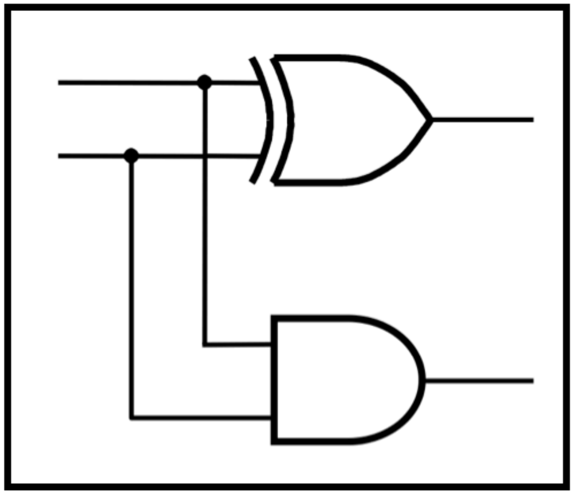


CprE 281: Digital Logic

Instructor: Alexander Stoytchev

http://www.ece.iastate.edu/~alexs/classes/

Latches

CprE 281: Digital Logic
Iowa State University, Ames, IA

Administrative Stuff

- HW 7 is out

- It is due next Monday (Oct 19) @ 4pm

Administrative Stuff

- **HW 8 is out**

- **It is due on Monday (Oct 26)**

Chapter 5

Terminology

- **Basic Latch –** is a feedback connection of two NOR gates or two NAND gates, which can store one bit of information. It can be set using the S input and reset to 0 using the R input.

- **Gated Latch –** is a basic latch that includes input gating and a control input signal. The latch retains its existing state when the control input is equal to 0. Its state may be changed when the control signal is equal to 1.

[Section 5.7 in the textbook]

Terminology

- **Two types of gated latches (the control input is the clock):**

- **Gated SR Latch –** uses the S and R inputs to set the latch to 1 or reset it to 0.

- **Gated D Latch –** uses the D input to force the latch into a state that has the same logic value as the D input.

[Section 5.7 in the textbook]

Terminology

- **Flip-Flop –** is a storage element that can have its output state changed only on the edge of the controlling clock signal.

- **Positive-edge triggered –** if the state changes when the clock signal goes from 0 to 1.

- **Negative-edge triggered –** if the state changes when the clock signal goes from 1 to 0.

[Section 5.7 in the textbook]

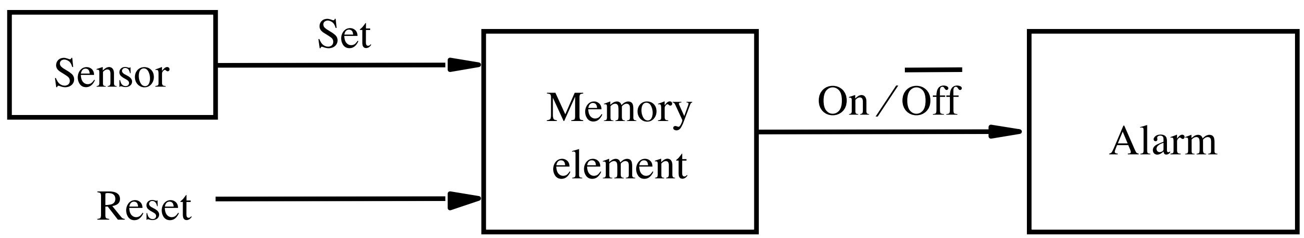
Terminology

The word ***latch*** is mainly used for storage elements, while clocked devices are described as ***flip-flops***.

A **latch** is level-sensitive, whereas a **flip-flop** is edge-sensitive. That is, when a latch is enabled it becomes transparent, while a flip flop's output only changes on a single type (positive going or negative going) of clock edge.

[http://en.wikipedia.org/wiki/Flip-flop_(electronics)]

Control of an alarm system



[Figure 5.1 from the textbook]

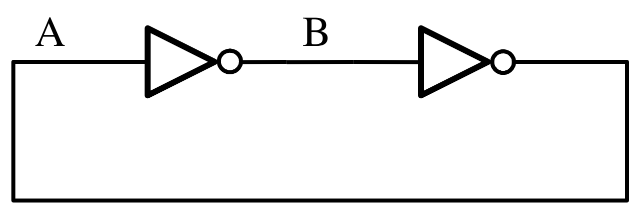
Motivation

So far, our circuits have just been converting inputs to outputs.

To do more advanced things (i.e. to make computers) we need components that can store data.

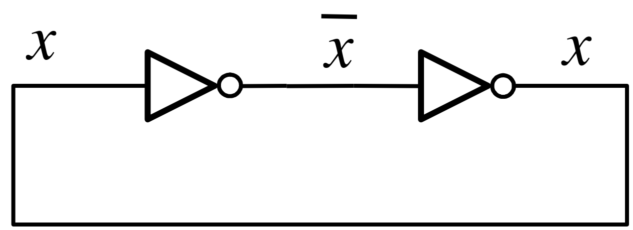
Can we make a component that “remembers” from the components that we know?

A simple memory element



[Figure 5.2 from the textbook]

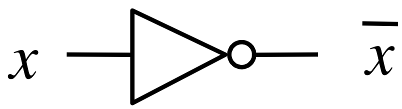
A simple memory element with NOT Gates

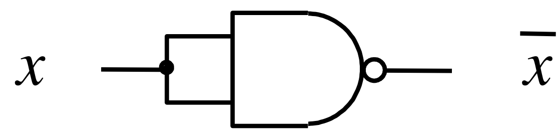


A Strange Loop

[http://animalsiadmire.blogspot.com/2011/07/stupid-snake-eating-itself.html]

Building a NOT Gate with NAND





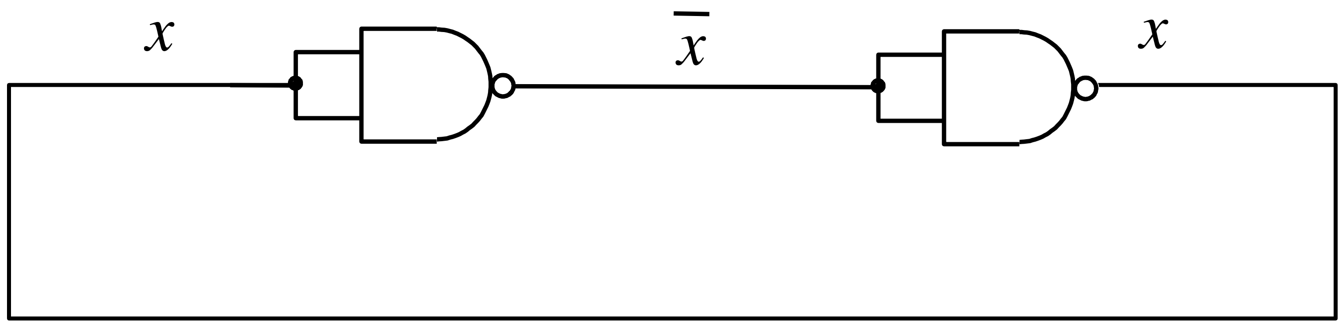
x	$\overline{x}$
0	1
1	0

x	x	f
0	0	1
[redacted]		
1	1	0

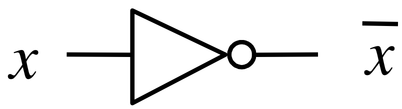
impossible combinations

Thus, the two truth tables are equal!

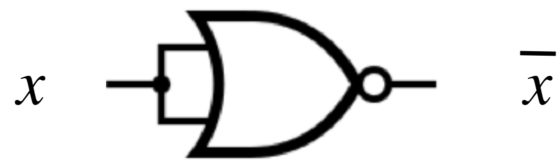
A simple memory element with NAND Gates



Building a NOT Gate with NOR



x	$\overline{x}$
0	1
1	0

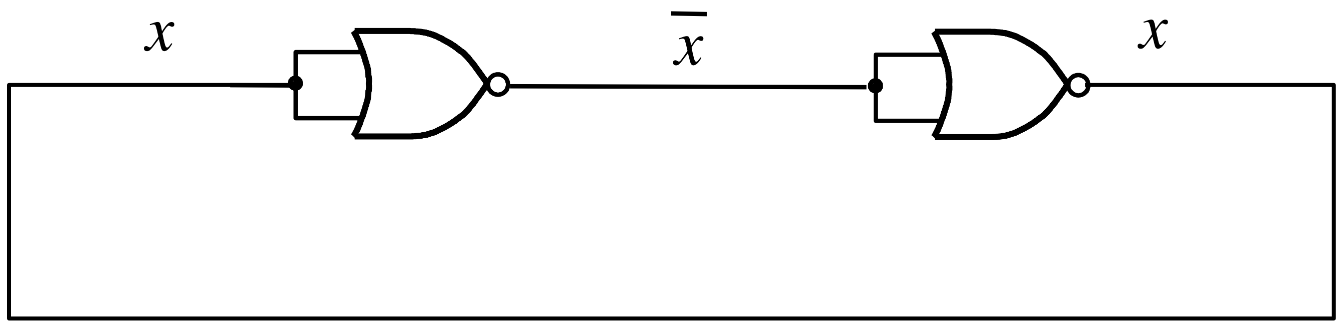


x	x	f
0	0	1
[redacted]		
1	1	0

impossible combinations

Thus, the two truth tables are equal!

A simple memory element with NOR Gates

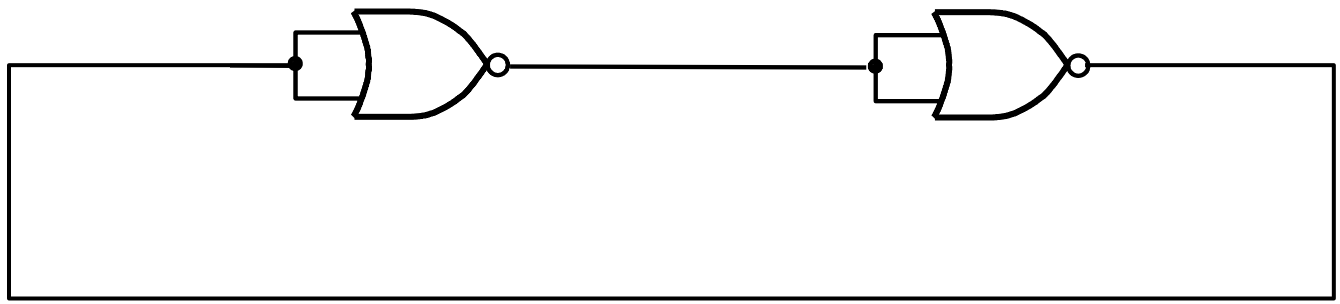


Basic Latch

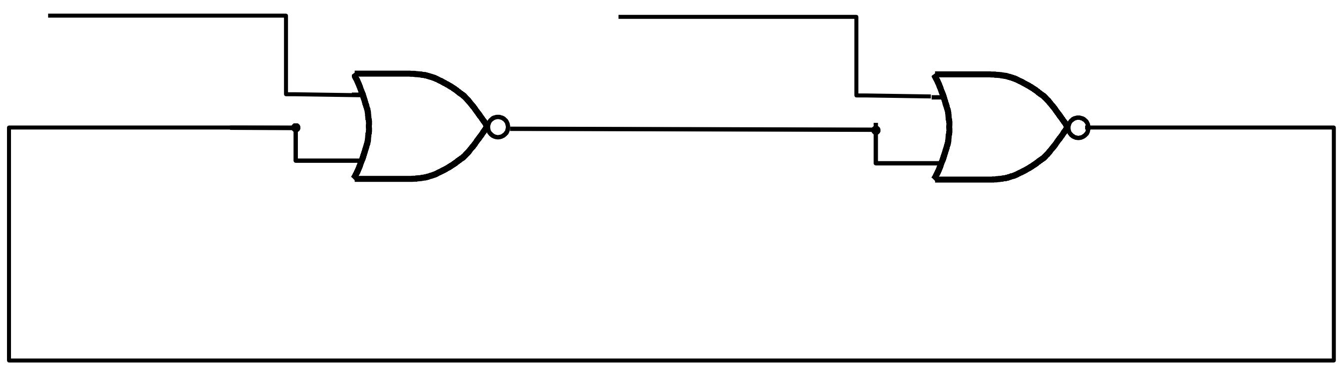
What is a latch?

http://luxuriahardware.com/images/detailed/13/P111408_m.jpg

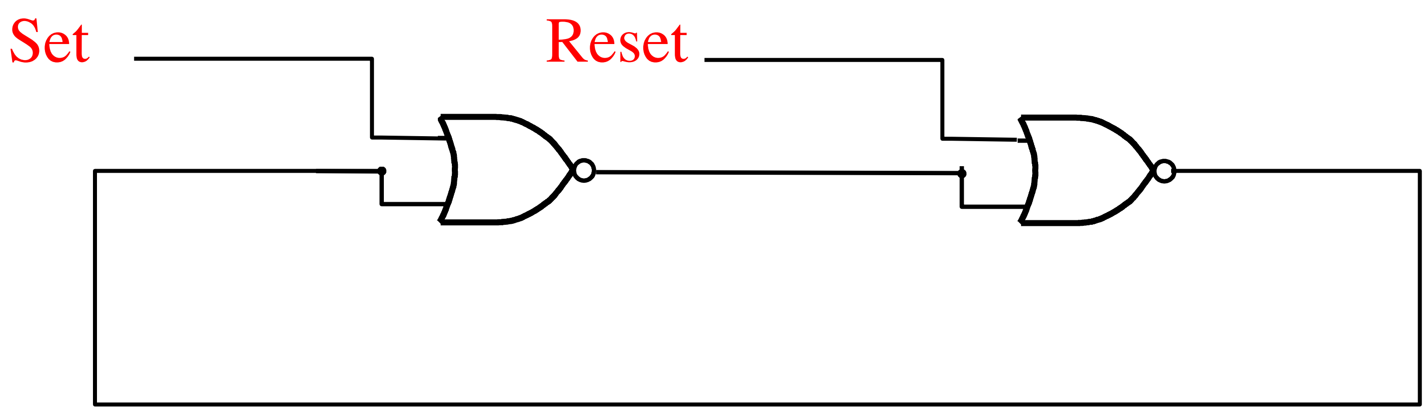
A simple memory element with NOR Gates



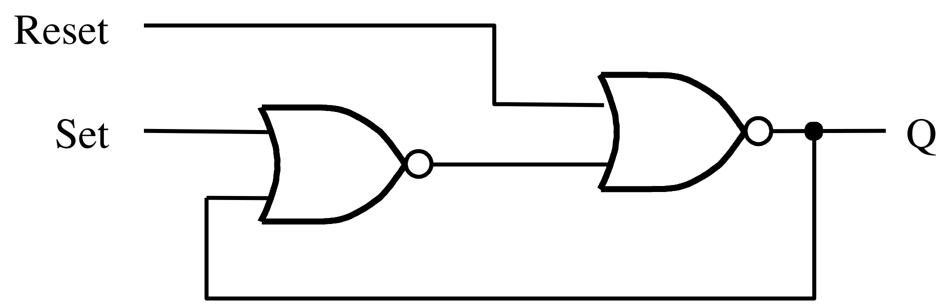
A simple memory element with NOR Gates



A simple memory element with NOR Gates

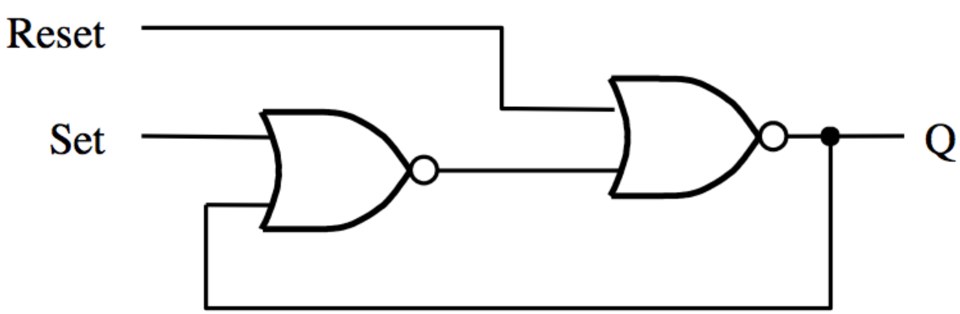


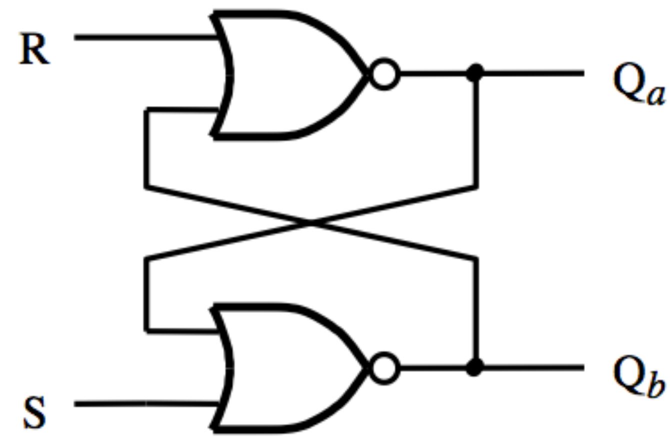
A memory element with NOR gates



[Figure 5.3 from the textbook]

