1. How Hamming(7,4) works

111	110	101	100	011	010	001
7	6	5	4	3	2	1
D3	D2	D1	C2	D0	C1	C0

   a. Each data bit must be covered (checked) by at least 2 parity bits

   b. Why we start numbering from 1

   c. Which parity bits check which data bits
      i. C0

      ii. C1

      iii. C2

      iv. If we had a C3

      v. In general

   d. Why we can XOR to determine the bit position

   e. Expanding it further
      i. Expanding to cover a larger data size

      ii. Covering more errors
         1. Given T errors

2. Sequential circuits
    a. So far, only discussed combinational circuits

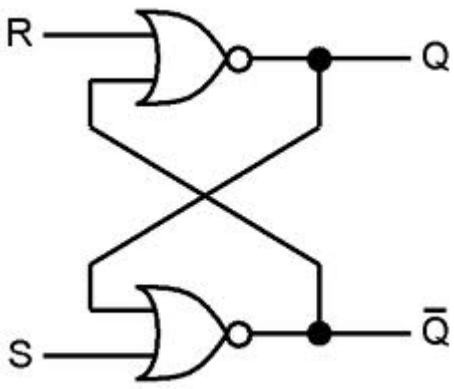
    b. Sequential circuits

    c. Examples

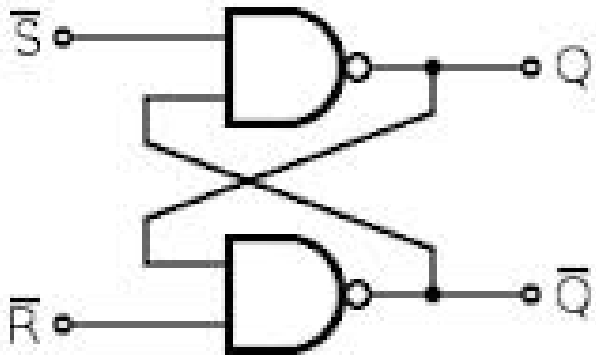
3. Latches

    a. S-R latches

    b. S-R latch implementation



S-R NOR Latch



S-R NAND Latch

    c. Characteristic tables

S	R	Q_{n+1}
0	0	
0	1	
1	0	
1	1	

S-R NOR Latch

S	R	Q_{n+1}
1	1	
1	0	
0	1	
0	0	

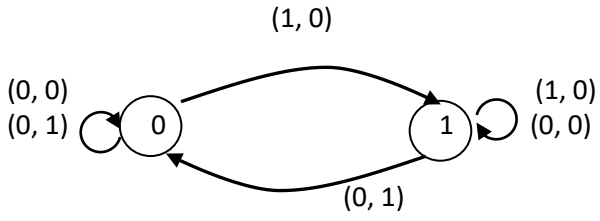
S-R NAND Latch

d. State transition table for S-R NOR latch

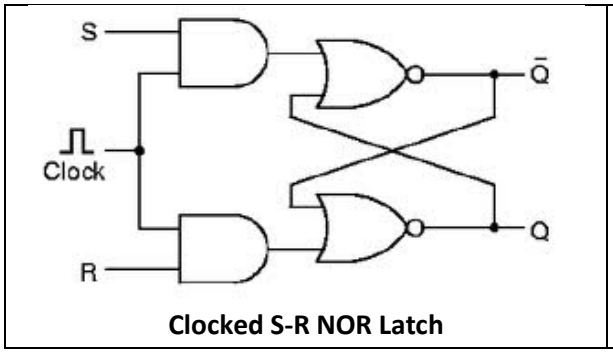
State Transition Table for S-R NOR Latch				
Present Inputs		Present State	Next State	Type of Circuit
S	R	Q	Q'	
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

e. State transition diagram
   i. Visual representation of circuit's output based on current state and input

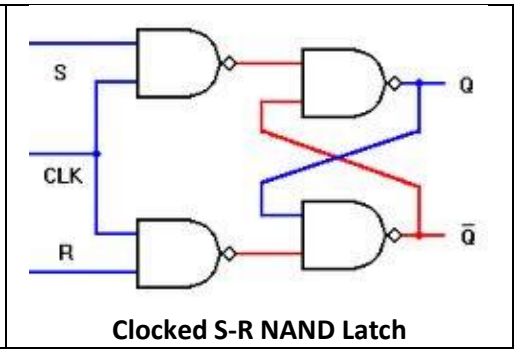
   ii. For every state, all possible output combinations need to be listed



4. Other latches
    a. Clocking the latches

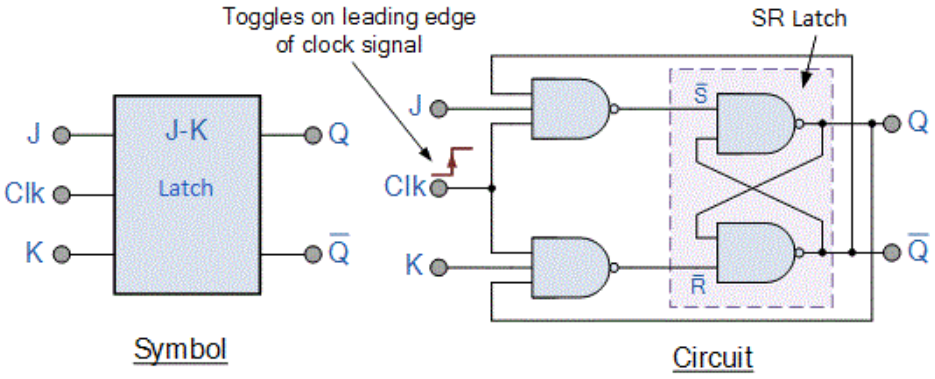


Clocked S-R NOR Latch

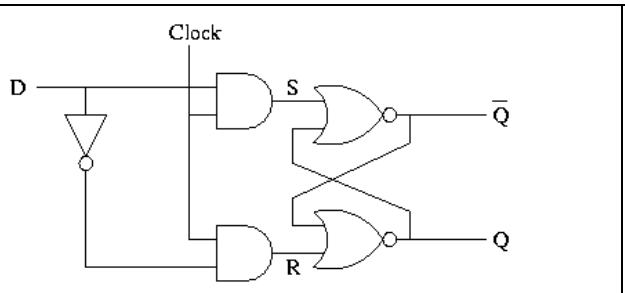


Clocked S-R NAND Latch

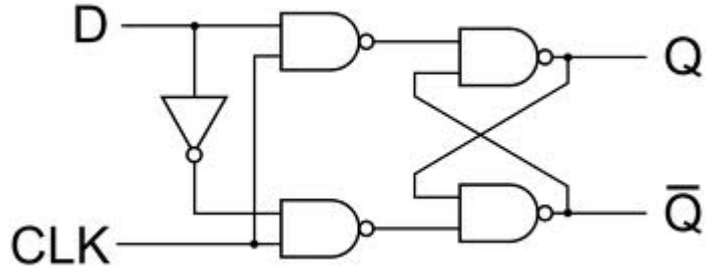
    b. Clocked J-K latch
        i. Like a S-R latch



    b. Clocked D latch



Clocked NOR D Latch



Clocked NAND D Latch