Flip-flop and Latches

Because race conditions will result in undefined behavior, maybe we can use them as real random number generators. *(unpractical)*

Latches

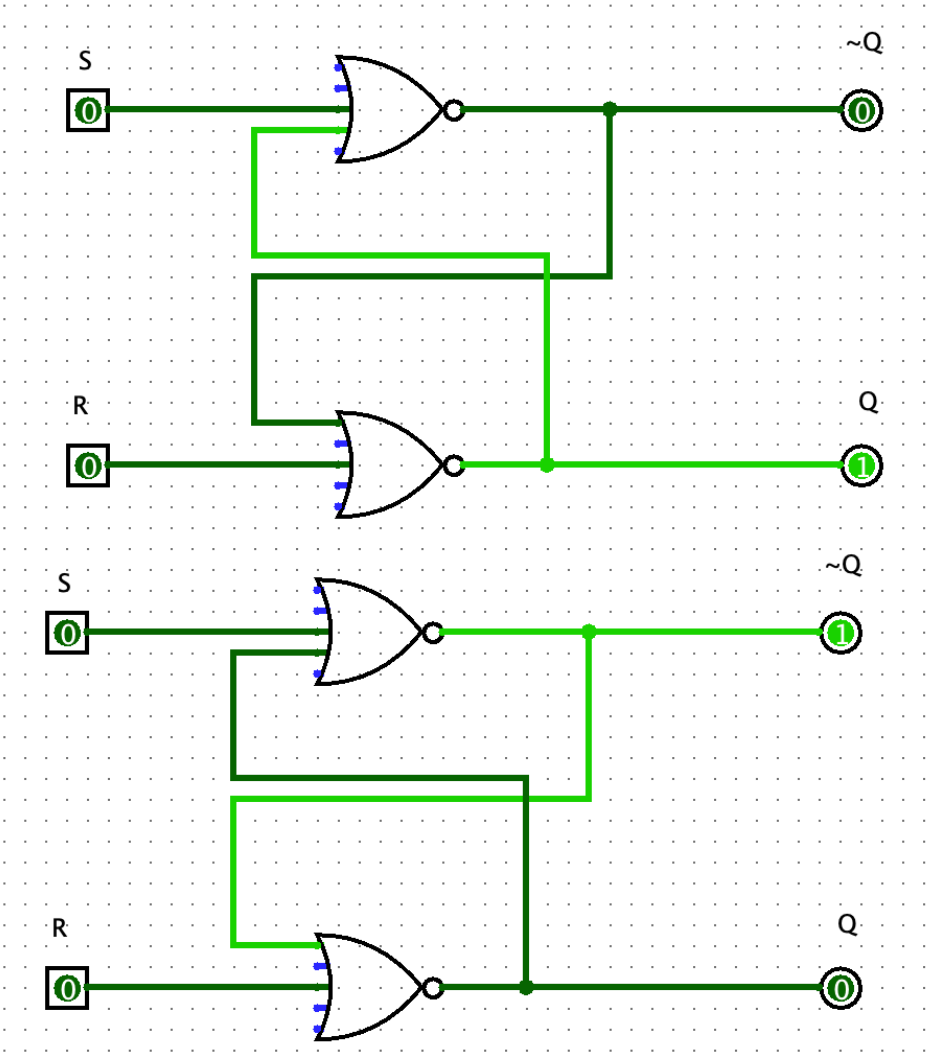
- a bistable multivibrator, that is, a device with **exactly two stable states**(high-output and low-output)
- A latch has a **feedback path**, so information can be retained by the device.
- Therefore latches can be memory devices, and can **store one bit of data for as long as the device is powered**
- latches are **used to "latch onto" information and hold in place**
- **not synchronous** devices, and **do not operate on clock edges** as flip-flops do

