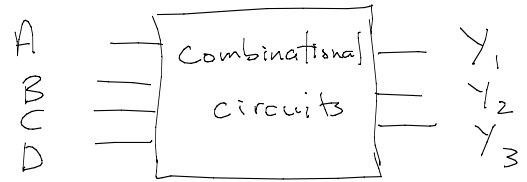


lecture 5

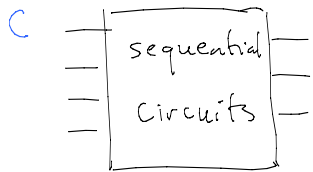
Sequential circuits I

last week



- signals carried on wires
- gate operations take time (e.g. carries in addition)

this week

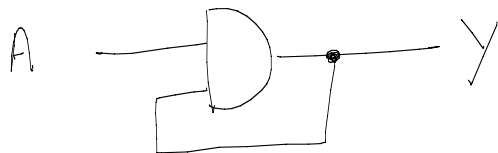


- Combinational circuits + memory
- Synchronized by a discrete clock C

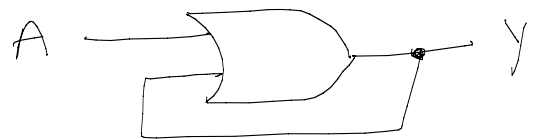
Memory (2 kinds)

- write it down
- repeat to yourself (feedback)

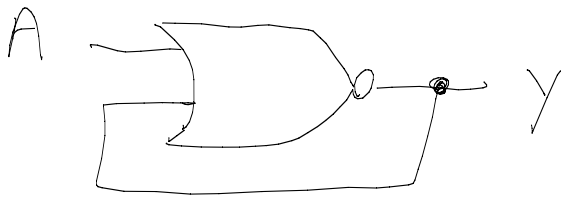
Feedback



A	Y
0	0
1	0 or 1
0 → 1	0
1 → 0	0



A	Y
0	0 or 1
1	1
0 → 1	1
1 → 0	1



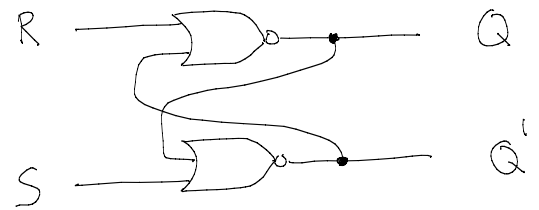
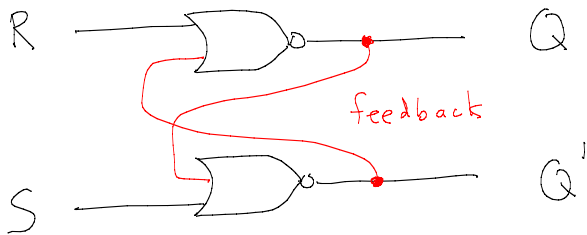
A	B	$\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

A	Y
0	0 ↔ 1 oscillate
1	0

Today

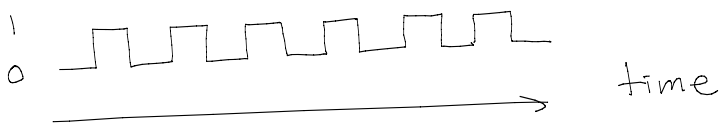
- RS latch
- clocked RS latch
- D latch
- D flip flop, T flip flop
- register

RS latch  
(‘reset’ 0, ‘set’ 1)



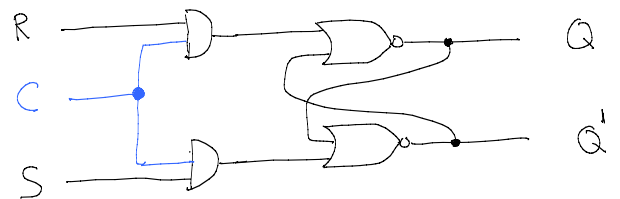
R	S	Q	Q'
0	0	hold previous values	
0	1	1	0
1	0	0	1
1	1	0	0

