Logic Circuits

- A *gate* is a circuit element that operates on a binary signal.

- Logic operations typically have three methods of description:

1. Equation symbol
2. Truth table
3. Circuit symbol

- The binary levels in logic gates are called 1 and 0.

- When levels refer to Boolean expressions they are referred to as true and false.
  Logic levels are *T*=True and *F*=False.
  Binary levels are 1=True and 0=False.

- When levels refer to electronic voltage levels they are called high and low.
  Logic levels are *H*=High and *L*=Low.
  Binary levels are 1=High and 0=Low.

Boolean Logic

- Boolean logic is based on the Boolean algebra with two operations: AND and OR.

$$A \bullet \bar{A} = 0$$

$$A + \bar{A} = 1$$

- A bar over a variable or expression represents the inverse value.

- Boolean algebra is commutative and distributive.

$$A \bullet B = B \bullet A$$

$$A + B = B + A$$

$$A \bullet (B + C) = (A \bullet B) + (A \bullet C)$$

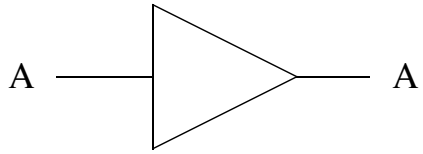
- DeMorgan's theorem links negation and the operations.

$$\overline{A \bullet B} = \bar{A} + \bar{B}$$

$$\overline{A + B} = \bar{A} \bullet \bar{B}$$

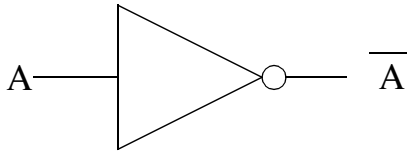
Unary Operations - 1 Input

- There are two unary operations.
- The identity operation leaves the value unchanged.



A_{in}	A_{out}
0	0
1	1

- The inverse operation reverses the value and is called NOT.

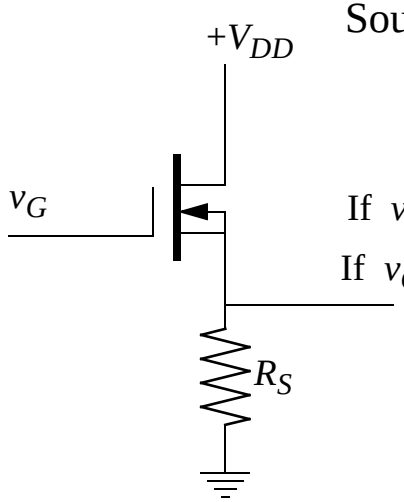


A_{in}	$\overline{A}_{out}$
0	1
1	0

Transistor Gates

- A single FET can form an identity, ...

Source Buffer Follower

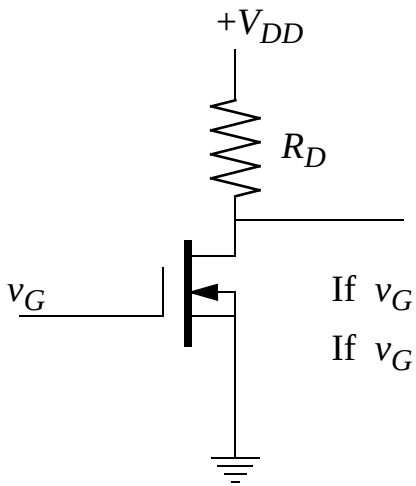


If $v_G = 0$, $v_S = v_G$

If $v_G = V_{DD}$, $v_S = v_G$

- ... or an inverse.