Two Different Ways to Draw the Same Circuit

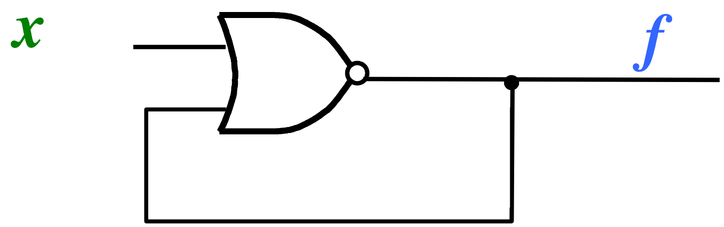




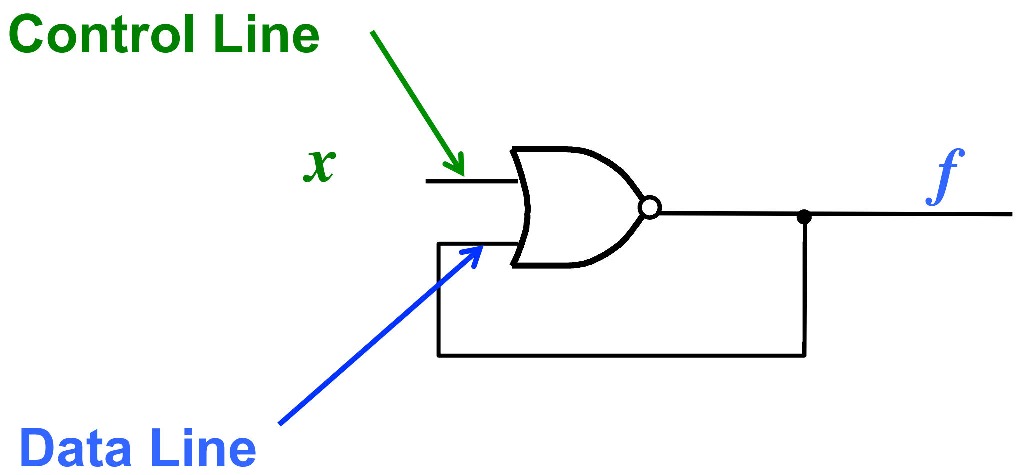
[Figure 5.3 & 5.4 from the textbook]

Before We Analyze the Basic Latch Let's Look at a Two Simpler Examples with Feedback

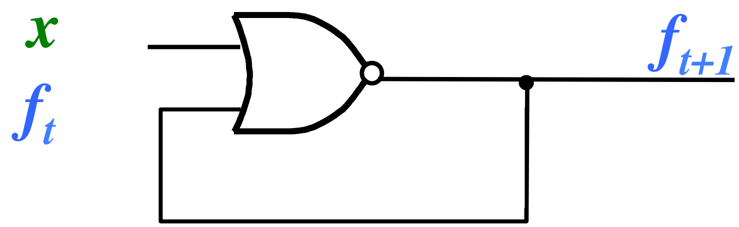
Let's Try to Analyze This Circuit



Let’s Try to Analyze This Circuit

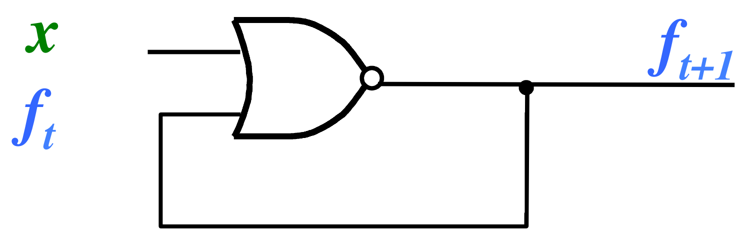


Let's Try to Analyze This Circuit



x	f_t	f_{t+1}
0	0	
0	1	
1	0	
1	1	

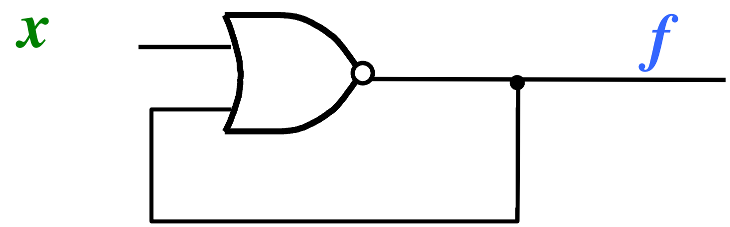
Let’s Try to Analyze This Circuit



x	f_t	f_{t+1}
0	0	1
0	1	0
1	0	0
1	1	0

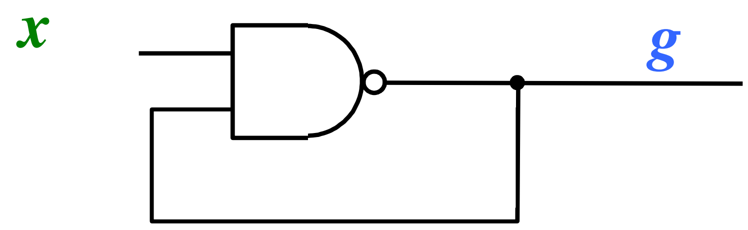
If x = 0, then *f* is negated.
If x = 1, then *f* is driven to 0.

Key Observation

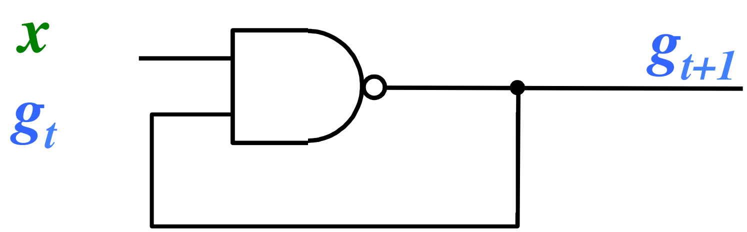


If a NOR's control line is 0, then that NOR just *negates* its data line. If the control line is 1, then the NOR's output is *driven* to 0, ignoring its data line.

Let's Try to Analyze This Circuit

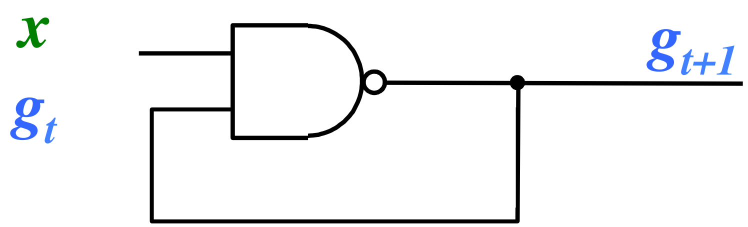


Let's Try to Analyze This Circuit



x	g_t	g_{t+1}
0	0	
0	1	
1	0	
1	1	

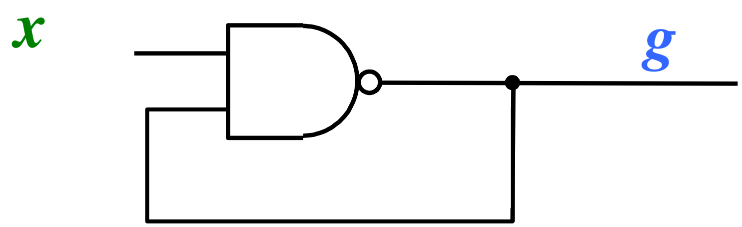
Let's Try to Analyze This Circuit



x	g_t	g_{t+1}
0	0	1
0	1	1
1	0	1
1	1	0

If $x = 0$, then g is driven to one.
If $x = 1$, then g is negated.

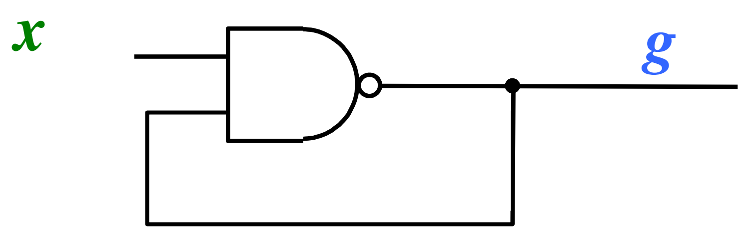
Key Observation



If a NAND's control line is 1, then that NAND just *negates* its data line. If the control line is 0, then the NAND's output is *driven* to 1, ignoring its data line.

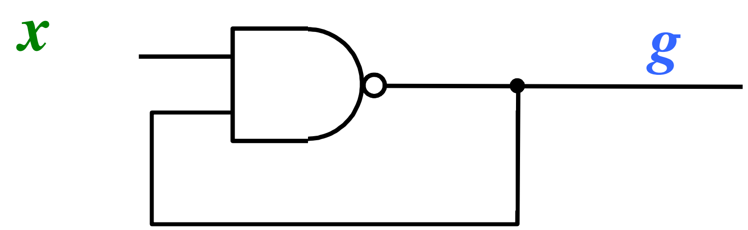
Output Oscillations

What would happen to *g* if we keep x=1 for a long time?



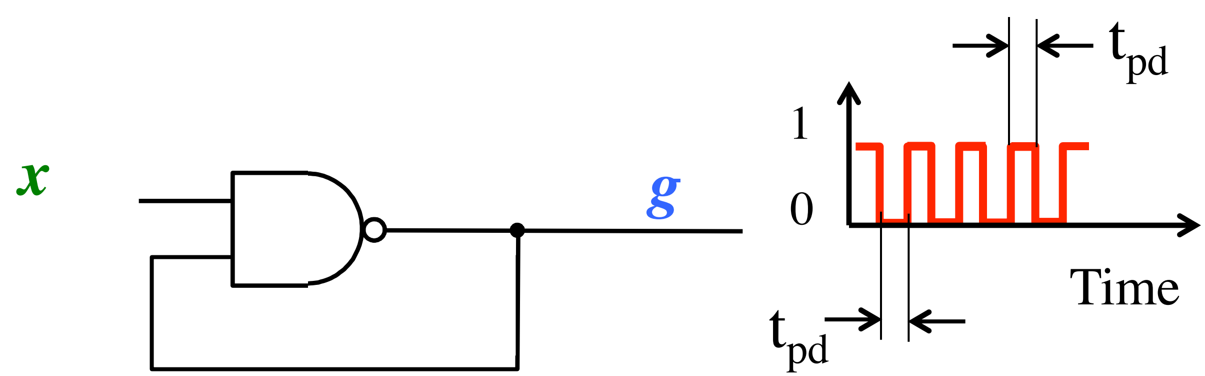
Output Oscillations

What would happen to *g* if we keep x=1 for a long time?



Output Oscillations

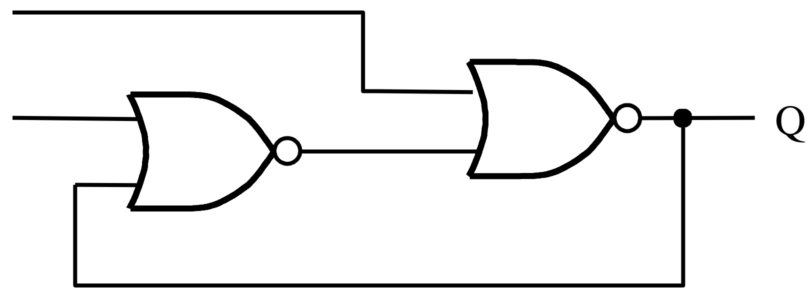
What would happen to *g* if we keep x=1 for a long time?



t_{pd} is the propagation delay through the NAND gate, which is small, but not zero.

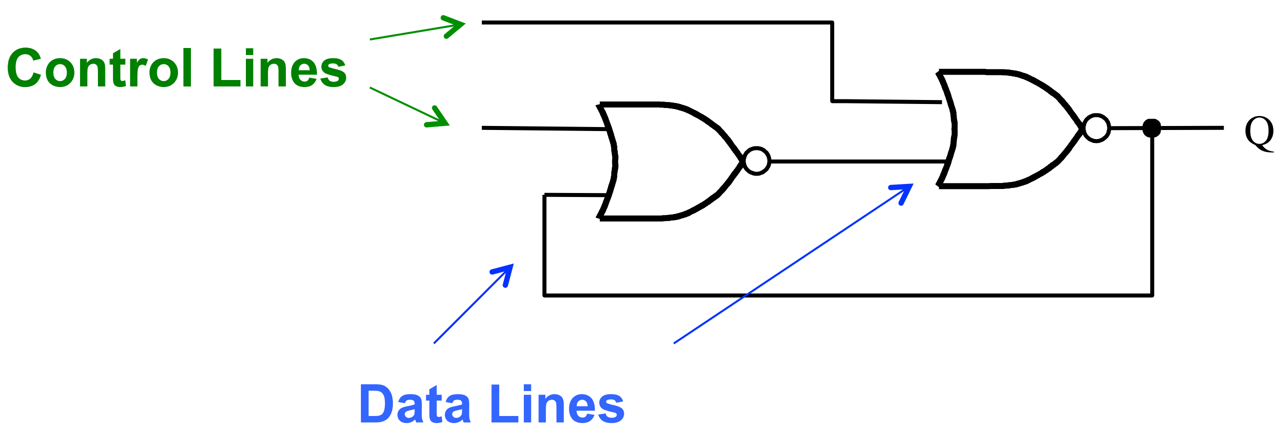
Back to the Basic Latch

The Basic Latch



[Figure 5.3 from the textbook]

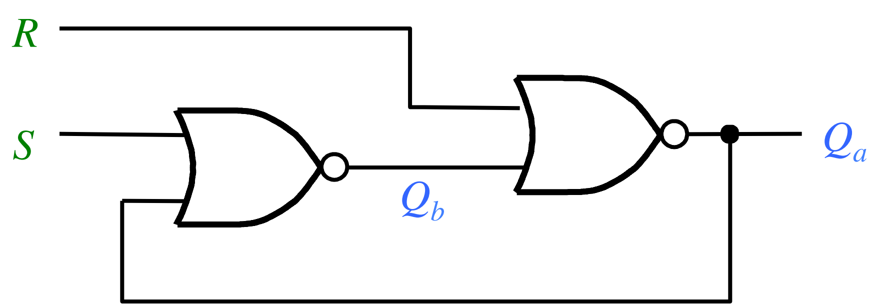
The Basic Latch



Two of the previous NOR memory elements put togetter so that the data is flipped twice.

[Figure 5.3 from the textbook]

