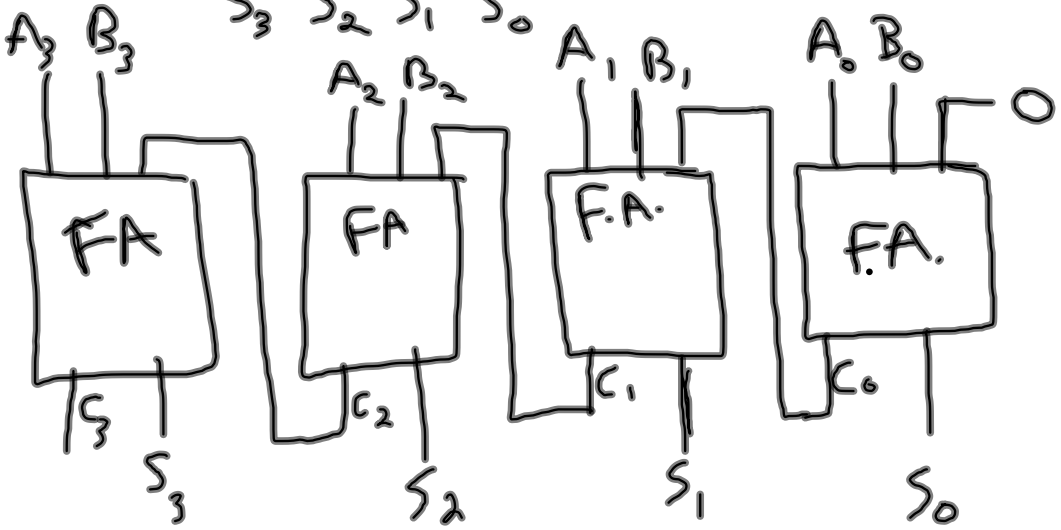
Full Adder

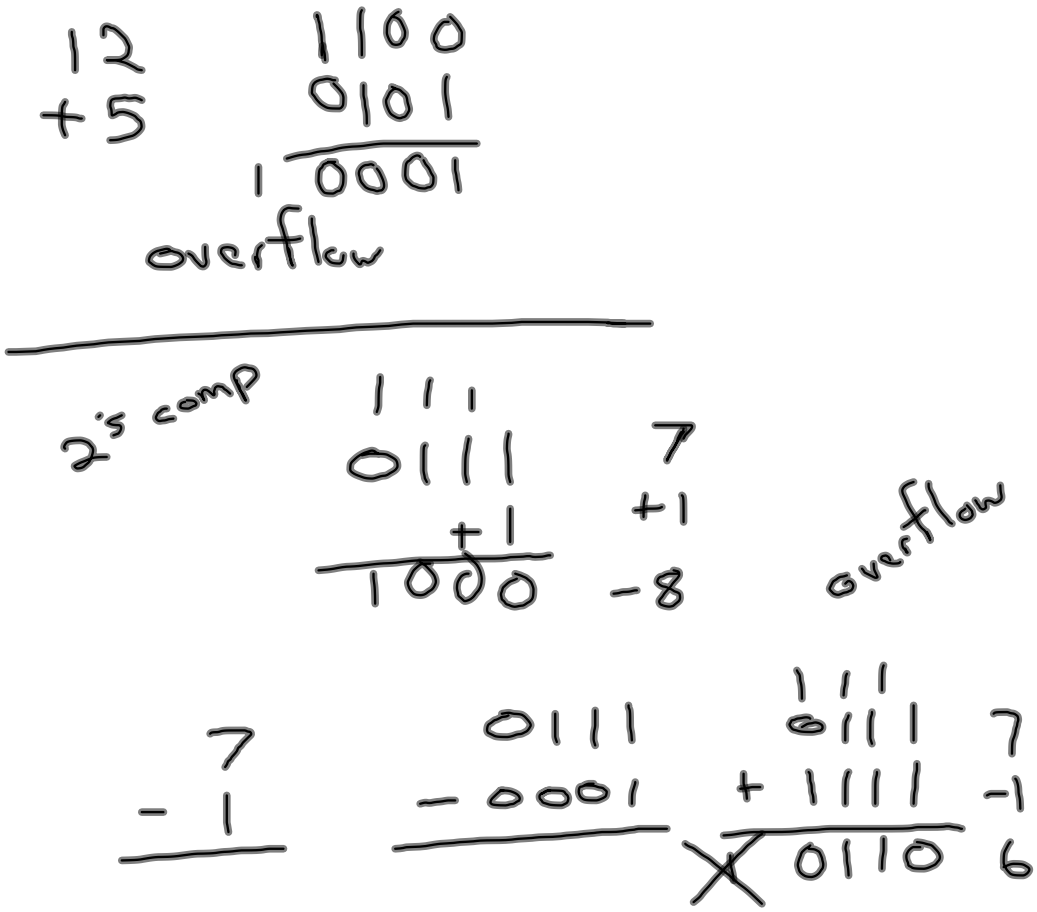


$A_3\ A_2\ A_1\ A_0$ } 4 b.t #'s

$+\ B_3\ B_2\ B_1\ B_0$

$S_3\ S_2\ S_1\ S_0$



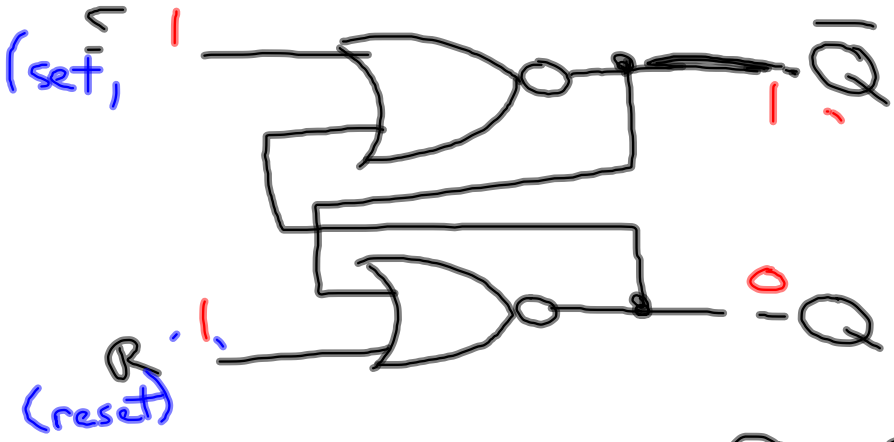

12
+5
1100
0101
10001
overflow
2's comp
111
0111
7
+1
1000
-8
overflow
7
- 1
0111
- 0001
111
0111
7
+ 1111
-1
0110
6

Combinational Logic

- output is determined by input

Sequential Logic

- output is determined by input and "state" (memory)