“Clock”



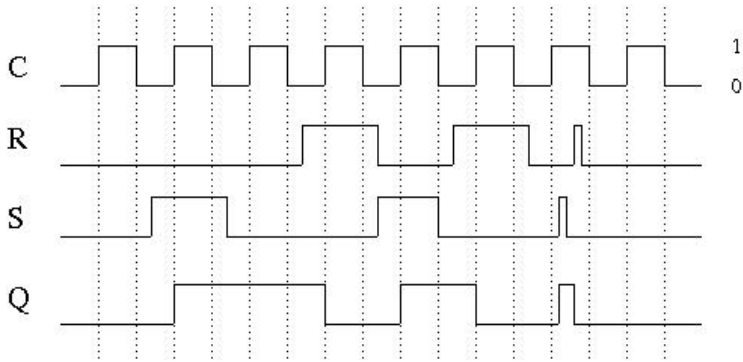
- doesn't count (just goes ON, OFF, ...)
- electronic implementation not discussed
- typical clock speed today is 3 GHz (10<sup>9</sup> cycles/sec)

Clocked RS latch



C=0				C=1			
R	S	Q	Q'	R	S	Q	Q'
0	0	hold		0	0	hold	
0	1	1	0	0	1	1	0
1	0	0	1	1	0	0	1
1	1	0	0	1	1	0	0

### Example



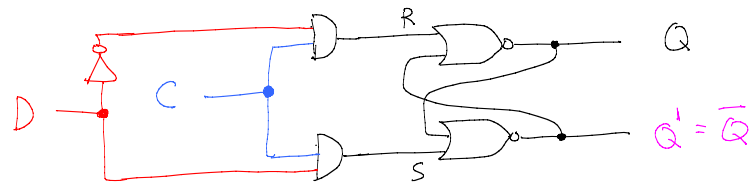
$C=0$

R	S	Q	Q'
0	0	hold	hold
0	1	0	1
1	0	1	0
1	1	hold	hold

$C=1$

R	S	Q	Q'
0	0	hold	hold
0	1	0	1
1	0	1	0
1	1	hold	hold

### D latch ("D" for data)

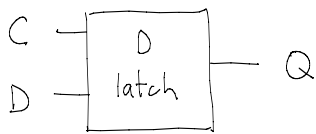
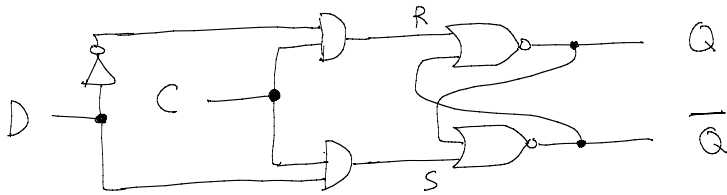


What does this circuit do?

when  $C=1$ ,  $D=1 \Rightarrow Q=1$   
 $D=0 \Rightarrow Q=0$

when  $C=0$ , hold values of  $Q, \bar{Q}$

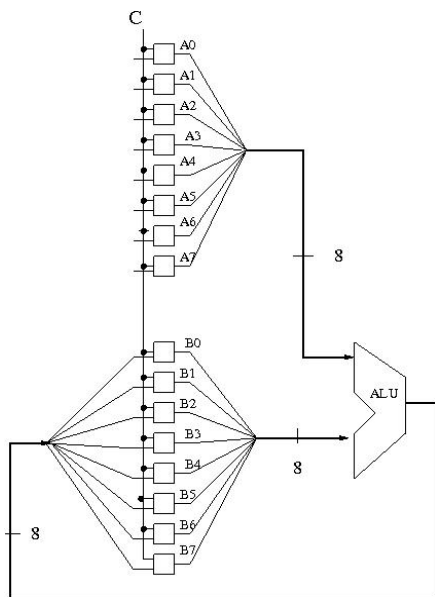
### D latch



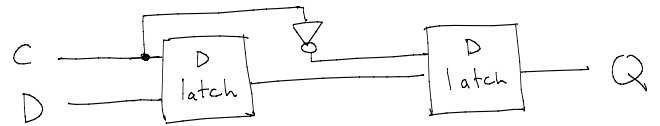
$C=0 \Rightarrow$  read only  
 $C=1 \Rightarrow$  read and write