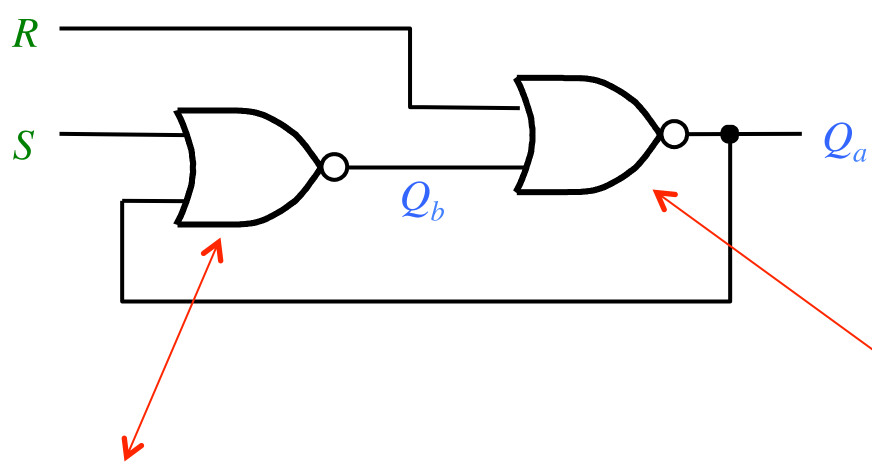
Analyzing The Basic Latch



S	Q_a	Q_b = NOR (S, Q_a)
0	0	
0	1	
1	0	
1	1	

R	Q_b	Q_a = NOR (R, Q_b)
0	0	
0	1	
1	0	
1	1	

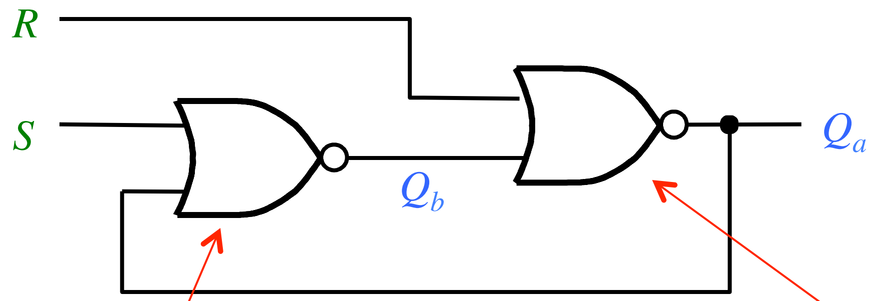
Analyzing The Basic Latch



S	Q_a	Q_b = NOR (S, Q_a)
0	0	1
0	1	0
1	0	0
1	1	0

R	Q_b	Q_a = NOR (R, Q_b)
0	0	1
0	1	0
1	0	0
1	1	0

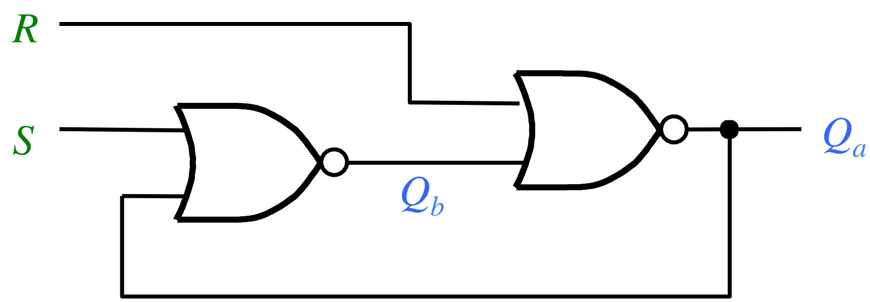
Analyzing The Basic Latch



S	Q_a	$Q_b = \text{NOR}(S, Q_a)$	
0	0	1	$\overline{Q_a}$
0	1	0	
1	0	0	0
1	1	0	

R	Q_b	$Q_a = \text{NOR}(R, Q_b)$	
0	0	1	$\overline{Q_b}$
0	1	0	
1	0	0	0
1	1	0	

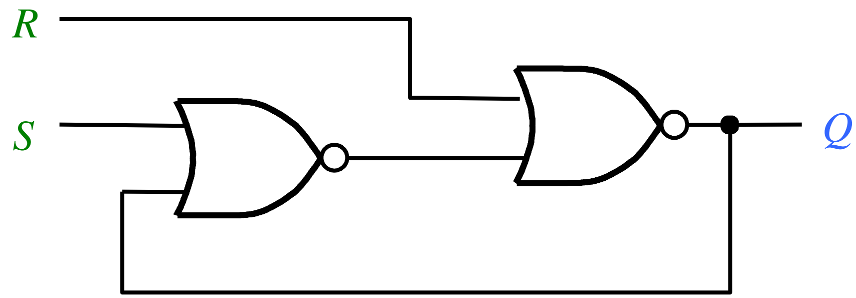
Analyzing The Basic Latch



S	Q_b
0	$\overline{Q}_a$
1	0

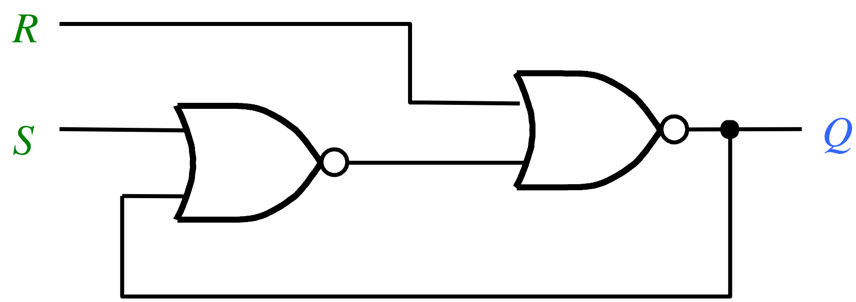
R	Q_a
0	$\overline{Q}_b$
1	0

Behavior of the Basic Latch



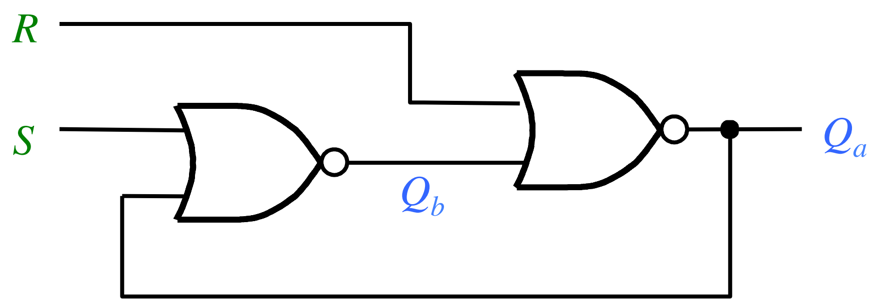
S	R	Q_{t+1}
0	0	
0	1	
1	0	
1	1	

Behavior of the Basic Latch



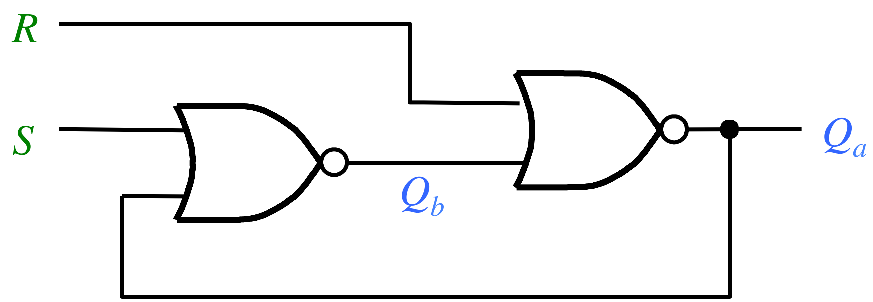
S	R	Q_{t+1}
0	0	Q_t
0	1	0
1	0	1
1	1	0

Behavior of the Basic Latch



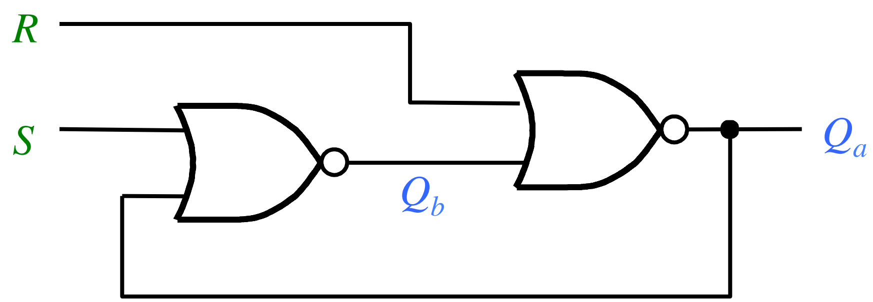
S	R	$Q_a(t+1)$	$Q_b(t+1)$
0	0		
0	1		
1	0		
1	1		

Behavior of the Basic Latch



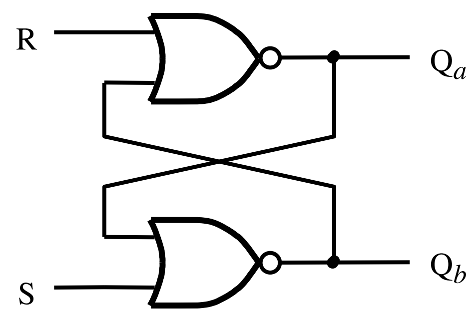
S	R	$Q_a(t+1)$	$Q_b(t+1)$
0	0	$Q_a(t)$	$Q_b(t)$
0	1	0	1
1	0	1	0
1	1	0	0

Behavior of the Basic Latch



S	R	$Q_a(t+1)$	$Q_b(t+1)$	
0	0	$Q_a(t)$	$Q_b(t)$	Latch
0	1	0	1	Reset
1	0	1	0	Set
1	1	0	0	Undesirable

Circuit and Characteristic Table



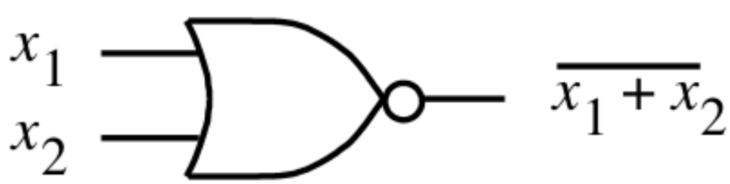
(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Characteristic table

[Figure 5.4a,b from the textbook]

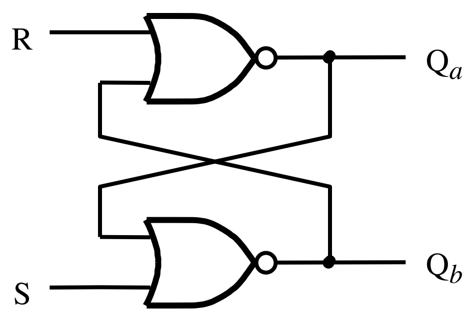
NOR Gate



NOR Gate Truth table

x_1	x_2	f
0	0	1
0	1	0
1	0	0
1	1	0

Circuit and Characteristic Table



(a) Circuit

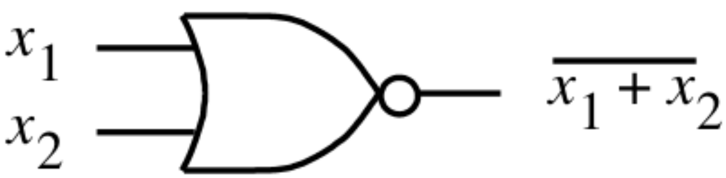
S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Characteristic table

A truth table should take the state into account.
A characteristic table takes only the inputs into account.

[Figure 5.4a,b from the textbook]

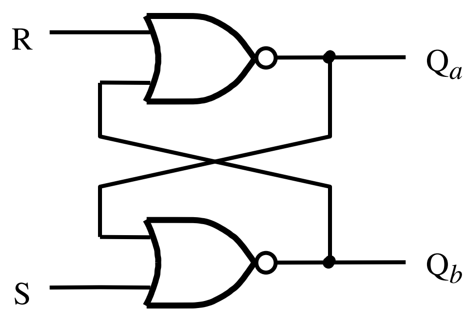
NOR Gate



NOR Gate Truth table

x_1	x_2	f
0	0	1
0	1	0
1	0	0
1	1	0

Circuit and Characteristic Table



(a) Circuit

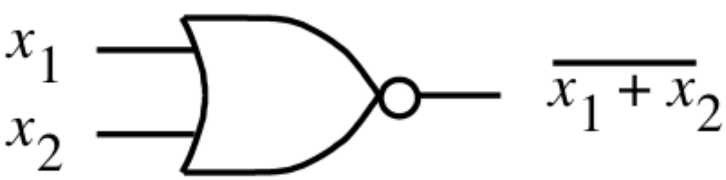
S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Characteristic table

Note that Q_a and Q_b are inverses of each other!

[Figure 5.4a,b from the textbook]

NOR Gate



NOR Gate Truth table

x_1	x_2	f
0	0	1
0	1	0
1	0	0
1	1	0

Oscillations and Undesirable States

- **When both S=1 and R=1 both outputs of the latch are equal to 0, i.e., Q_a=0 and Q_b=0.**

- **Thus, the two outputs are no longer complements of each other.**

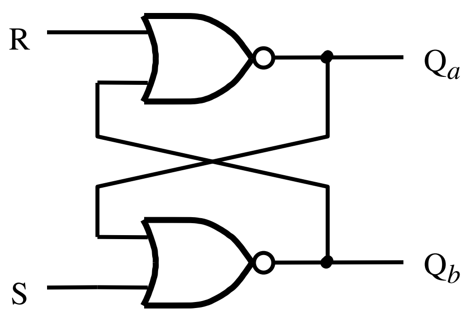
- **This is undesirable as many of the circuits that we will build later with these latches rely on the assumption that the two outputs are always complements of each other.**

- **(This is obviously not the case for the basic latch, but we will patch it later to eliminate this problem).**

Oscillations and Undesirable States

- **An even bigger problem occurs when we transition from S=R=1 to S=R=0.**
- **When S=R=1 we have $Q_a=Q_b=0$. After the transition to S=R=0, however, we get $Q_a=Q_b=1$, which would immediately cause $Q_a=Q_b=0$, and so on.**
- **If the gate delays and the wire lengths are identical, then this oscillation will continue forever.**
- **In practice, the oscillation dies down and the output settles into either $Q_a=1$ and $Q_b=0$ or $Q_a=0$ and $Q_b=1$.**
- **The problem is that we can't predict which one of these two it will settle into.**

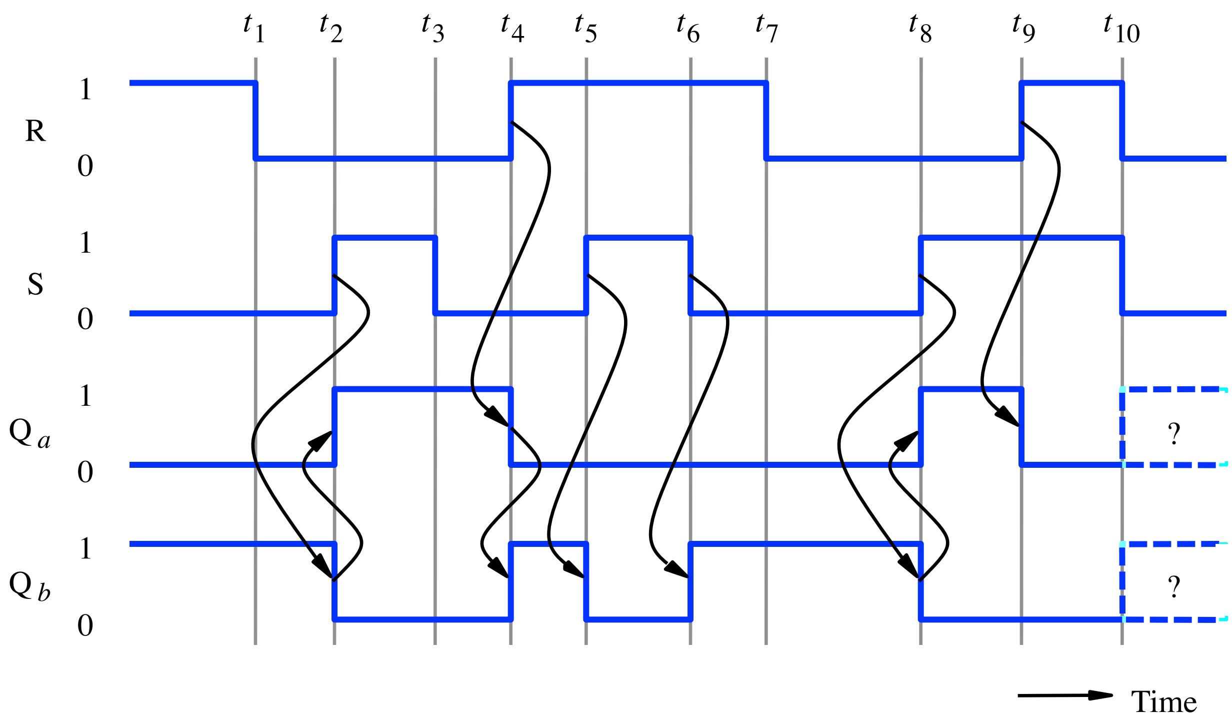
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

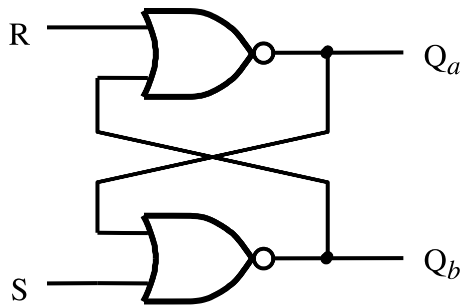
(b) Characteristic table



(c) Timing diagram

[Figure 5.4 from the textbook]

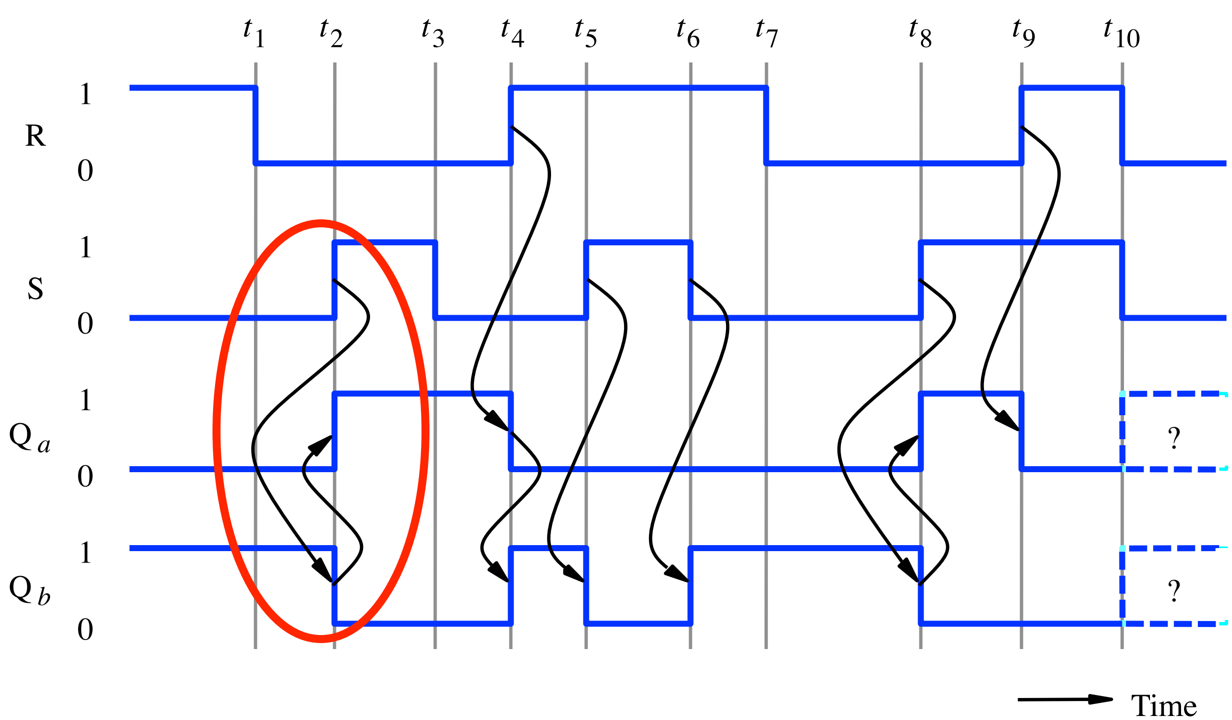
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