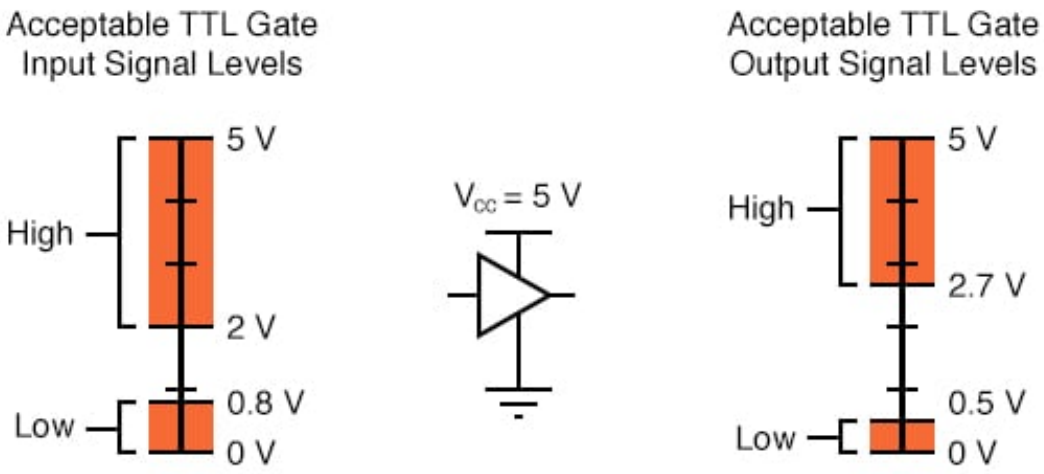
S-R latch

(set-reset latch)

- it can "memorize" a bit

- At the very begining, both S and R are 0, another two inputs are also 0(voltage under certain amount are all considered to be 0)





Both NOR gates want to output a 1, the faster one will make the slower one output 0. In reality, we don't know which one will be faster, so the result can be either the first one or the second one.

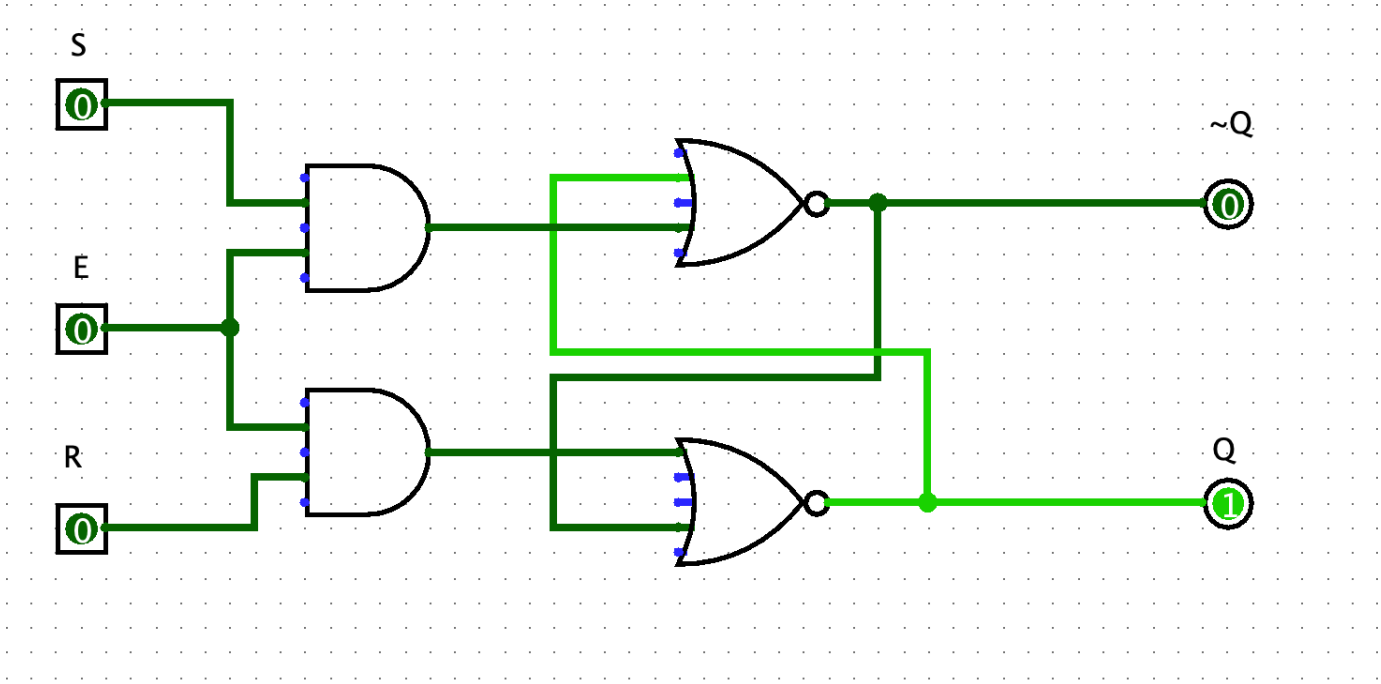
- Q is the bit the latch currently store, $\bar{Q}$ is always the inverse of Q
- boolean expression
  - $Q^{n+1} = \neg(R \vee \neg(Q^n \vee S)) = S + \bar{R}Q$
  - $\bar{Q}^{n+1} = \neg(S \vee \neg(R \vee \bar{Q}^n)) = \neg(S \vee \bar{R} \wedge Q^n) = \bar{S}R + \bar{Q^n}$
  - $SR = 0$
  - it can be simplified via K-map or De-Morgan's law
- "Truth table"