- We can hold values in D latches when  $C=0$ .
- But when  $C=1$ , the circuits are open.
- This becomes a problem when we want feedback in circuits containing D latches

$$x = x + y$$



### D flip flop



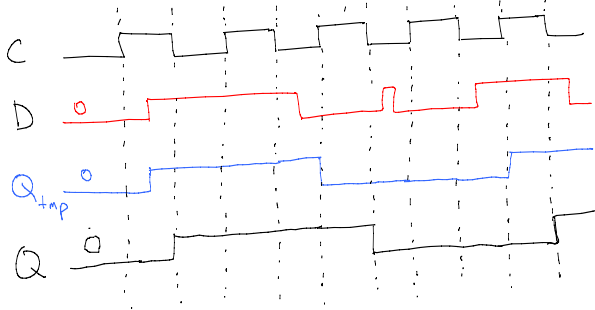
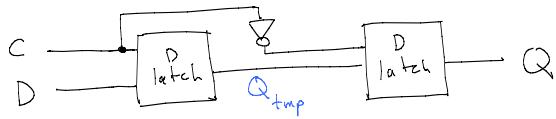
$C=1 \Rightarrow$

- write D into first D latch
- Q holds previous value

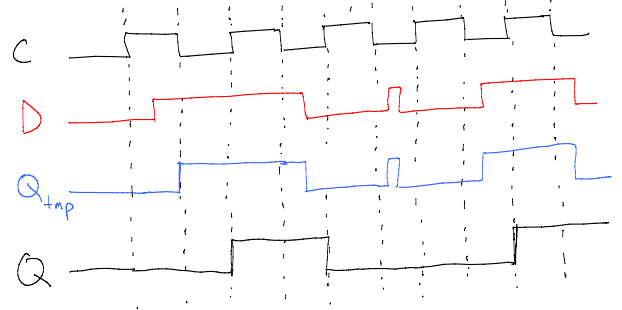
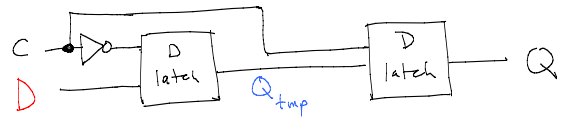
$C=0 \Rightarrow$

- stop writing into 1<sup>st</sup> D latch
- copy value from 1<sup>st</sup> to 2<sup>nd</sup> D latch, Q takes (possibly) different value

## D flip flop (falling edge triggered)

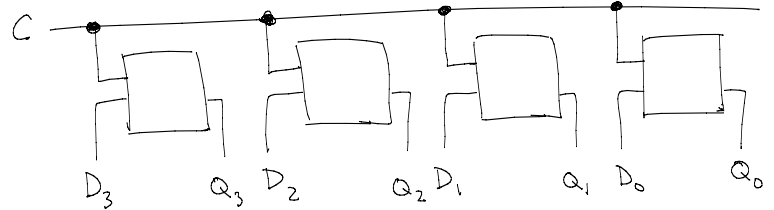


## D flip flop (rising edge triggered)



- Clock cycle is long enough to allow all gates to stabilize.
- Clock synchronizes all flipflops, allowing us to treat time as a sequence of discrete R/W steps (hence "sequential circuit")
- From now on, ignore all variations within a clock cycle (e.g. carries in adder)

## Register (set of flip flops that are grouped together)



## Shift register (falling edge)