Common Source Inverter

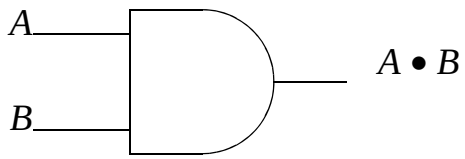


If $v_G = 0$, $v_D = V_{DD} - v_G = V_{DD}$

If $v_G = V_{DD}$, $v_D = V_{DD} - v_G = 0$

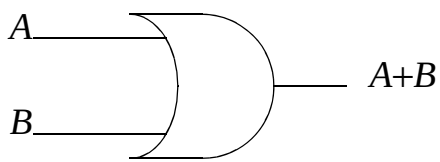
Binary Operations - 2 Inputs

- There are two basic binary operations.
- The AND operation acts like multiplication.



A	B	$A \bullet B$
0	0	0
0	1	0
1	0	0
1	1	1

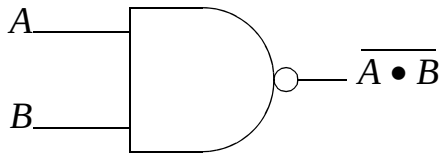
- The OR operation acts like addition.



A	B	$A+B$
0	0	0
0	1	1
1	0	1
1	1	1

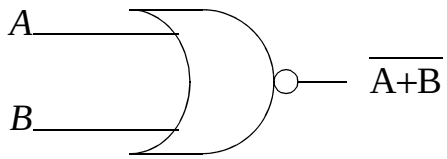
Compound Operations

- The combination of NOT and AND is NAND.



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

- The combination of NOT and OR is NOR.



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

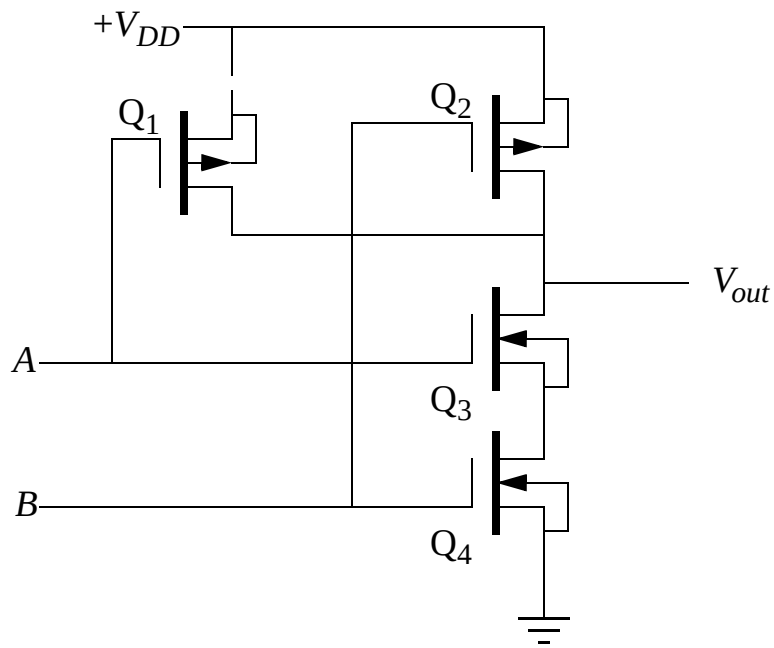
- Either NAND or NOR gates can be used to create other logic gates.

$$\overline{(\overline{A \bullet B}) \bullet (\overline{A \bullet B})} = A \bullet B$$

$$\overline{(\overline{A + A}) + (\overline{B + B})} = A \bullet B$$

Electronic NAND

- The CMOS implementation of a NAND can be found in the 4011B.



The CMOS MOSFETs are connected as switches.

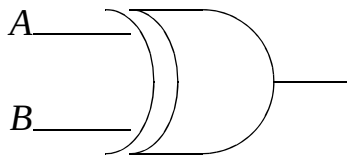
HIGH at A and B turn on Q_3 and Q_4 while turning off Q_1 and Q_2.

LOW at A and B turn on Q_1 and Q_2 while turning off Q_3 and Q_4.

- If both Q_3 and Q_4 are on then V_{out} is at ground, otherwise either Q_1 or Q_2 will be on pulling V_{out} up to V_{DD}.