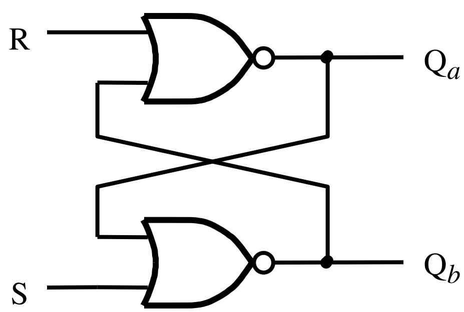
(b) Characteristic table



(c) Timing diagram

[Figure 5.4 from the textbook]

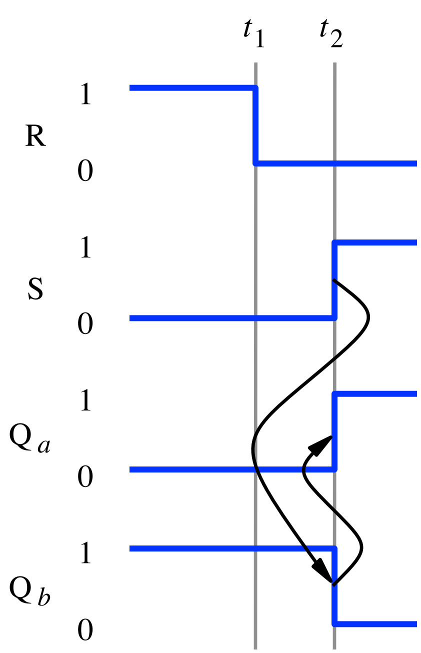
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

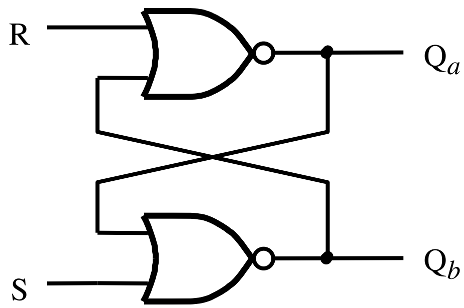
S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Characteristic table



(c) Timing diagram

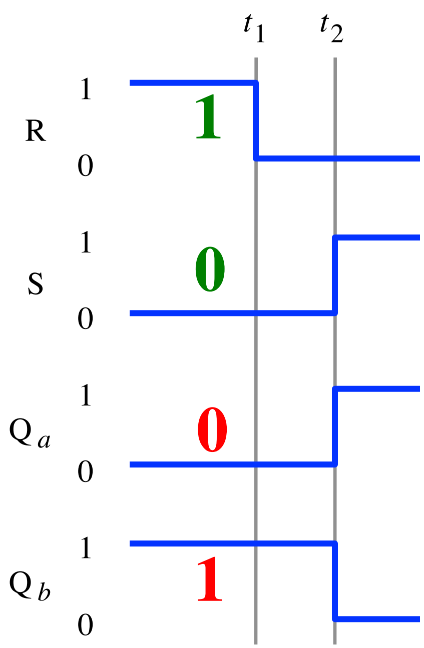
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

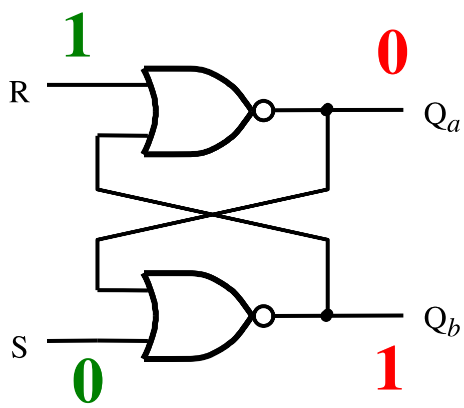
S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Characteristic table



(c) Timing diagram

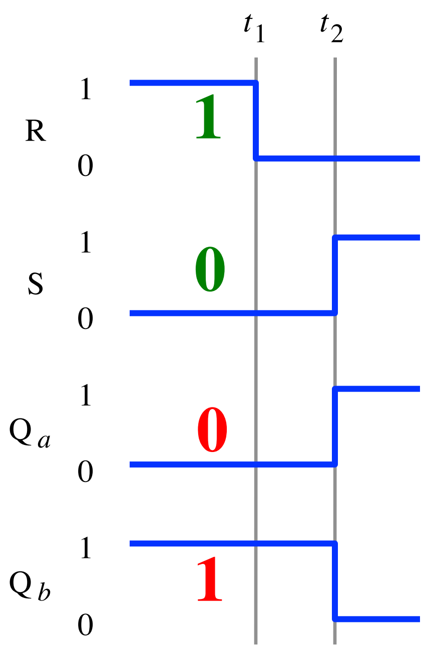
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

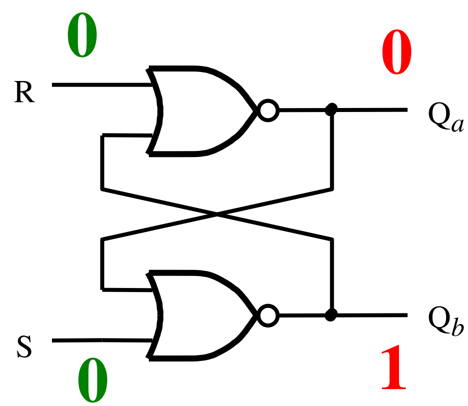
S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Characteristic table



(c) Timing diagram

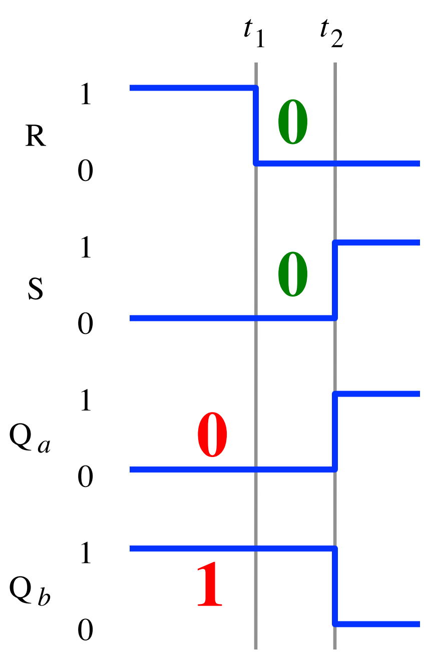
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

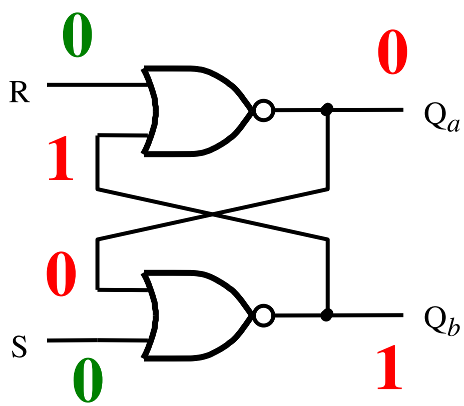
S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Characteristic table



(c) Timing diagram

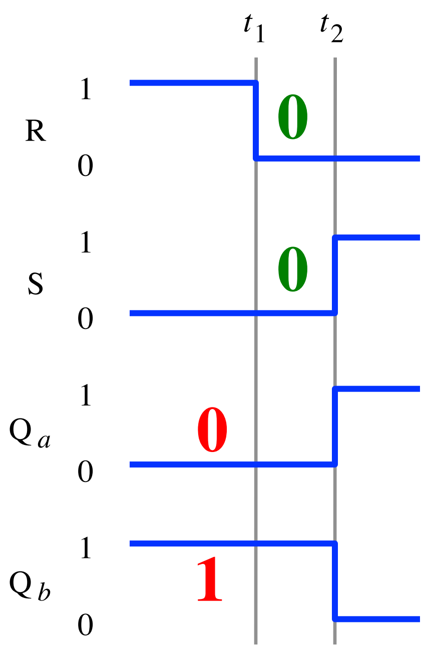
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

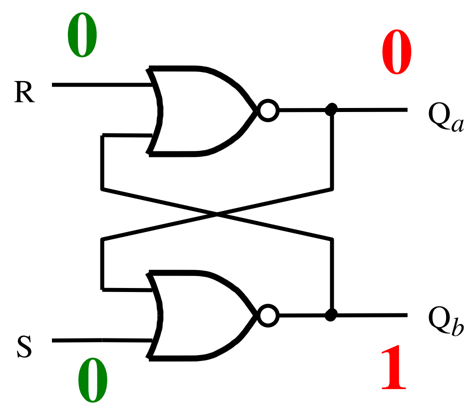
S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Characteristic table



(c) Timing diagram

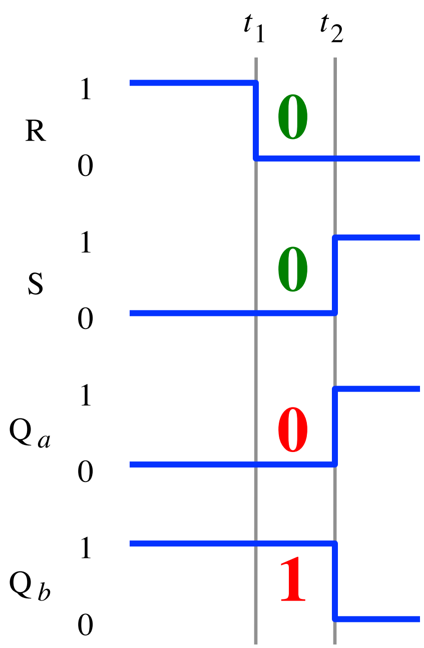
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

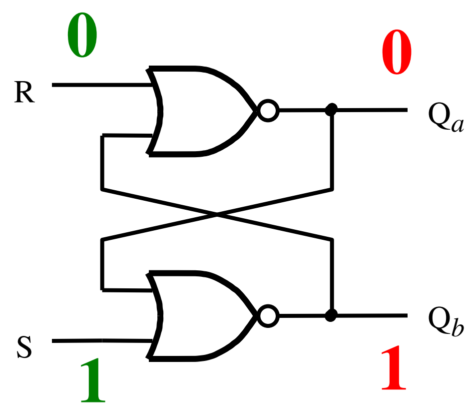
S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Characteristic table



(c) Timing diagram

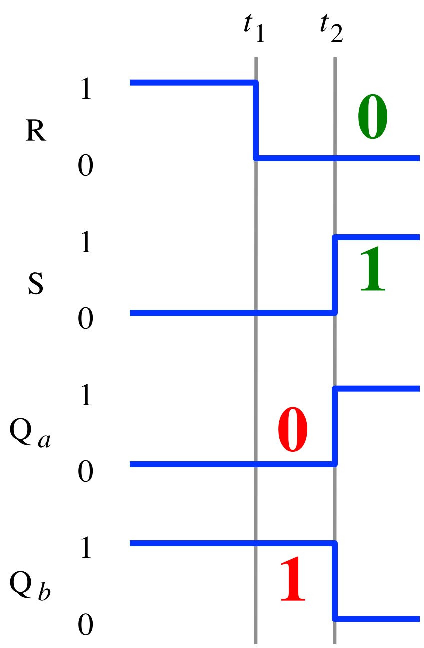
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

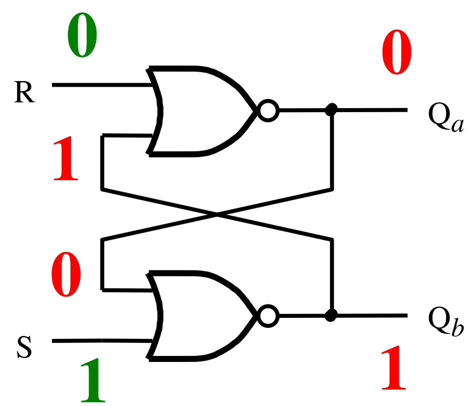
S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Characteristic table



(c) Timing diagram

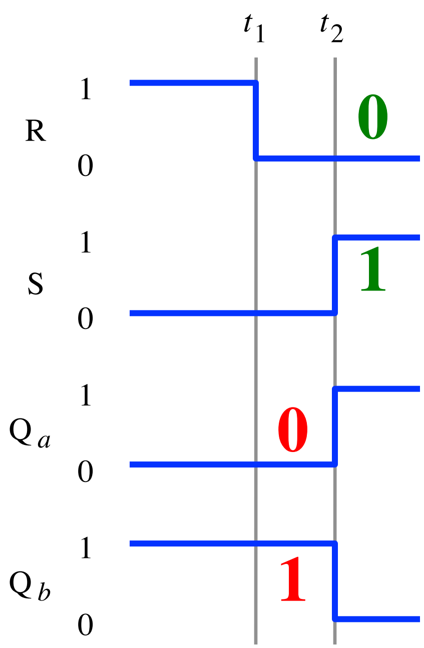
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

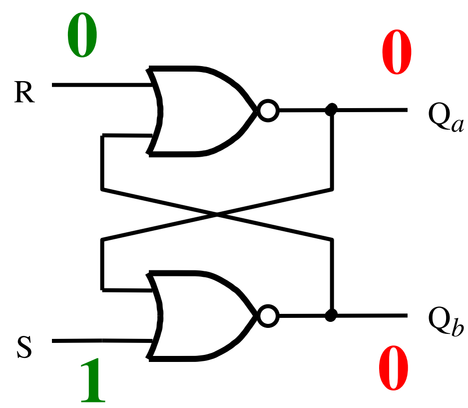
S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Characteristic table



(c) Timing diagram

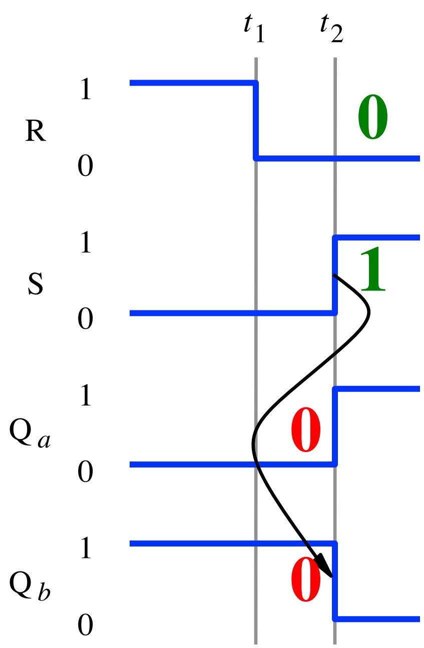
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

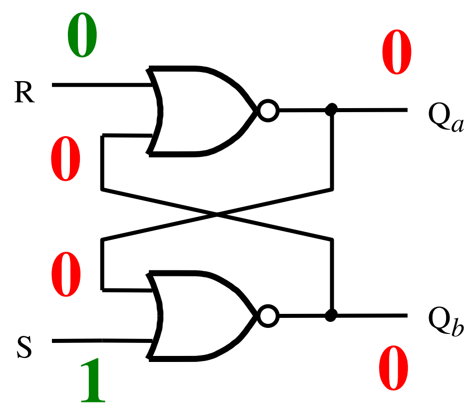
(b) Characteristic table



(c) Timing diagram

For a brief moment the latch goes through the undesirable state Q_a=0 and Q_b=0.

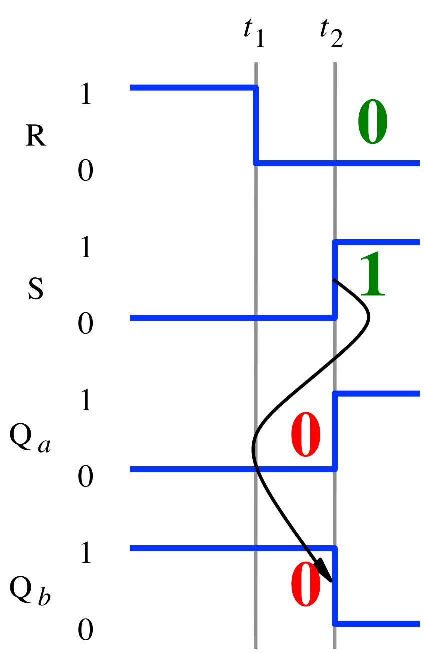
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

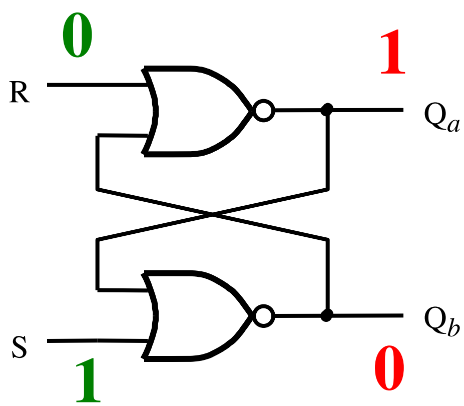
(b) Characteristic table



(c) Timing diagram

But these zeros loop around ...