S	R	Q^n	$\bar{Q}^n$	Q^{n+1}	$\bar{Q}^{n+1}$
0	0	1	0	1	0
0	0	0	1	0	1
1	0	1	0	1	0
1	0	0	1	1	0
0	1	1	0	0	1
0	1	0	1	0	1
1	1	1	0	0	0
1	1	0	1	0	0

*(1,1) means to set and reset the latch simultaneously, this will result in racing condition, whose behavior is undefined, so this kind of input is **invalid***

- SR latch can also be built with NAND gate(positive low)

Gated SR latch

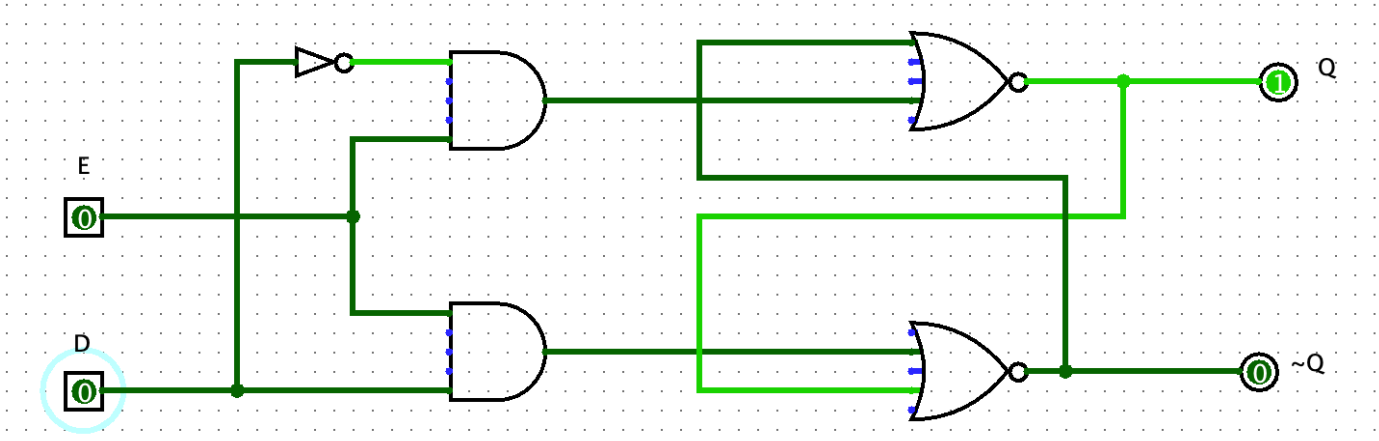


- just add two AND gates to control the input
- E stand for enable, if E is 1, the value of S and R can be sent into the SR latch, otherwise the inputs are "gated".

D latch

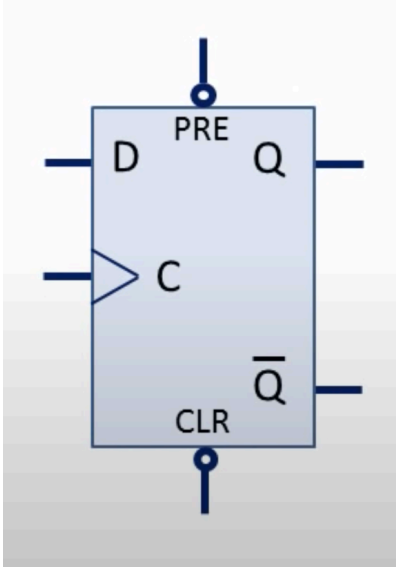
"Data latch"

we want a latch which can automatically switch its status with only one input

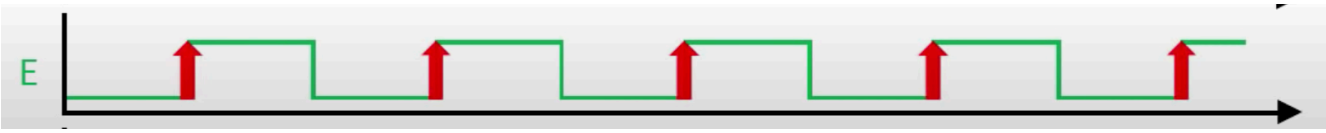


- just add an inverter for gated S-R latch, therefore we can input 1,0 or 0,1 with only one input.
- when E is 1, D input can set the status of the latch, otherwise the input is ignored.