Exclusive OR

- The exclusive or is a common comound gate.
- The XOR selects inputs that differ.



$$A \oplus B = (A + B) \bullet \overline{(A \bullet B)}$$

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

- There are a number of equivalent logical constructions for XOR.

$$A \oplus B = (A \bullet \bar{B}) + (\bar{A} \bullet B)$$

$$A \oplus B = \overline{\overline{(A \bullet \bar{B})} \bullet \overline{(\bar{A} \bullet B)}}$$

Logic Types

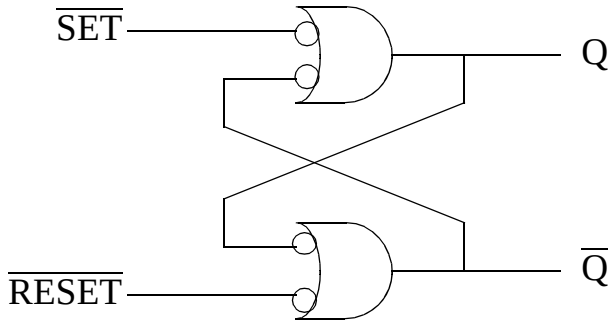
- There are two common types of logic gates: TTL - Transistor-Transistor Logic and CMOS - Complementary Metal-Oxide-Semiconductor.

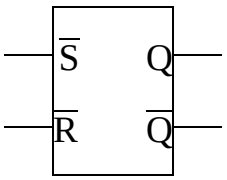
Properties of TTL vs. CMOS

Property	**TTL**	**CMOS**
Power Supply	4.75 to 5.25 V	2 to 15 V
Input Current	LOW sources 0.5 mA	none (static charge may damage)
Input Threshold	1.3 V	0.3 V_{DD} to 0.5 V_{DD} (except HCT)
Output	LOW - 0.2 V HIGH - 3.7 to 4.4 V	LOW - 0 V HIGH - V_{DD}
Power Consumption	considerable	minimal except at high frequency
Other	threshold sensitive to noise	high output imped-nace (~100 Ω)

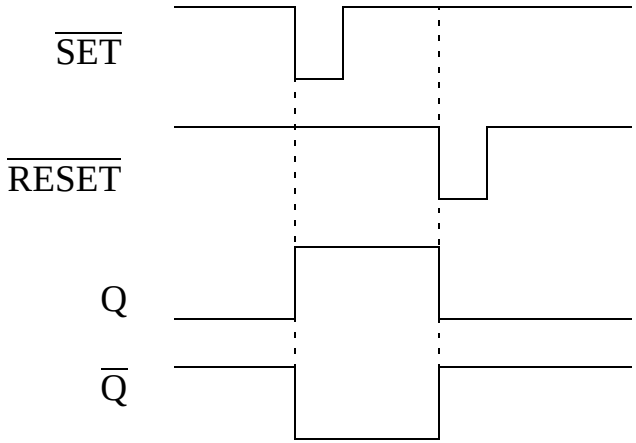
Flip Flops

- A flip-flop can be forced into one state or the other and hold it when some inputs change.
- The RS (Set-Reset) flip-flop is forced into one state or another with an input to SET or RESET.
- Negative inputs are mean that the effect is active when the signal is low.



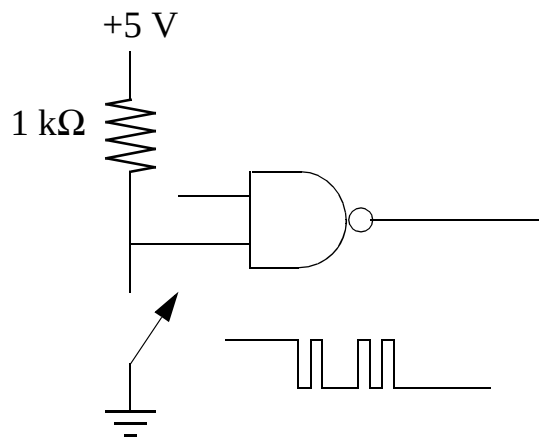


- The SET and RESET cannot be used simultaneously or there is an ambiguous result.