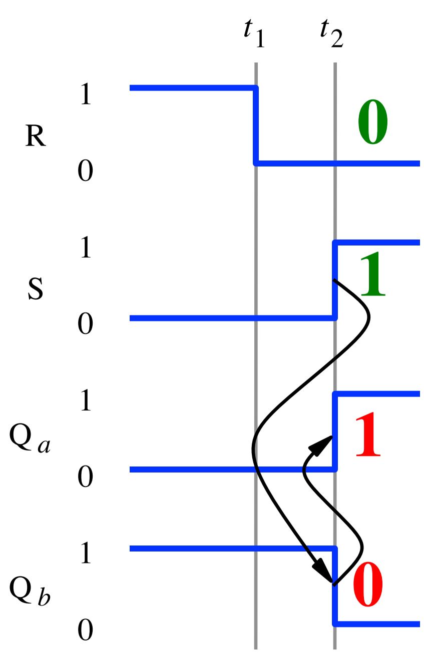
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

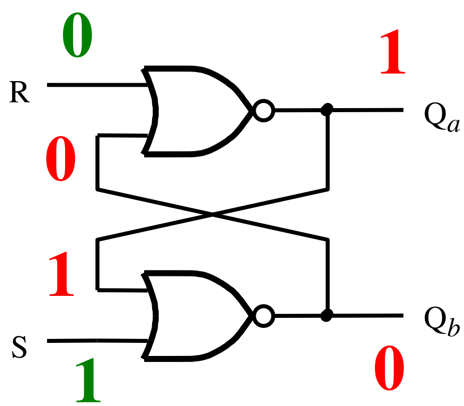
(b) Characteristic table



(c) Timing diagram

... and set it to Q_a=1 and Q_b=0.

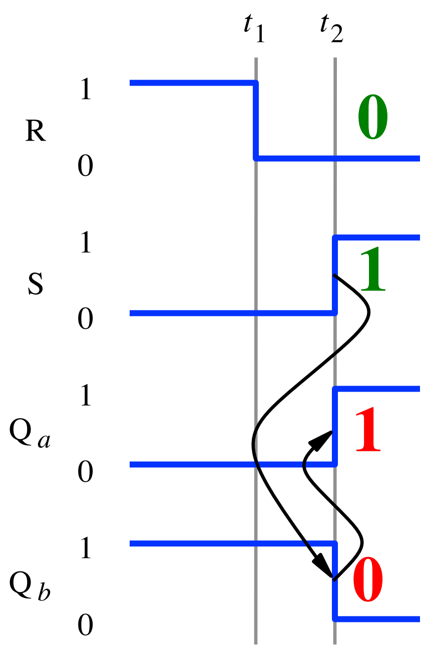
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

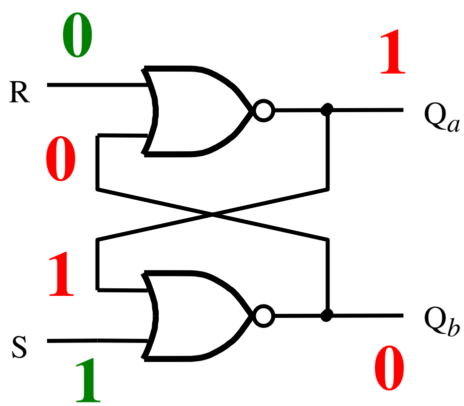
(b) Characteristic table



(c) Timing diagram

The new values also loop around ...

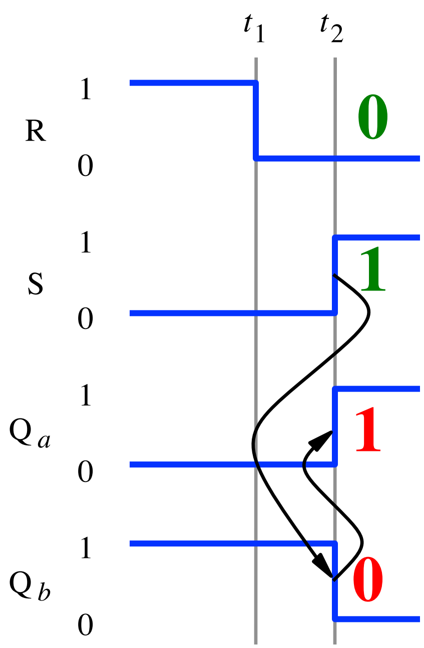
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

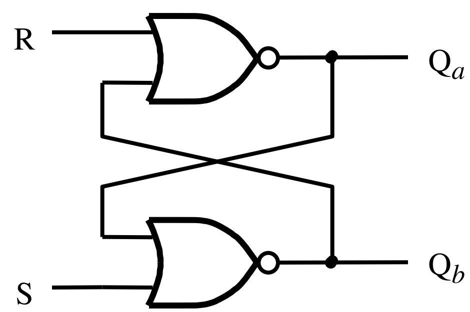
(b) Characteristic table



(c) Timing diagram

... but they leave the outputs the same.

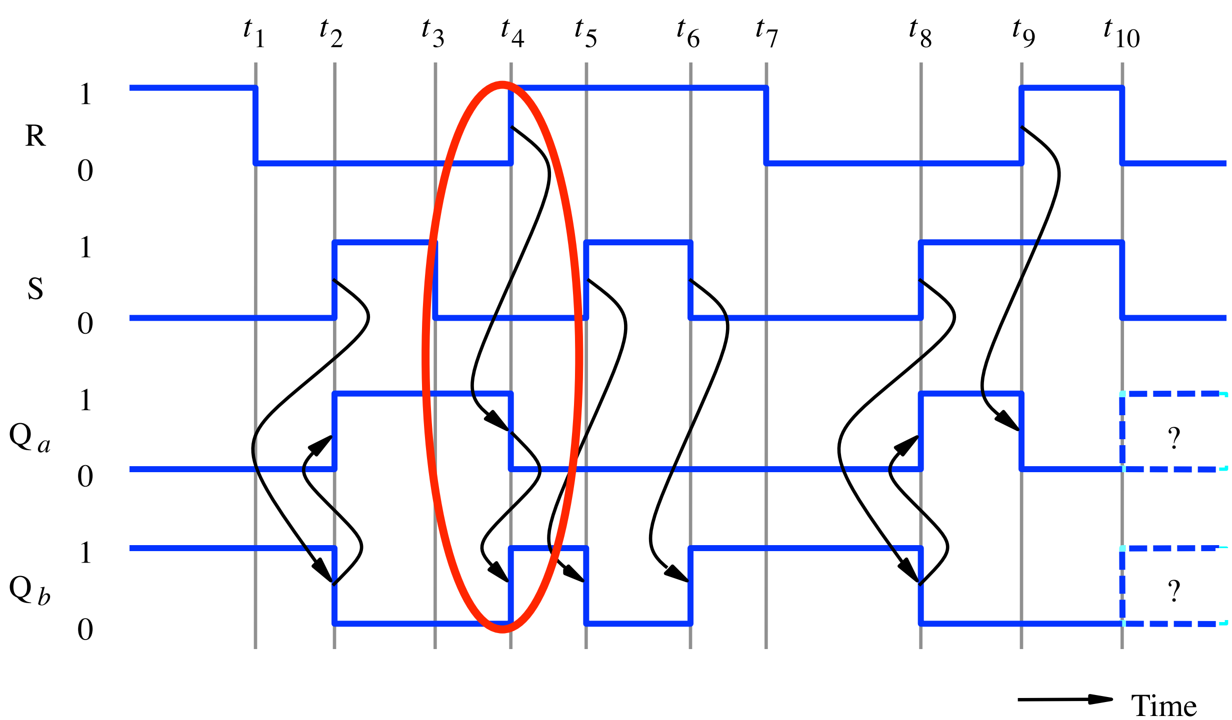
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

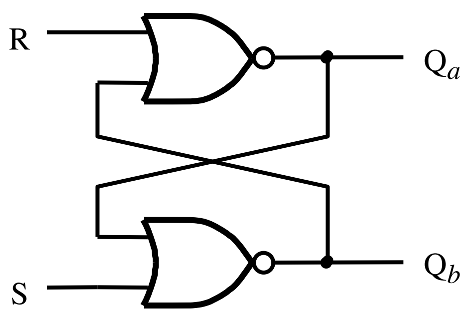
(b) Characteristic table



(c) Timing diagram

[Figure 5.4 from the textbook]

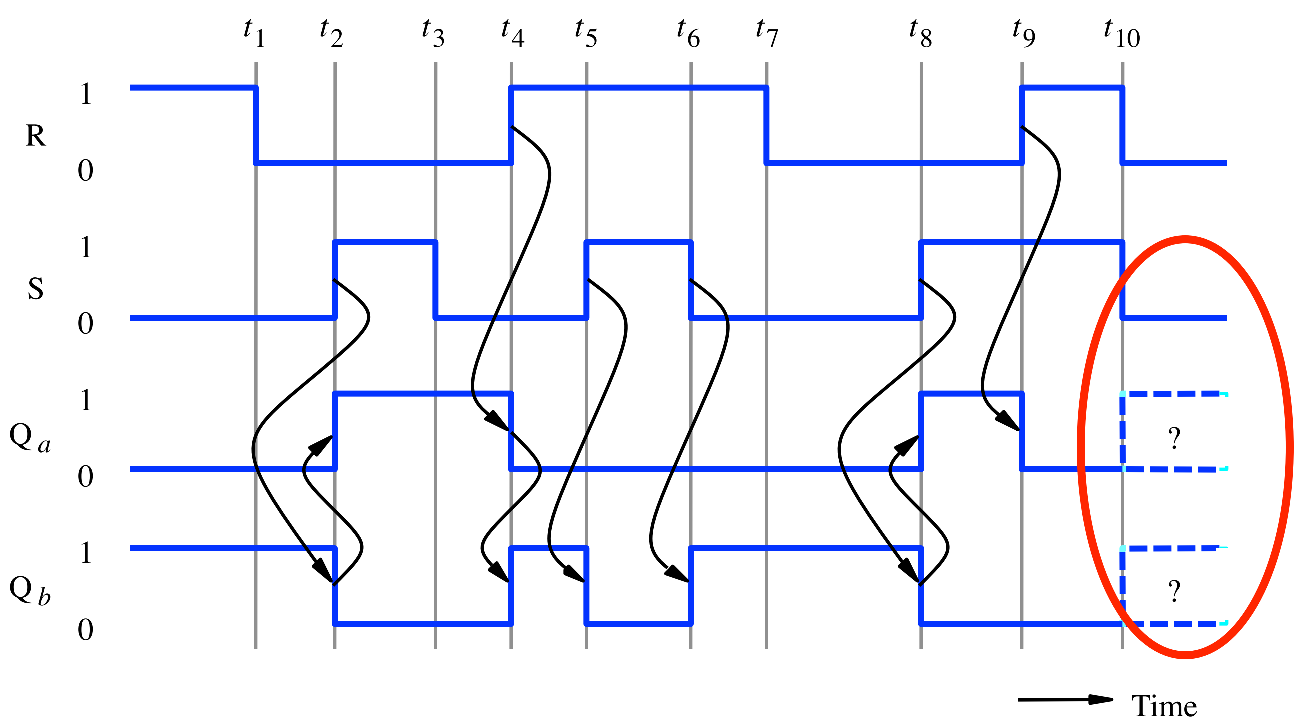
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

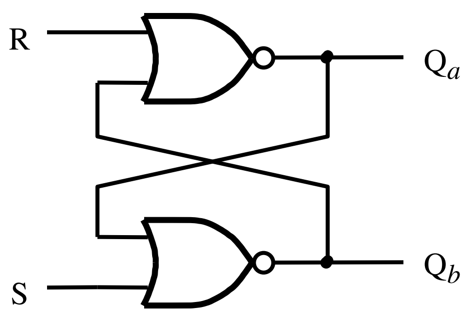
(b) Characteristic table



(c) Timing diagram

[Figure 5.4 from the textbook]

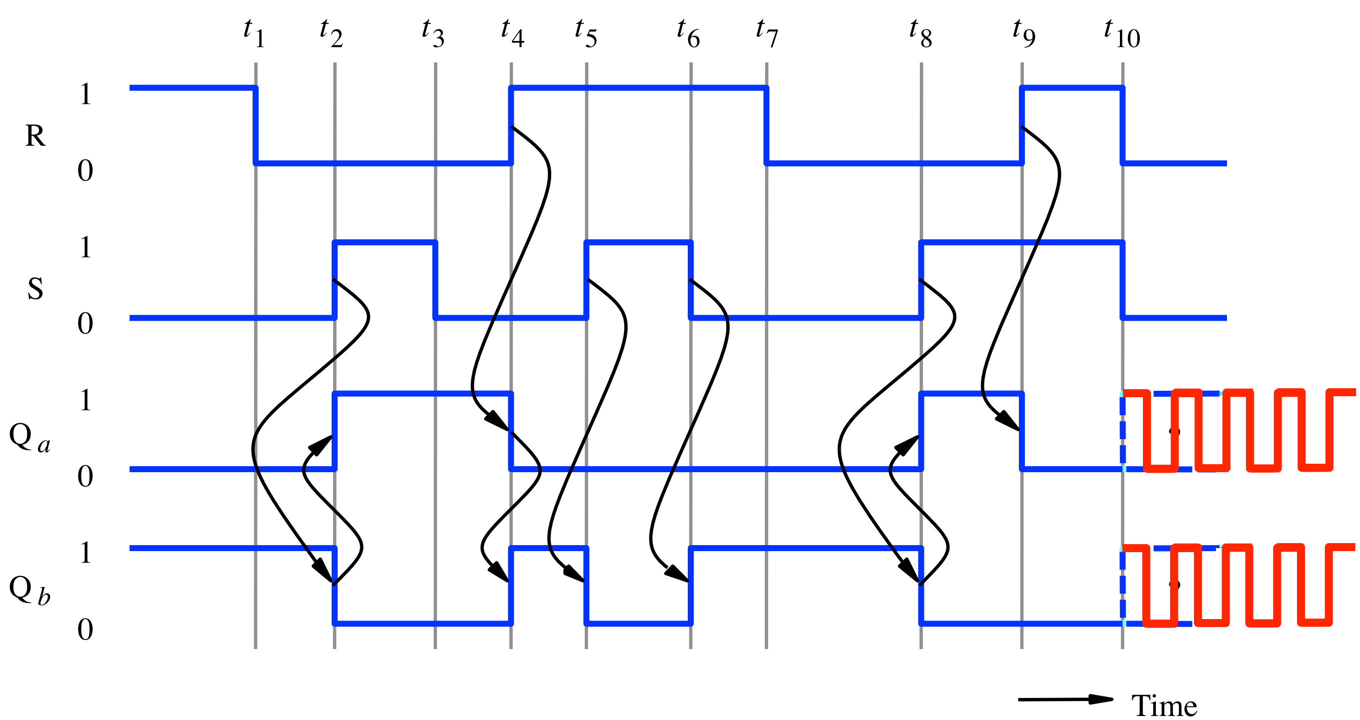
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

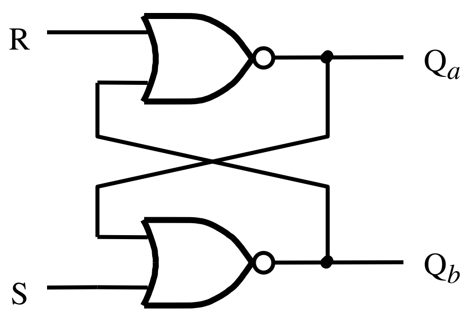
(b) Characteristic table



(c) Timing diagram

[Figure 5.4 from the textbook]

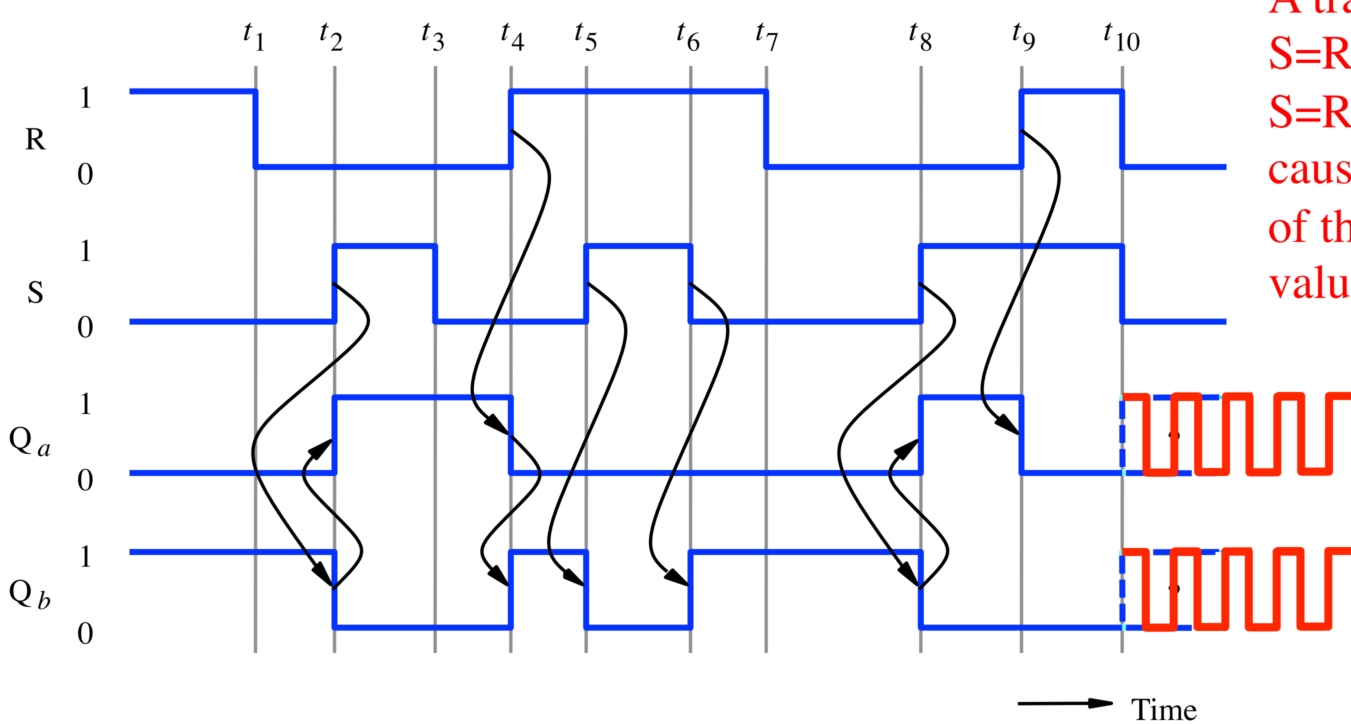
Timing Diagram for the Basic Latch with NOR Gates