Clocked D latch(D Flip-Flop)

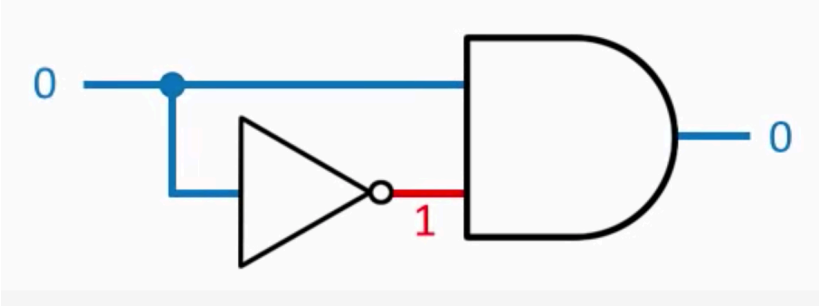


- in computer, everything is controlled by the clock pulse, so does the enable of D latch

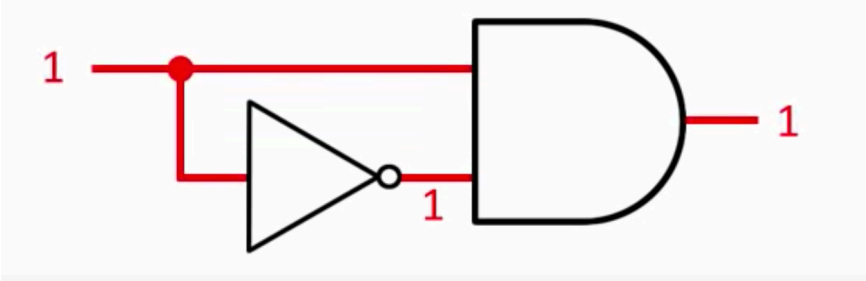


  However, the time period where D latch can be set is still too long, we only want a very tiny time slice, for instance, the positive edge.

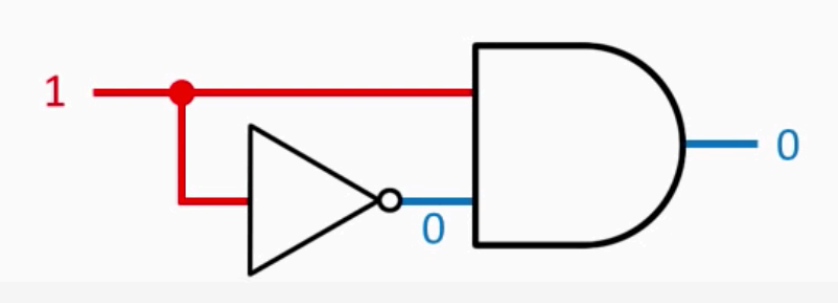
- So how can we detect a rising edge? We take advantage of the delay of gates.