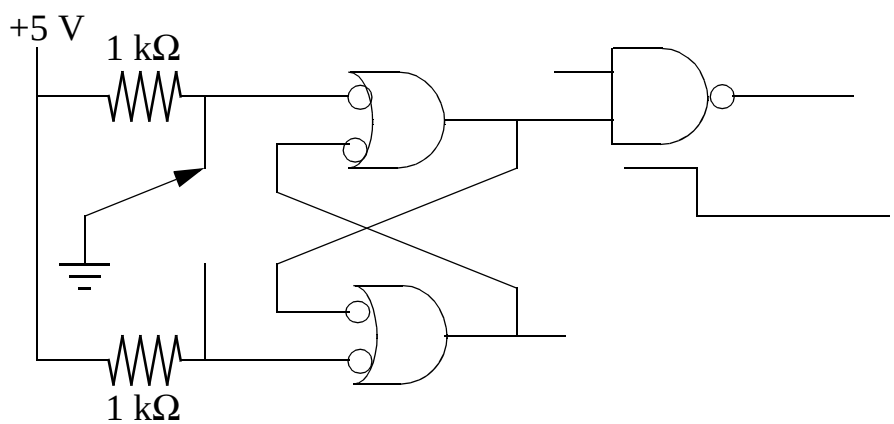


Switch Debouncer

- A mechanical switch will bounce on its contact and provide multiple pulses.

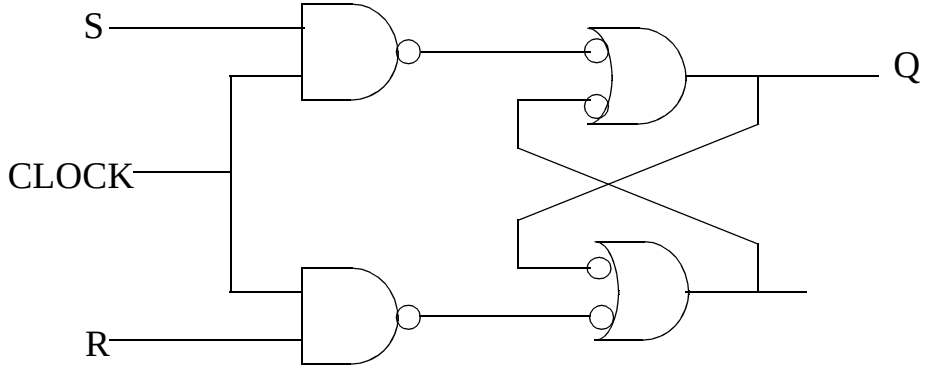


- A flip-flop is sensitive to the first bounce, but none after until it is reset.



Triggered Flip-Flops

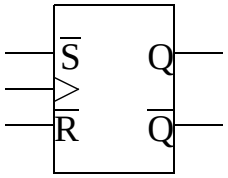
- A clock can be used to control which input is used.



The extra NAND gates allow the CLOCK to control whether S and R make it to the flip-flop.

If CLOCK=0 the inputs to the flip-flop are disabled and Q stays constant.