(a) Circuit

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

(b) Characteristic table



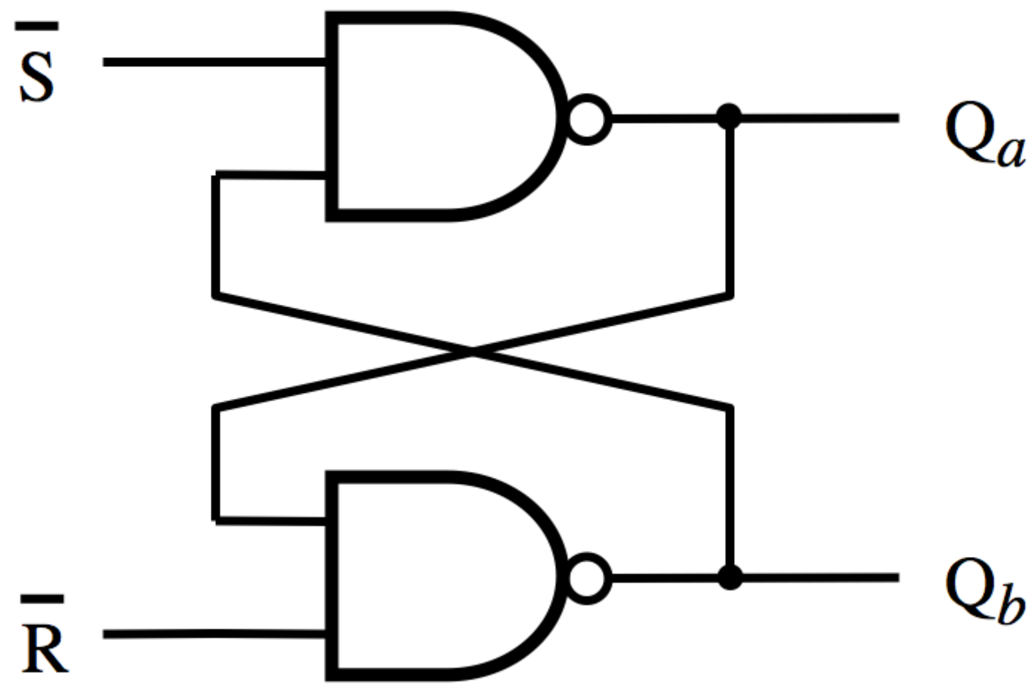
A transition from S=R=1 to S=R=0 causes oscillations of the two output values Qa and Qb.

(c) Timing diagram

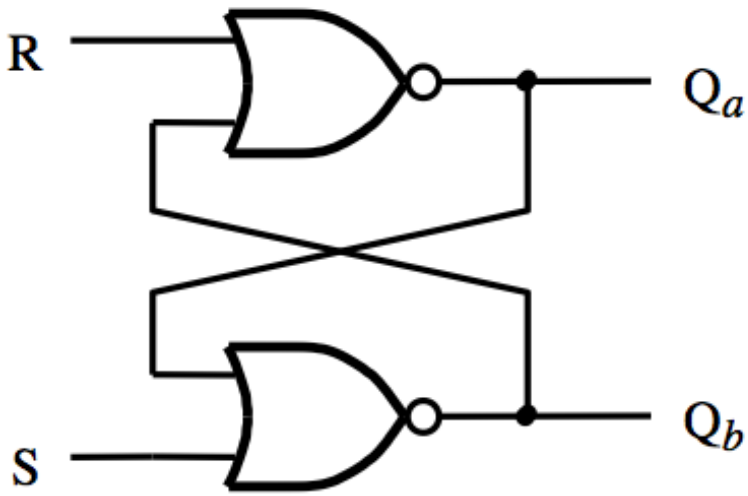
[Figure 5.4 from the textbook]

Basic Latch with NAND Gates

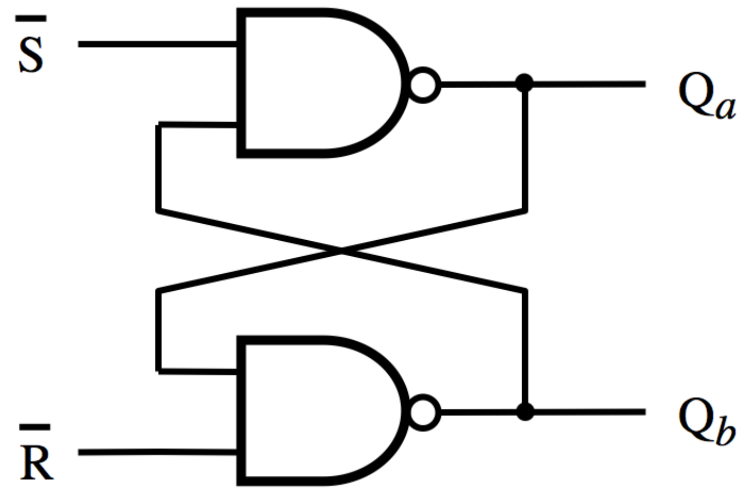
Circuit for the Basic Latch with NAND Gates



Basic Latch
(with NOR Gates)



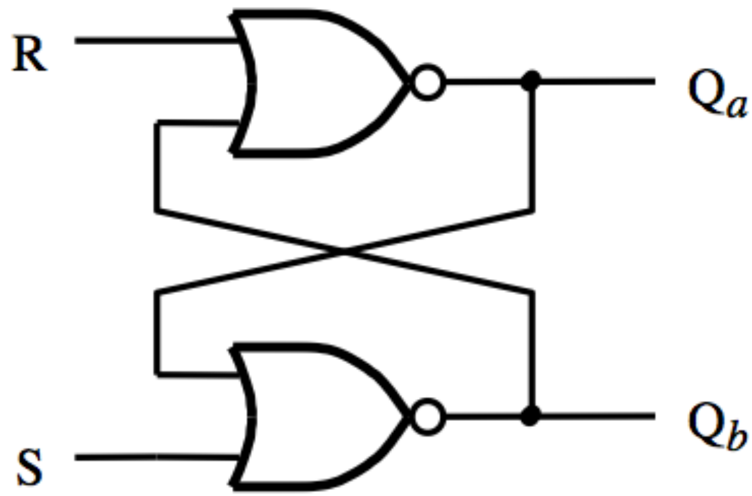
Basic Latch
(with NAND Gates)



Notice that in the NAND case the two inputs are swapped and negated.

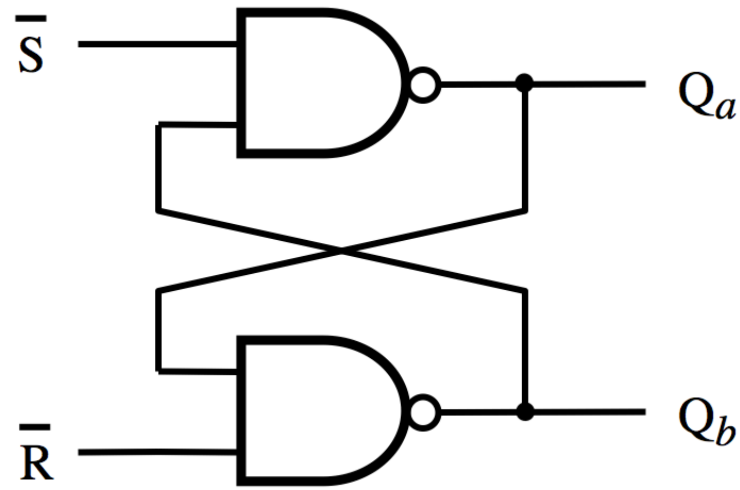
The labels of the outputs are the same in both cases.

Basic Latch
(with NOR Gates)



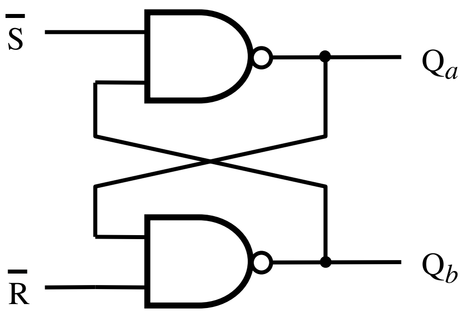
SR Latch

Basic Latch
(with NAND Gates)



$\overline{S}\overline{R}$ Latch

Circuit and Characteristic Table



(a) Circuit

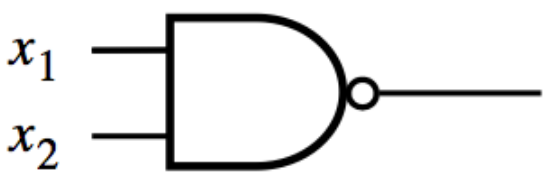
$\overline{S}$	$\overline{R}$	Q_a	Q_b	
0	0	1	1	
0	1	1	0	
1	0	0	1	
1	1	0/1	1/0	(no change)

(b) Characteristic table (version 1)

S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	1	1	

(c) Characteristic table (version 2)

NAND Gate

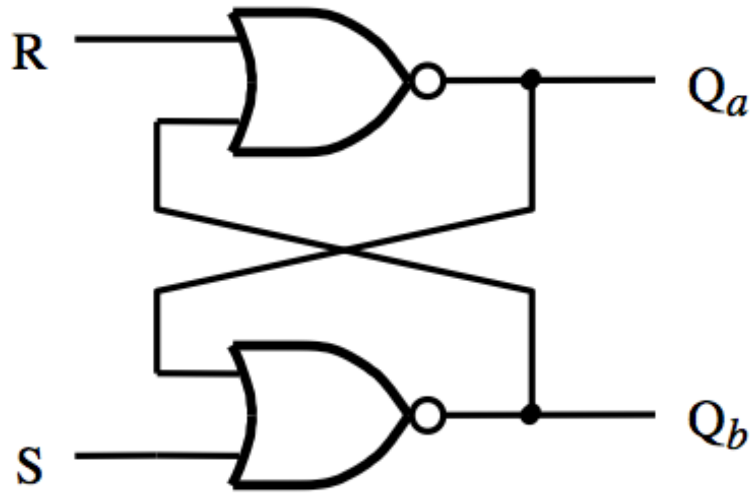


$\overline{x_1 \cdot x_2}$

NAND Gate Truth table

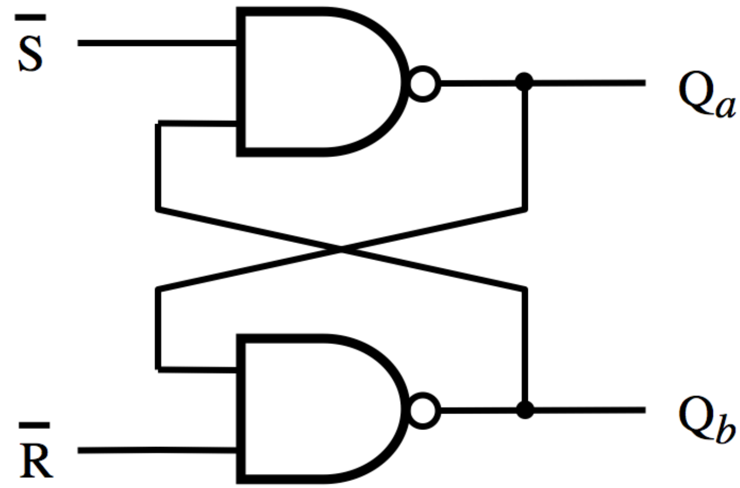
x_1	x_2	f
0	0	1
0	1	1
1	0	1
1	1	0

Basic Latch
(with NOR Gates)



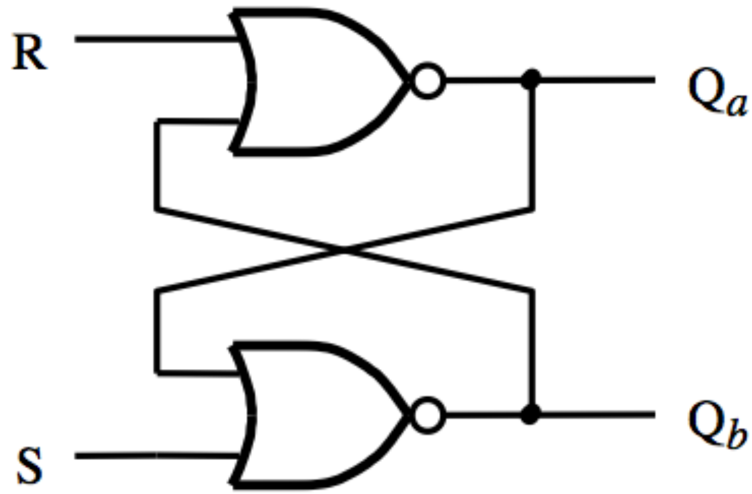
S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	0	0	

Basic Latch
(with NAND Gates)



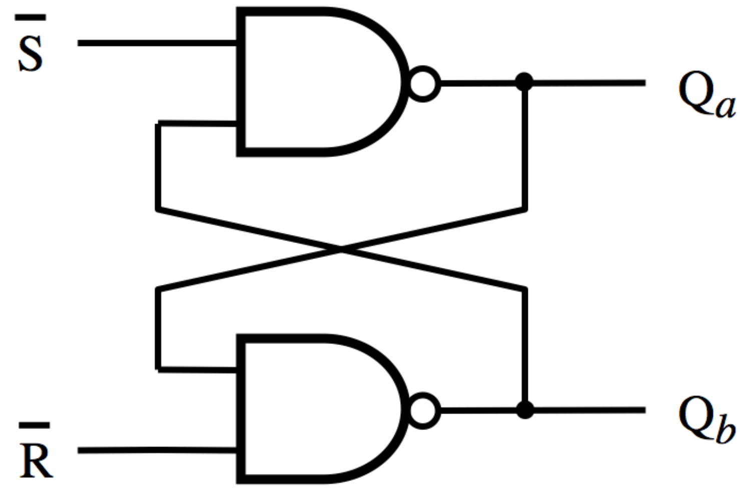
S	R	Q_a	Q_b	
0	0	0/1	1/0	(no change)
0	1	0	1	
1	0	1	0	
1	1	1	1	

Basic Latch
(with NOR Gates)



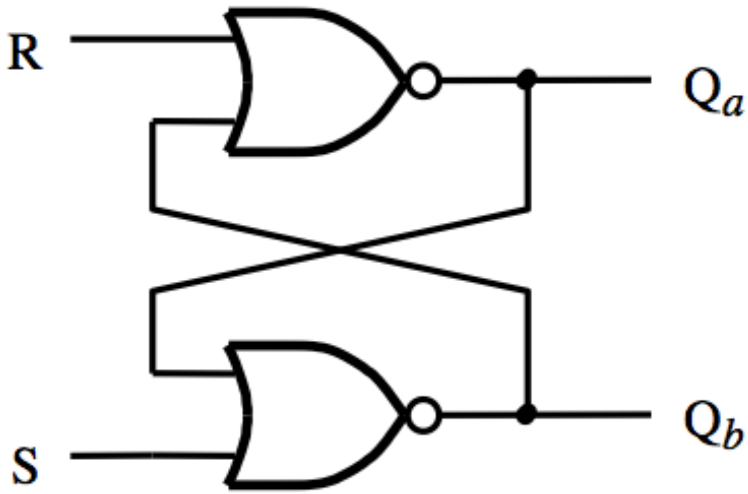
S	R	Q_a	Q_b		
0	0	0/1	1/0	(no change)	Latch
0	1	0	1		Reset
1	0	1	0		Set
1	1	0	0		Undesirable

Basic Latch
(with NAND Gates)



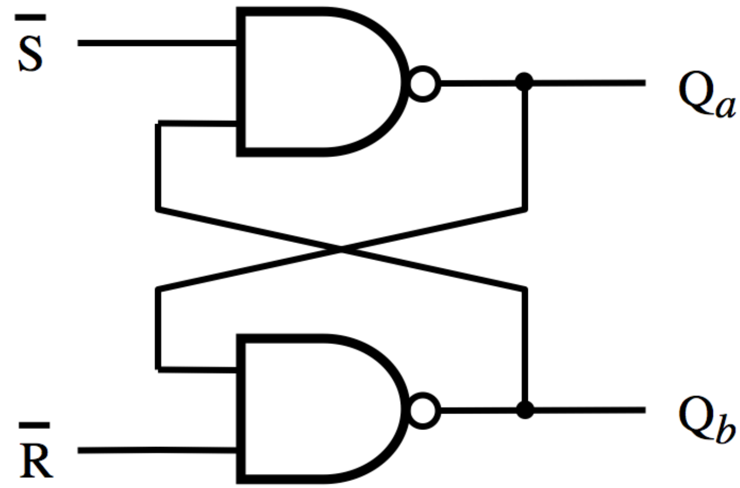
S	R	Q_a	Q_b		
0	0	0/1	1/0	(no change)	Latch
0	1	0	1		Reset
1	0	1	0		Set
1	1	1	1		Undesirable

Basic Latch
(with NOR Gates)



S	R	Q_a	Q_b		
0	0	0/1	1/0	(no change)	Latch
0	1	0	1		Reset
1	0	1	0		Set
1	1	0	0		Undesirable

Basic Latch
(with NAND Gates)



S	R	Q_a	Q_b		
0	0	0/1	1/0	(no change)	Latch
0	1	0	1		Reset
1	0	1	0		Set
1	1	1	1		Undesirable

The two characteristic tables are the same
(except for the last row, which is the undesirable configuration).