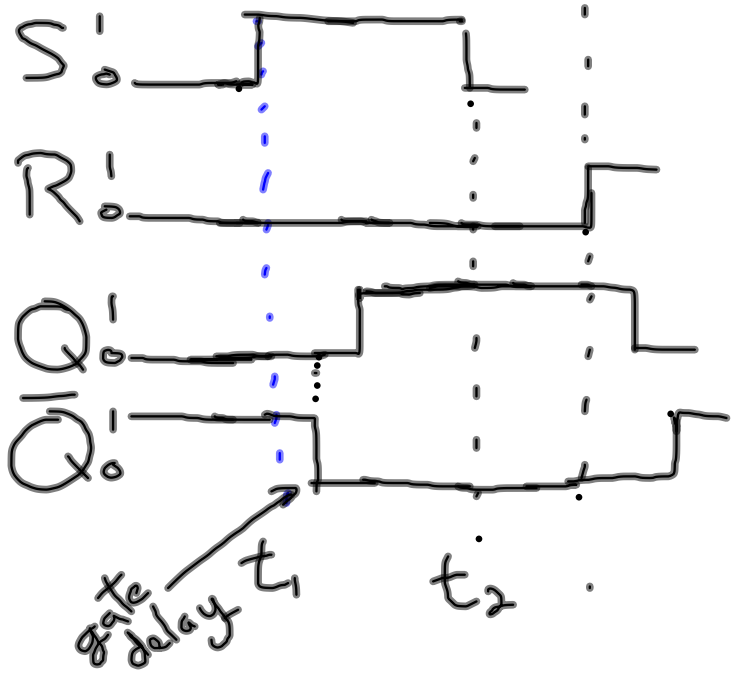


A	B	A NOR B
0	0	1
0	1	0
1	0	0
1	1	0

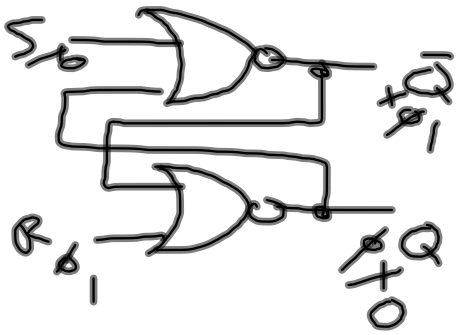
Q_t	S	R	Q_{t+1}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	X
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	X

R	S	Q
0	0	Q
0	1	1
1	0	0
1	1	X

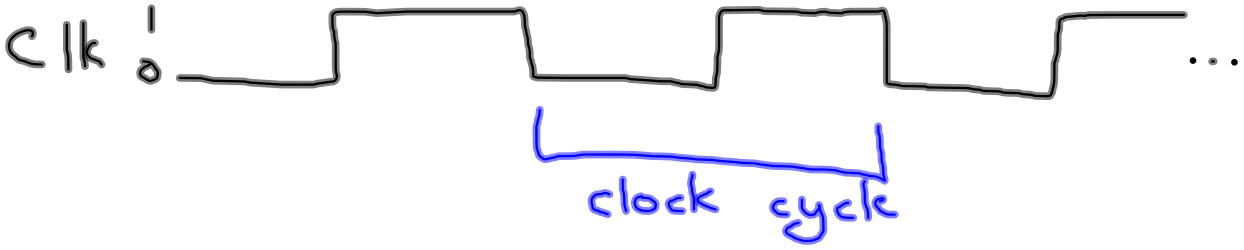
Timing Diagrams:



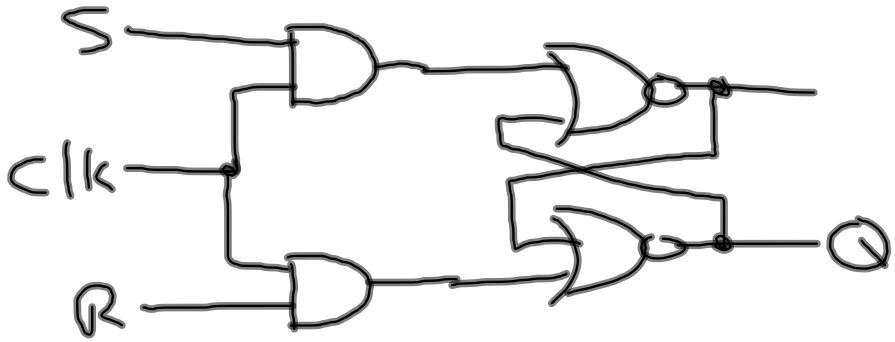
S-R Latch



Clock



Clocked S-R latch



D latch