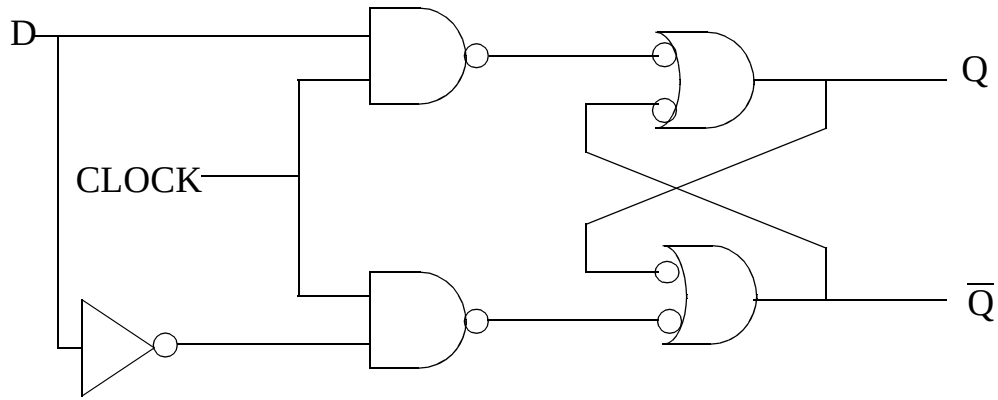
If CLOCK=1 the inputs are active and Q samples S and R.



- A disadvantage to the circuit is that if S or R change during the clock pulse, only the final state of the RS flip-flop is preserved when CLOCK=0 again.

D-Type Flip-Flop

- The D-type flip-flop is identical to a clocked RS flip-flop, except one input is inverted to form the other input.



- The truth table shows states where the circuit retains the previous value.

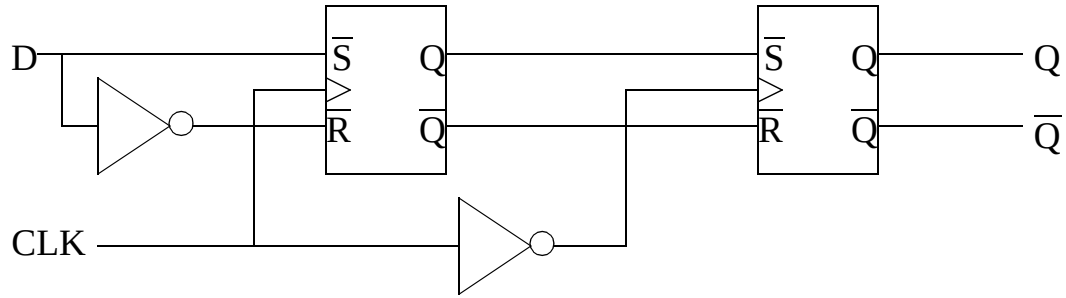
CLK	D	Q	$\overline{Q}$
0	0	Q	$\overline{Q}$
0	1	Q	$\overline{Q}$
1	0	0	1
1	1	1	0

- With only one input the indeterminate state (R=S=1) is avoided.
- D can make many transistions while CLK = 1, only the last level is stored when CLK = 0.

Edge-Triggered Flip-Flops

- Preceding flip-flops were level-sensistive and and active when the clock level was correct.
- The master-slave flip-flop is an edge-sensitive trigger.

- This circuit consists of two level-sensitive D-type flip-flops.



During CLK=1, the first flip-flop is enabled, but the second is disabled (memory only).

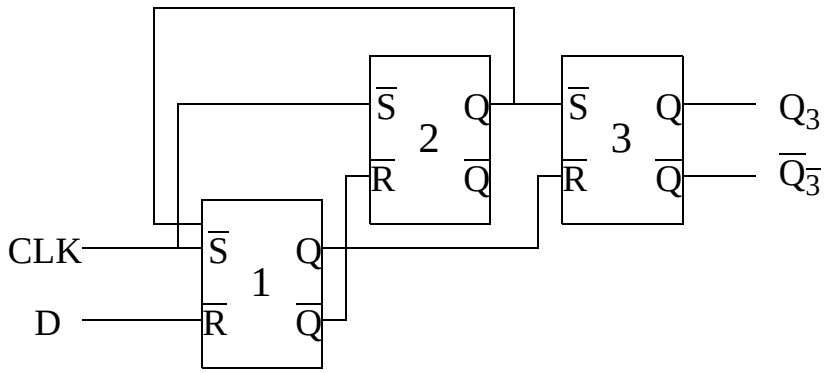
During CLK=0, the first flip-flop is disabled, but the second is enabled, so it samples what ever is held at that time on flip-flop 1.