Oscillations and Undesirable States

- **The basic latch with NAND gates also suffers form oscillation problems, similar to the basic latch implemented with NOR gates.**

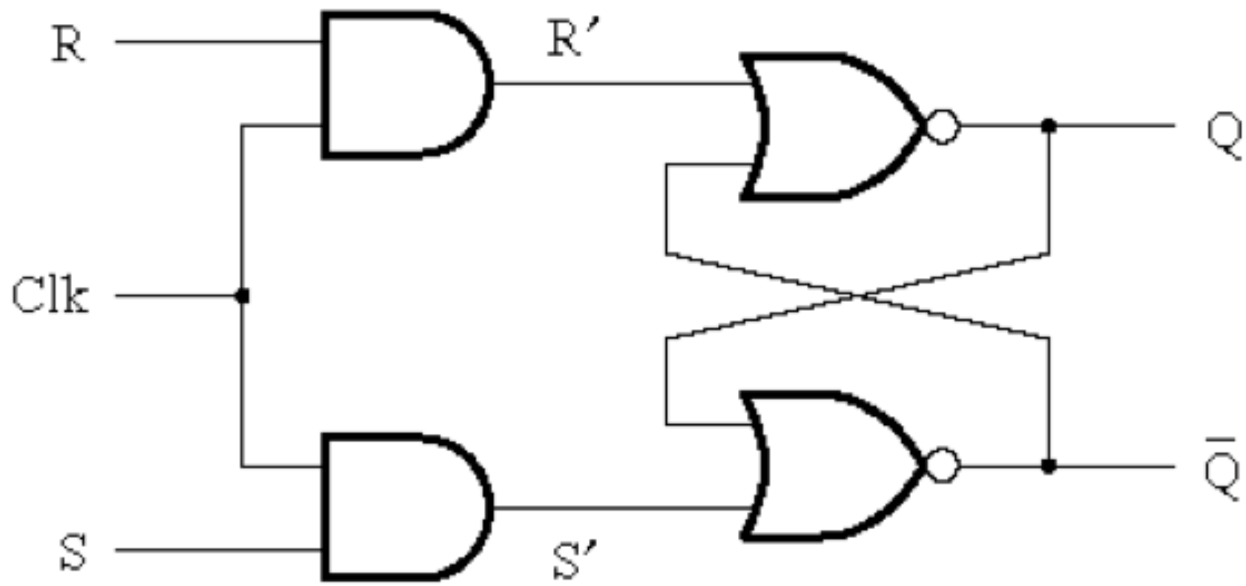
- **Try to do this analysis on your own.**

Gated SR Latch

Motivation

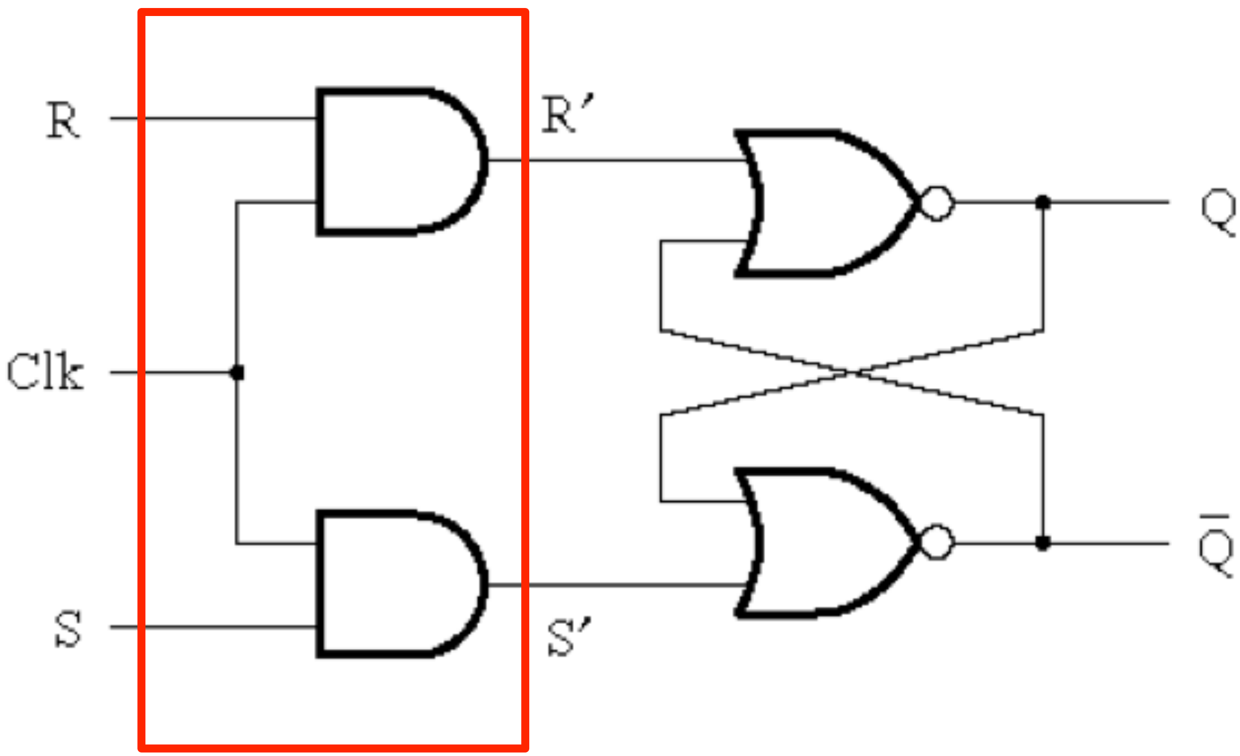
- The basic latch changes its state when the input signals change
- It is hard to control when these input signals will change and thus it is hard to know when the latch may change its state.
- We want to have something like an Enable input
- In this case it is called the “Clock” input because it is desirable for the state changes to be synchronized

Circuit Diagram for the Gated SR Latch



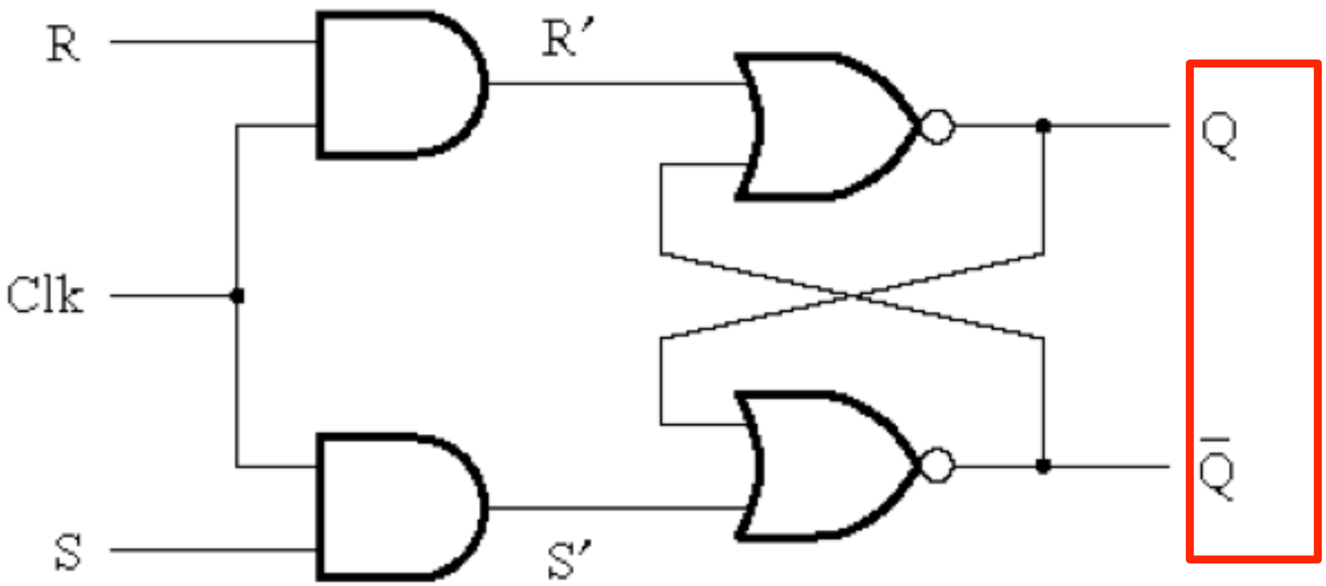
[Figure 5.5a from the textbook]

Circuit Diagram for the Gated SR Latch



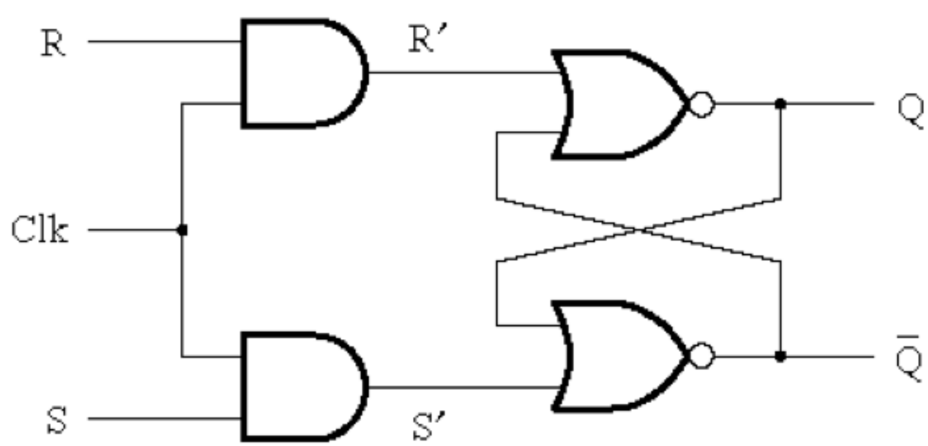
This is the "gate"
of the gated latch

Circuit Diagram for the Gated SR Latch



Notice that these are complements of each other

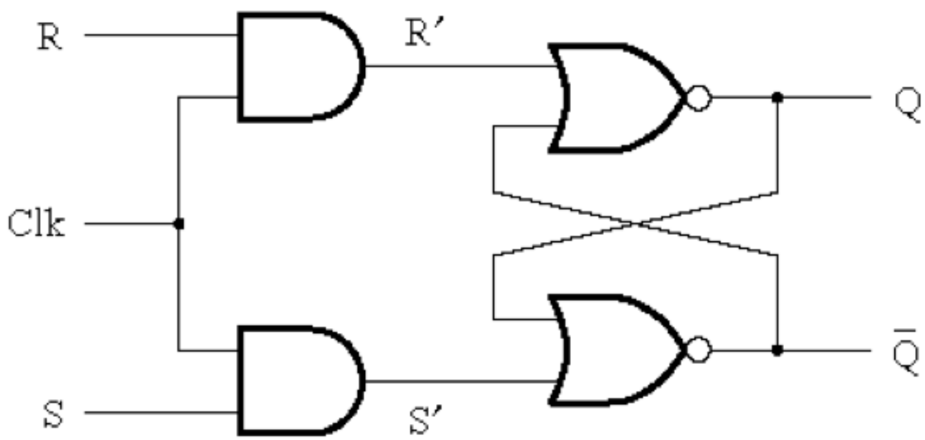
Circuit Diagram and Characteristic Table for the Gated SR Latch

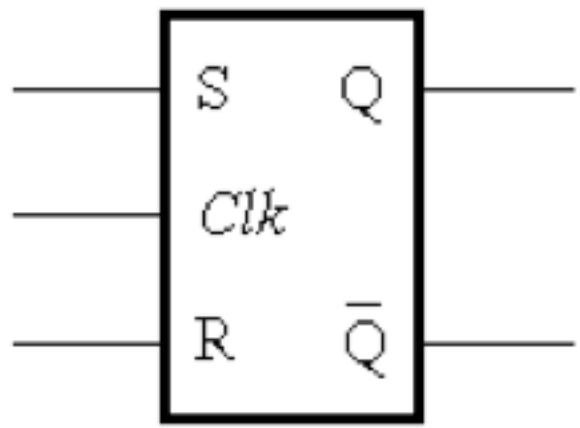


Clk	S	R	Q(t + 1)
0	x	x	Q(t) (no change)
1	0	0	Q(t) (no change)
1	0	1	0
1	1	0	1
1	1	1	x

[Figure 5.5a-b from the textbook]

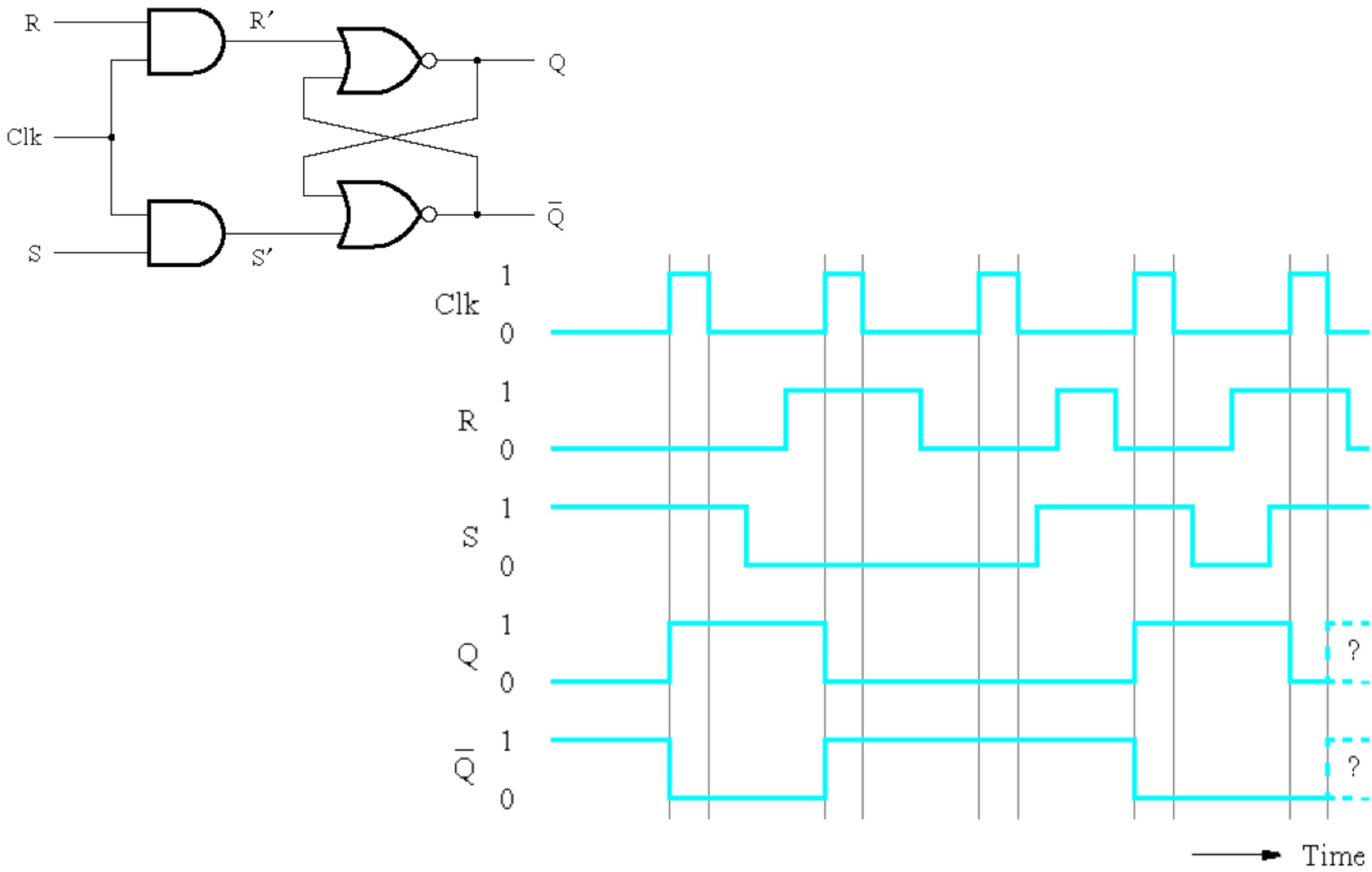
Circuit Diagram and Graphical Symbol for the Gated SR Latch