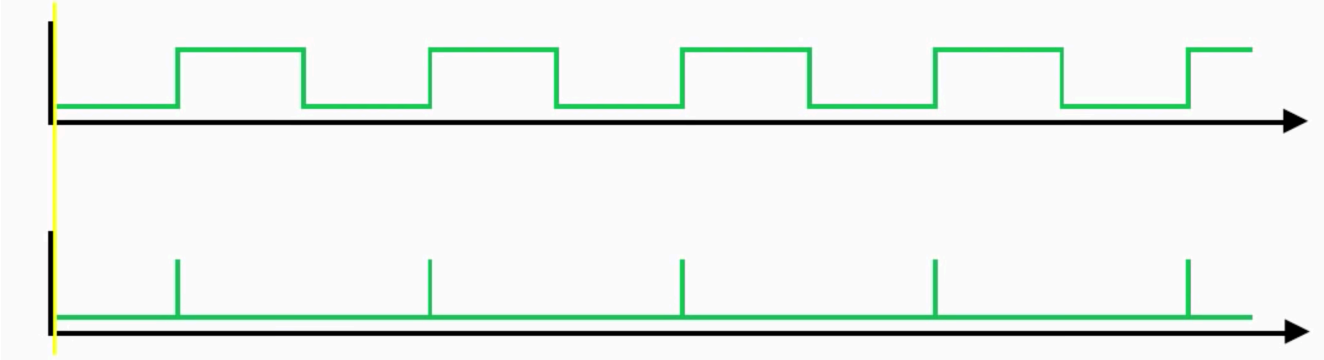
↓



↓

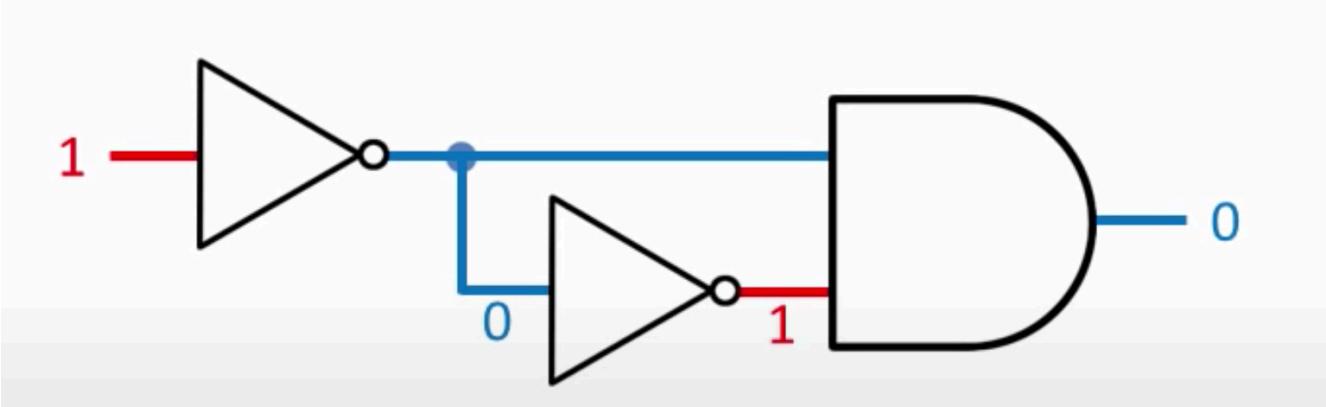


The output of inverter will not change immediately, so the output of this circuit will still be 1, for a very very short period.