The output Q can only change exactly as CLK goes from 1 to 0.

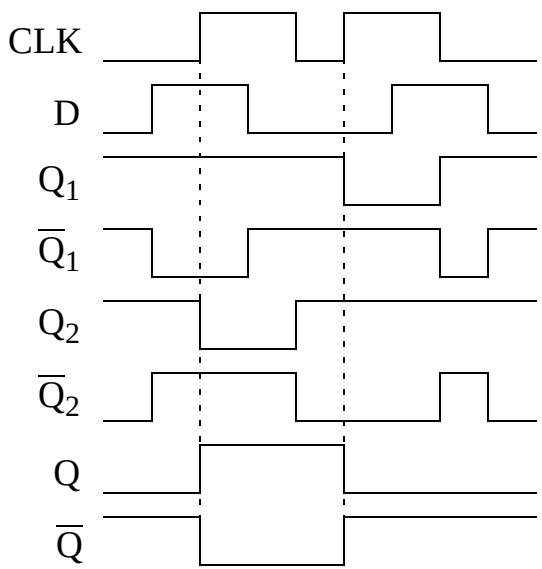
<u>This is a negative edge trigger.</u>

Flip-Flop Timing

- An edge-triggered flip-flop can be built from three RS flip flops.

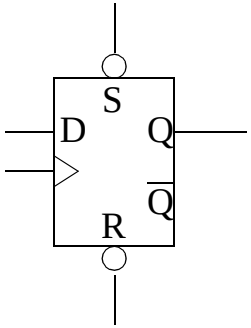


- The timing diagram shows the positive edge triggers in the statges.



Set and Clear Inputs

- Most edge-triggered flip-flops come with both set and clear options that work like S and R from an RS flip-flop.



If SET=0, then Q is forced to 1, and $\overline{Q}$ to 0.

If CLEAR=0 (RESET), then $\overline{Q}$ is forced to 1, and Q to 0.

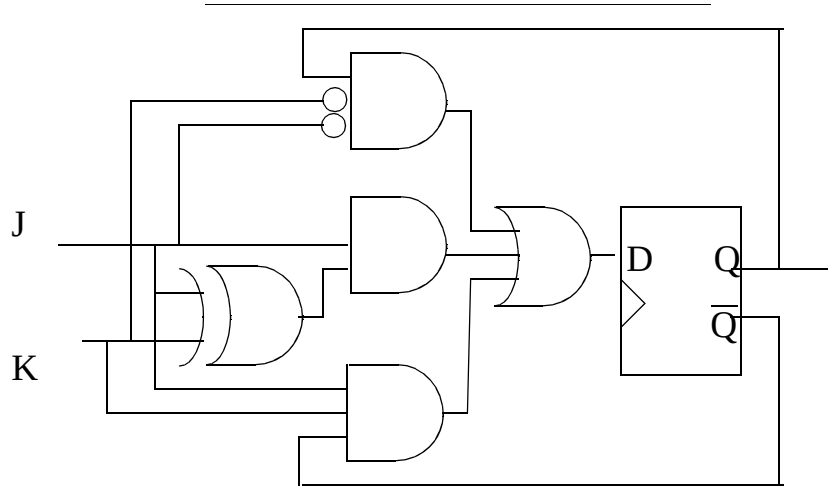
SET and CLEAR take effect regardless of the state of CLK.

- D-type flip-flops also come with SET and CLEAR.