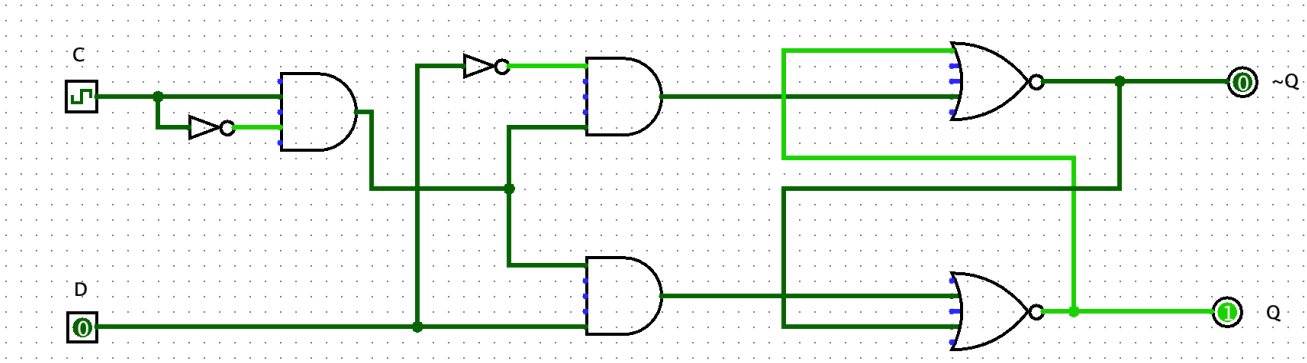


we call this a positive edge detection device

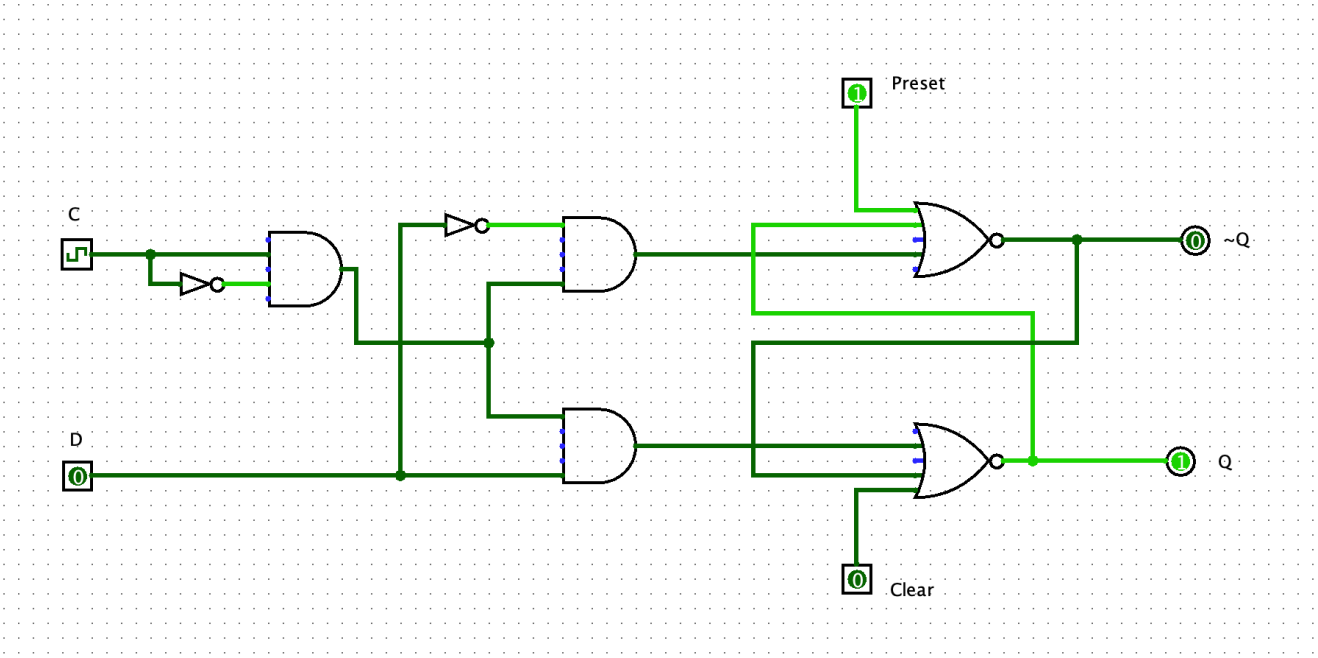
Also, we can build a negative edge detection



- Then we connect the output of this device to the input of D latch to build a clocked D latch



- We can also make an enhancement which allow us to set and clear Q regardless of anything.

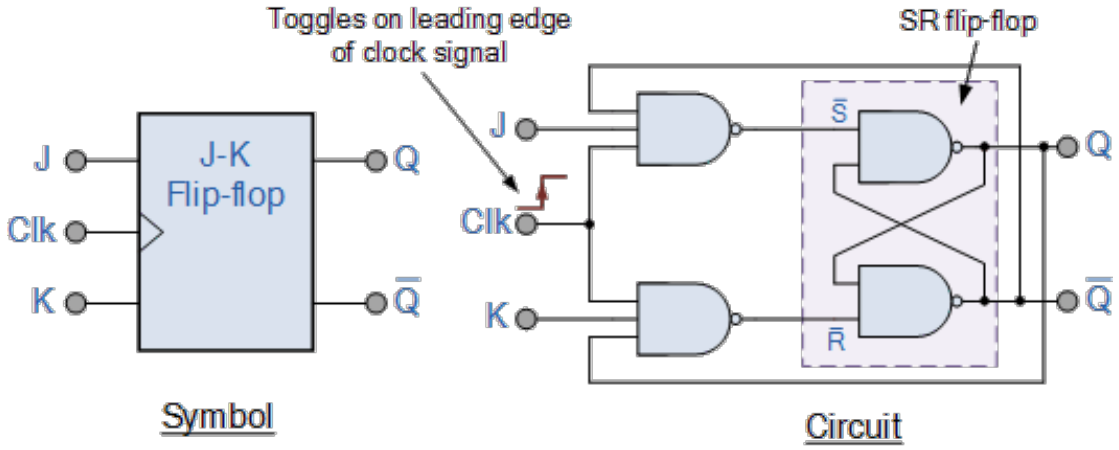


if Preset is 1, Q is set to 1 immediately, no matter what status the whole circuit is in, so does Clear. This is often used to initialize the latch.