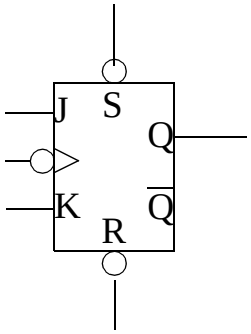
JK Flip-Flops

- The JK flip-flop uses 2-input combinatoric logic with feedback to set the D input of flip-flop.

$$D = (J \oplus K) \bullet J + J \bullet K \bullet \overline{Q} + \overline{J} \bullet \overline{K} \bullet Q$$

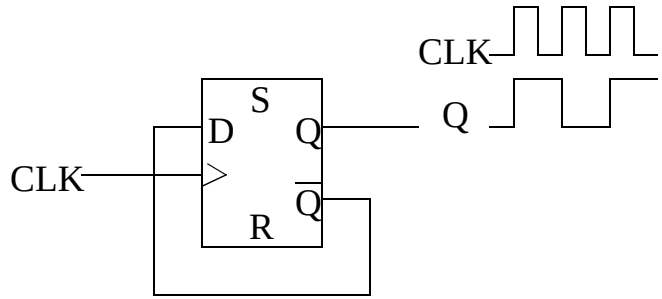


- When both J and K are low then the output remains as it was.
- When either J or K alone are high then the output matches the state of J.
- When both J and K are high then the output switches state.
- The JK flip-flop also comes with SET and CLEAR in a single chip.



Dividers

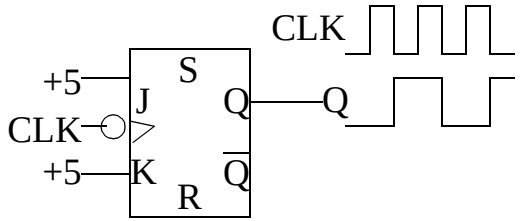
- A divider uses the negative output to toggle the output at slower rate than the input.
- A D-type flip-flop can easily become a divide by two circuit.



D is always set with $\overline{Q}$, so at each rising edge Q switches.

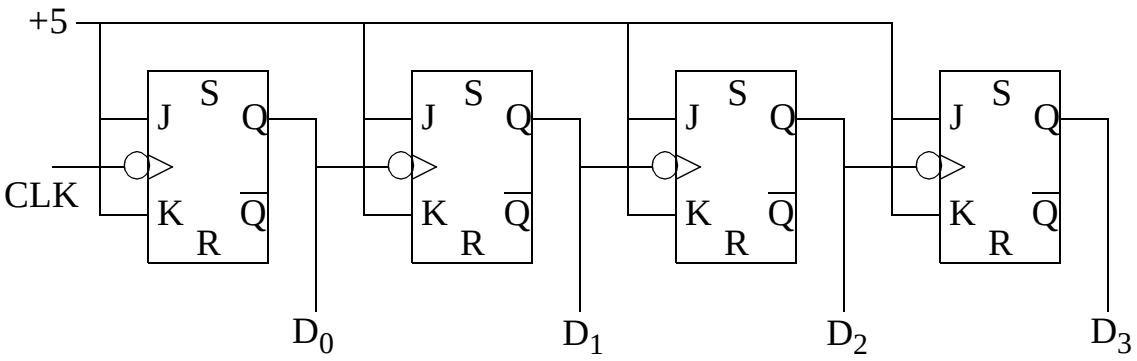
There is no confusion at the rising edge since there is a 10ns propagation delay through the flip-flop (74HC74), and the output needs to be stable for only 3 ns.

- The JK flip-flop can use the internal toggle setting to become a divide by two circuit.



Multistage Divider

- Multiple divide by two circuits can be combined to form a divide by 2^n circuit.



Each JK flip-flop is set to divide the clock by 2.

D_0 is dividing the input clock by 2.

D_1 is dividing the input clock by 4.

D_2 is dividing the input clock by 8.

D_3 is dividing the input clock by 16.

This can be extended to any arbitrary length.

Divide by 16 Truth Table

CLK	D_0	D_1	D_2	D_3	Count
0	0	0	0	0	0
1	0	0	0	0	0
0	1	0	0	0	1
1	1	0	0	0	1
0	0	1	0	0	2
1	0	1	0	0	2
0	1	1	0	0	3
...	...	...	...	...	...
1	0	1	1	1	14
0	1	1	1	1	15
1	1	1	1	1	15
0	0	0	0	0	0