Flip-Flop

JK Flip-Flop

J and K have no significance except that they are adjacent letters in the alphabet. It was named so to honor Jack Kilby



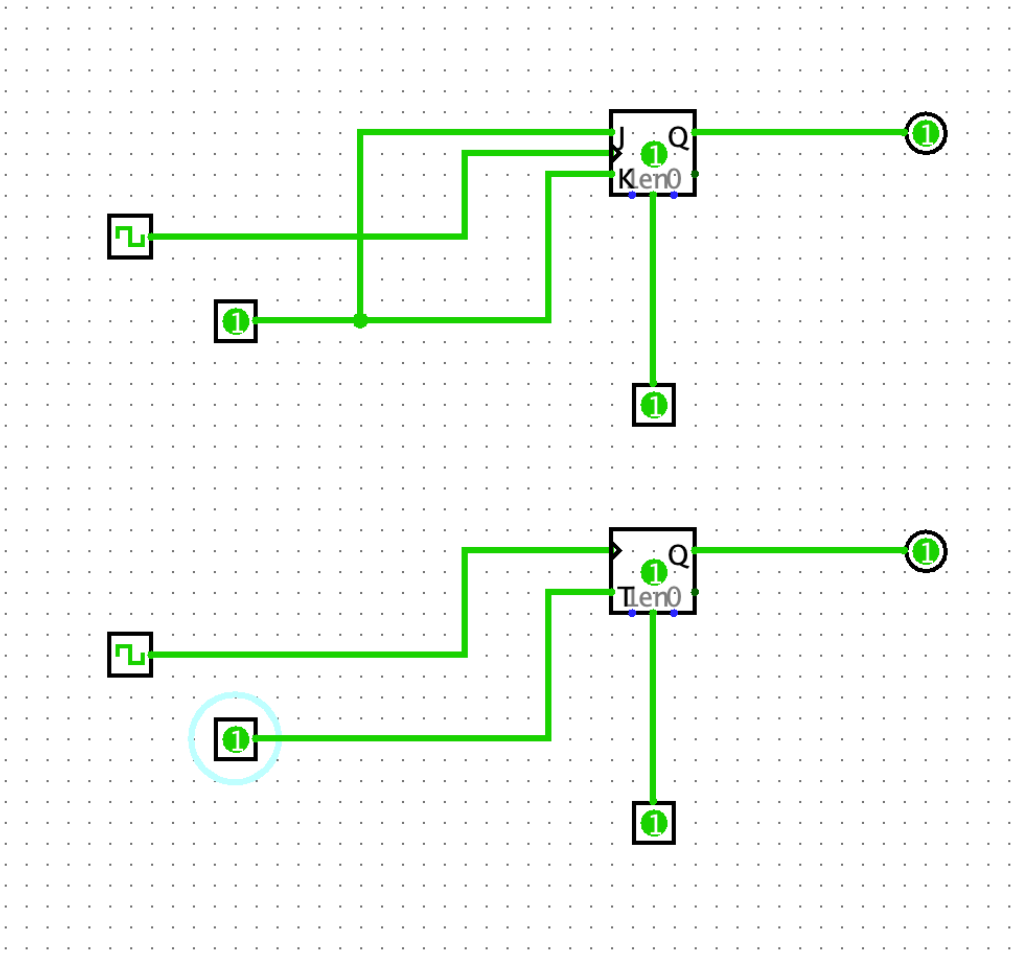
- It is just a Gated SR latch with two feedback line, when (1,1) condition happens, make a toggle.

	Clock	Input		Output		Description
	Clk	J	K	Q	$\overline{Q}$	
same as for the SR Latch	X	0	0	1	0	Memory no change
	X	0	0	0	1	
	⁻↓_	0	1	1	0	Reset Q » 0
	X	0	1	0	1	
	⁻↓_	1	0	0	1	Set Q » 1
	X	1	0	1	0	
toggle action	⁻↓_	1	1	0	1	Toggle
	⁻↓_	1	1	1	0	

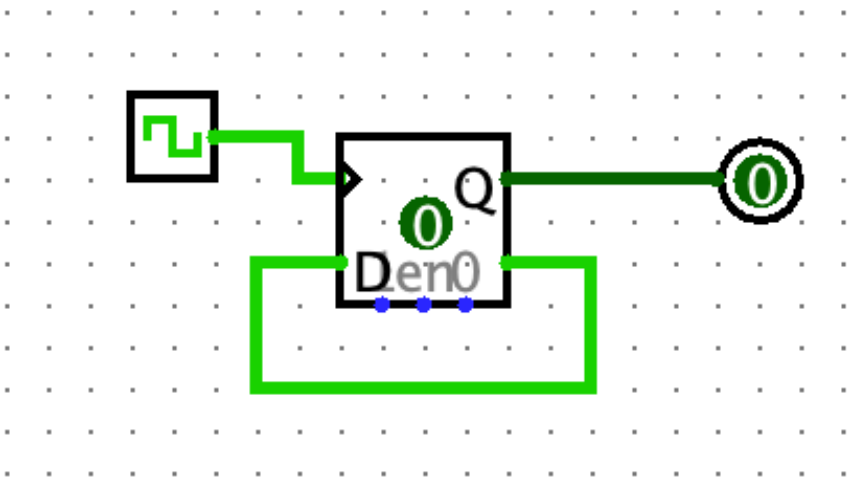
- Characteristic equation
  - $Q^{n+1} = J\bar{Q}^n + \bar{K}Q^n$

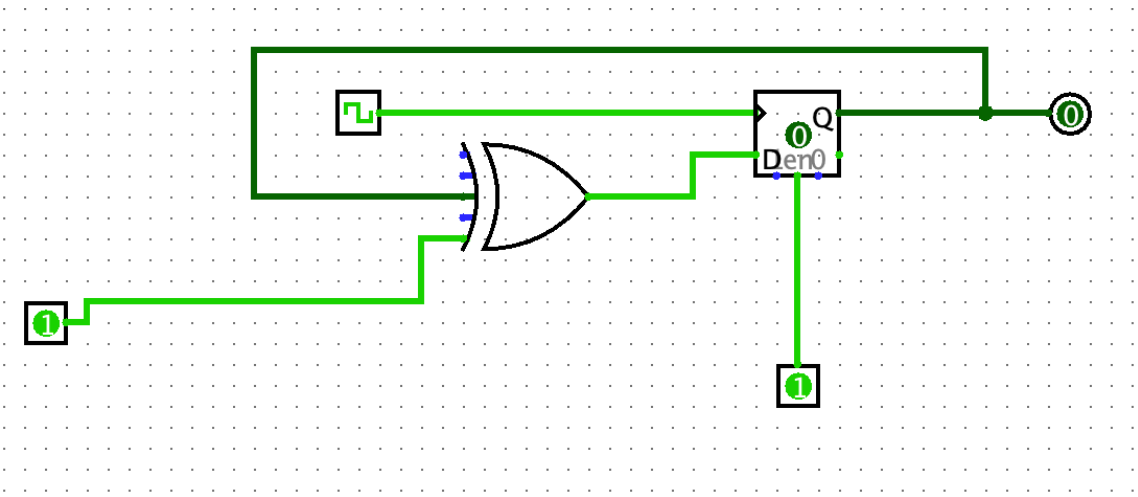
T Flip-Flop

Toggle Flip Flop



- just connect the J input and K input of JKFF, you will get a TFF
- when the positive edge comes, if the T input is 1, the FF toggles, otherwise it stay still.
- Or we can build it from a DFF





For the reason that:

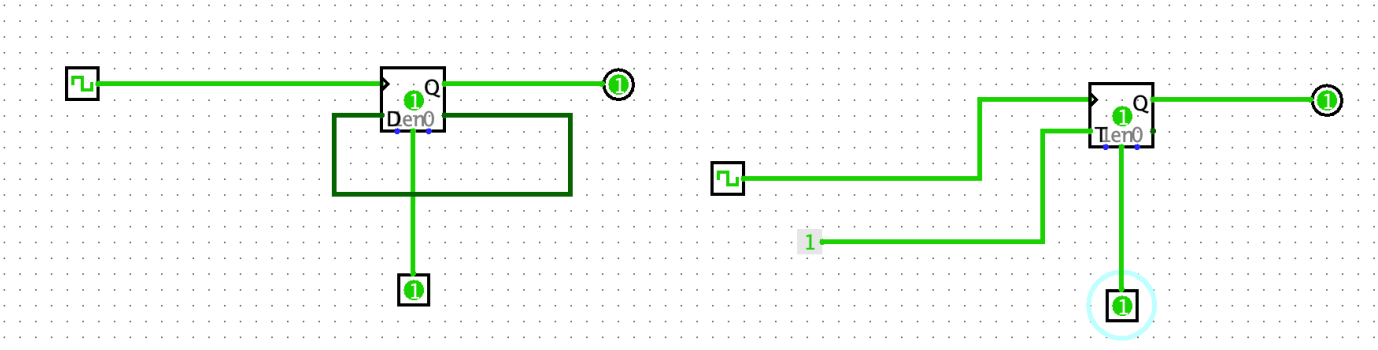
$$D = T\bar{Q}^n + \bar{T}Q^n = T \oplus Q^n$$

> Any expression following the AB' + A'B form (two AND gates and an OR gate) may be replaced by a single Exclusive-OR gate.

- characteristic equation
  - $Q^{n+1} = Q^n \oplus T$

T' FF

make the input of T FF always 1



- when the clock pulse comes, it toggles.