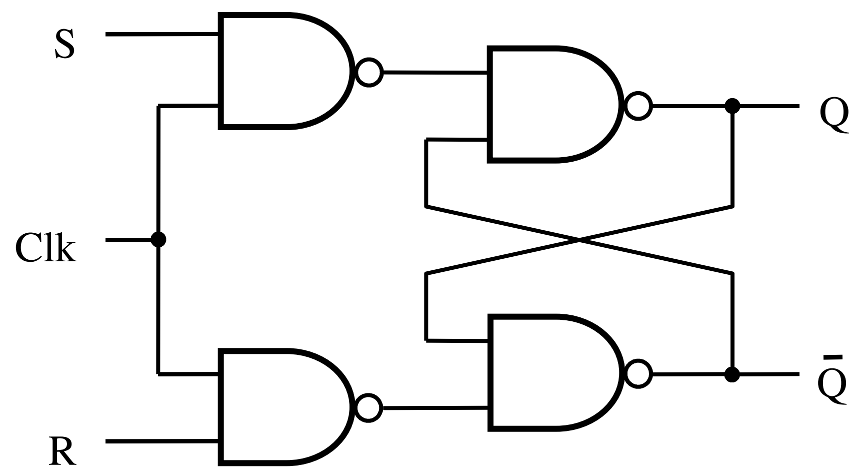
[Figure 5.5a,c from the textbook]

Timing Diagram for the Gated SR Latch



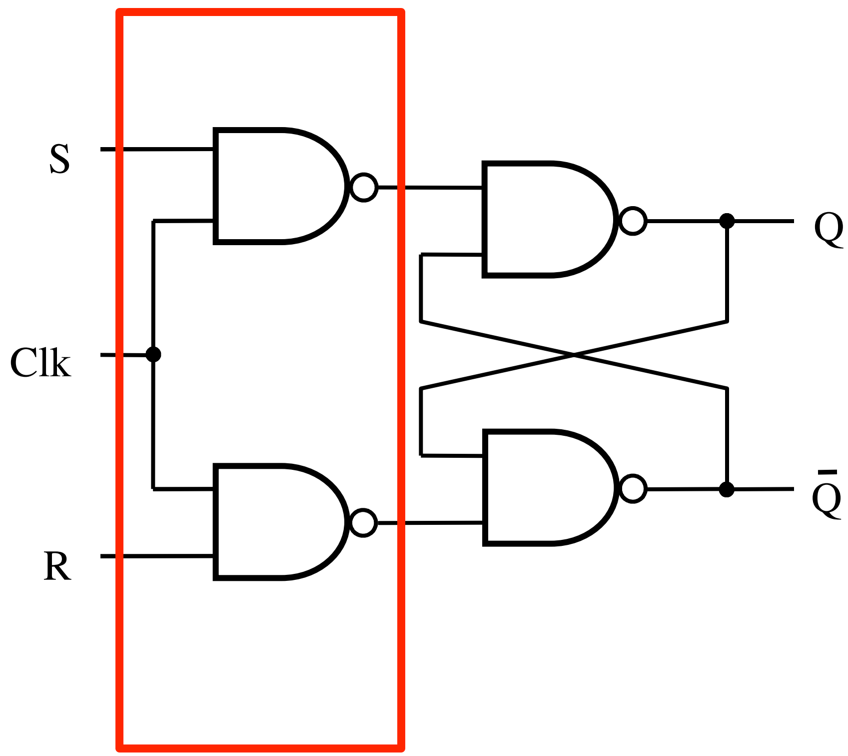
[Figure 5.5c from the textbook]

Gated SR latch with NAND gates



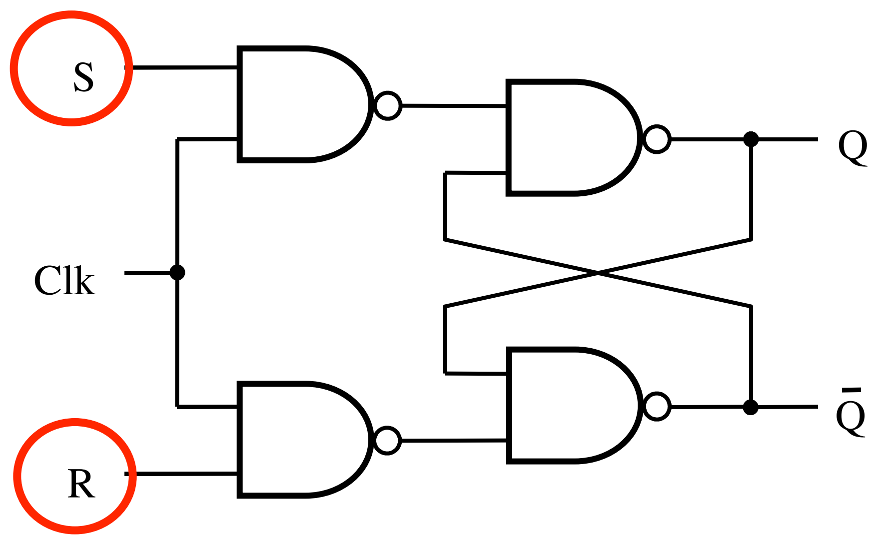
[Figure 5.6 from the textbook]

Gated SR latch with NAND gates



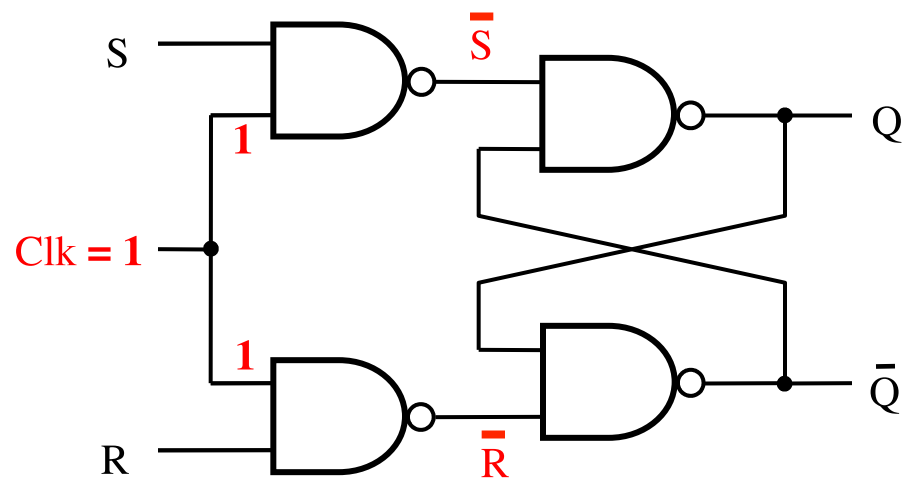
In this case the "gate" is constructed using NAND gates! Not AND gates.

Gated SR latch with NAND gates



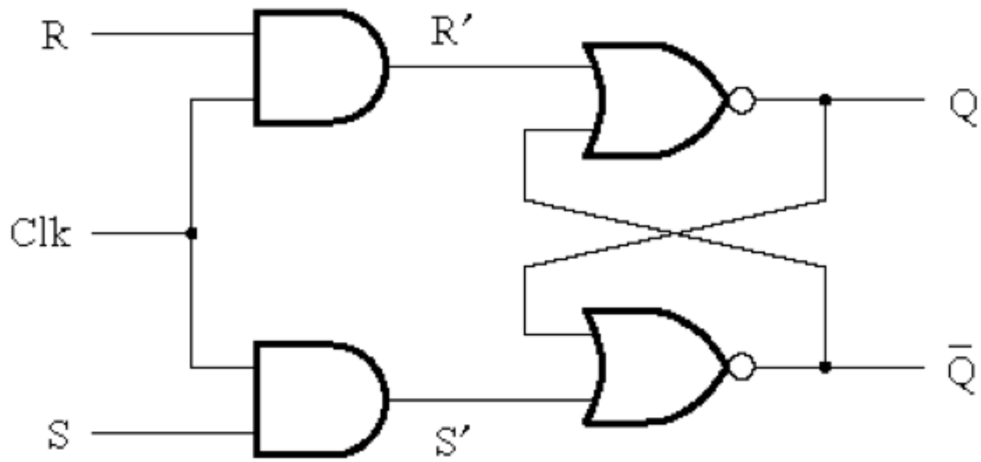
Also, notice that the positions of S and R are now swapped.

Gated SR latch with NAND gates

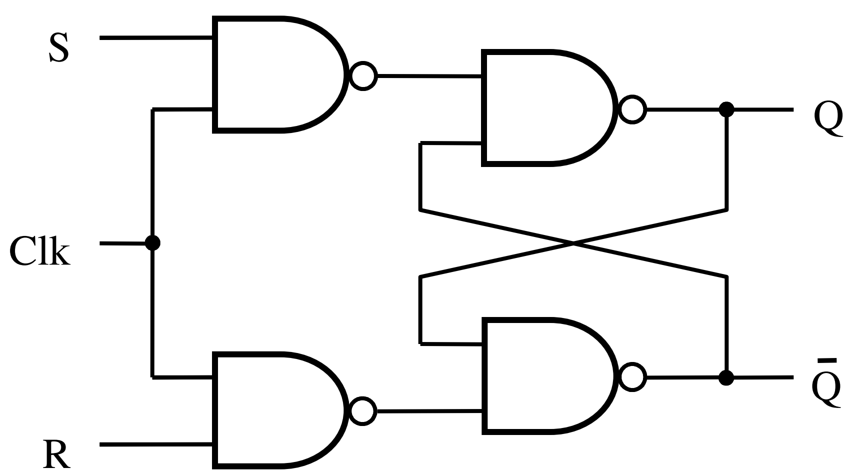


Finally, notice that when Clk=1 this turns into the basic latch with NAND gates, i.e., the $\overline{S}\,\overline{R}$ Latch.

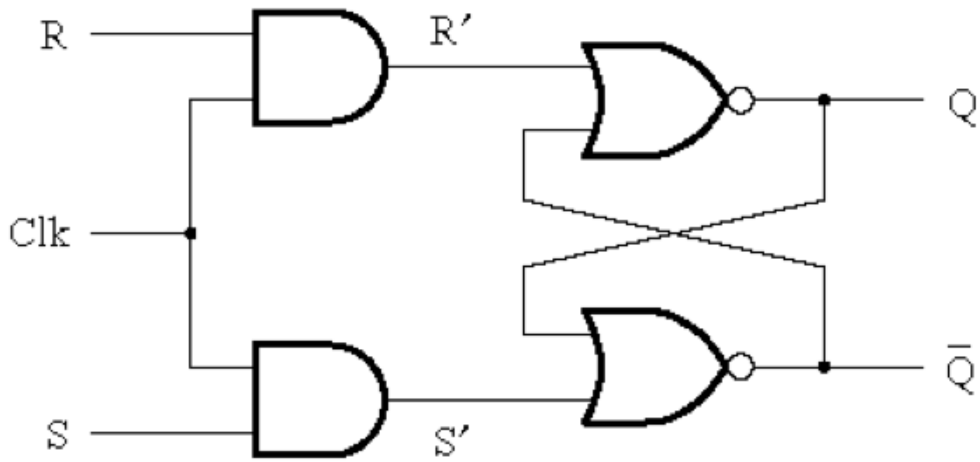
Gated SR latch with NOR gates

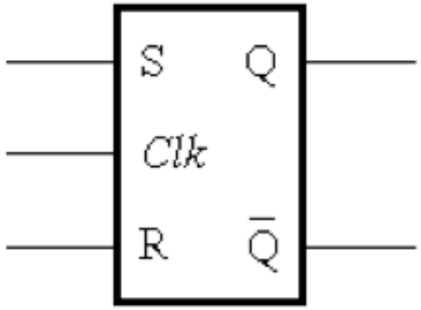


Gated SR latch with NAND gates

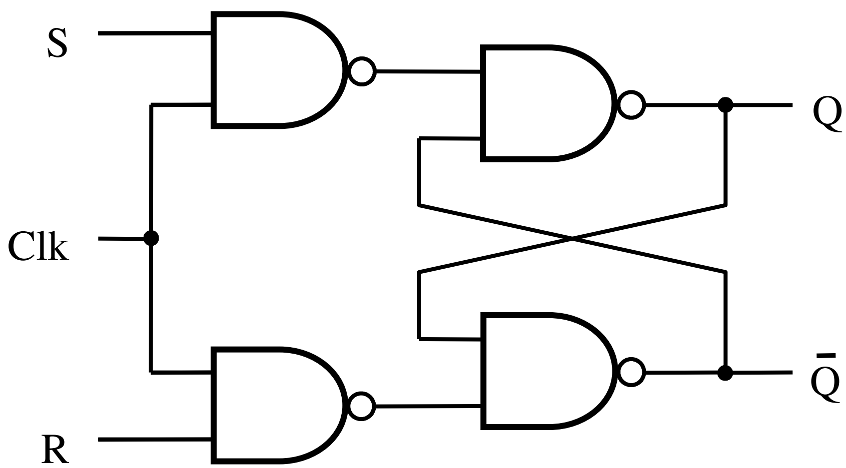


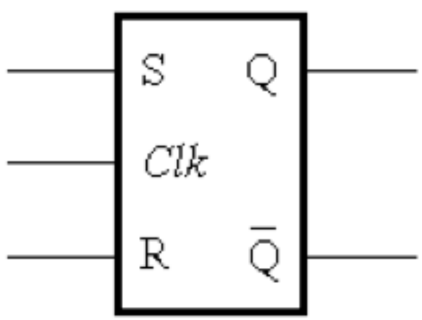
Gated SR latch with NOR gates





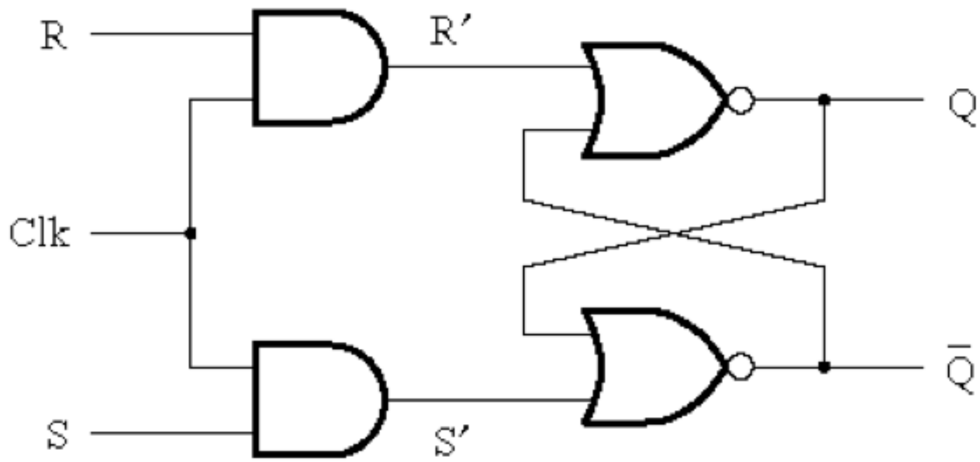
Gated SR latch with NAND gates





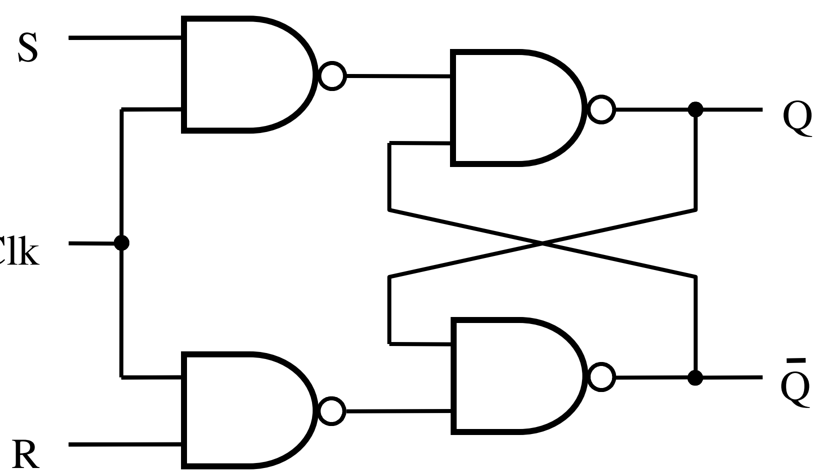
Graphical symbols are the same

Gated SR latch with NOR gates



Clk	S	R	$Q(t+1)$
0	x	x	$Q(t)$ (no change)
1	0	0	$Q(t)$ (no change)
1	0	1	0
1	1	0	1
1	1	1	x (undesirable)

Gated SR latch with NAND gates



Clk	S	R	$Q(t+1)$
0	x	x	$Q(t)$ (no change)
1	0	0	$Q(t)$ (no change)
1	0	1	0
1	1	0	1
1	1	1	x (undesirable)

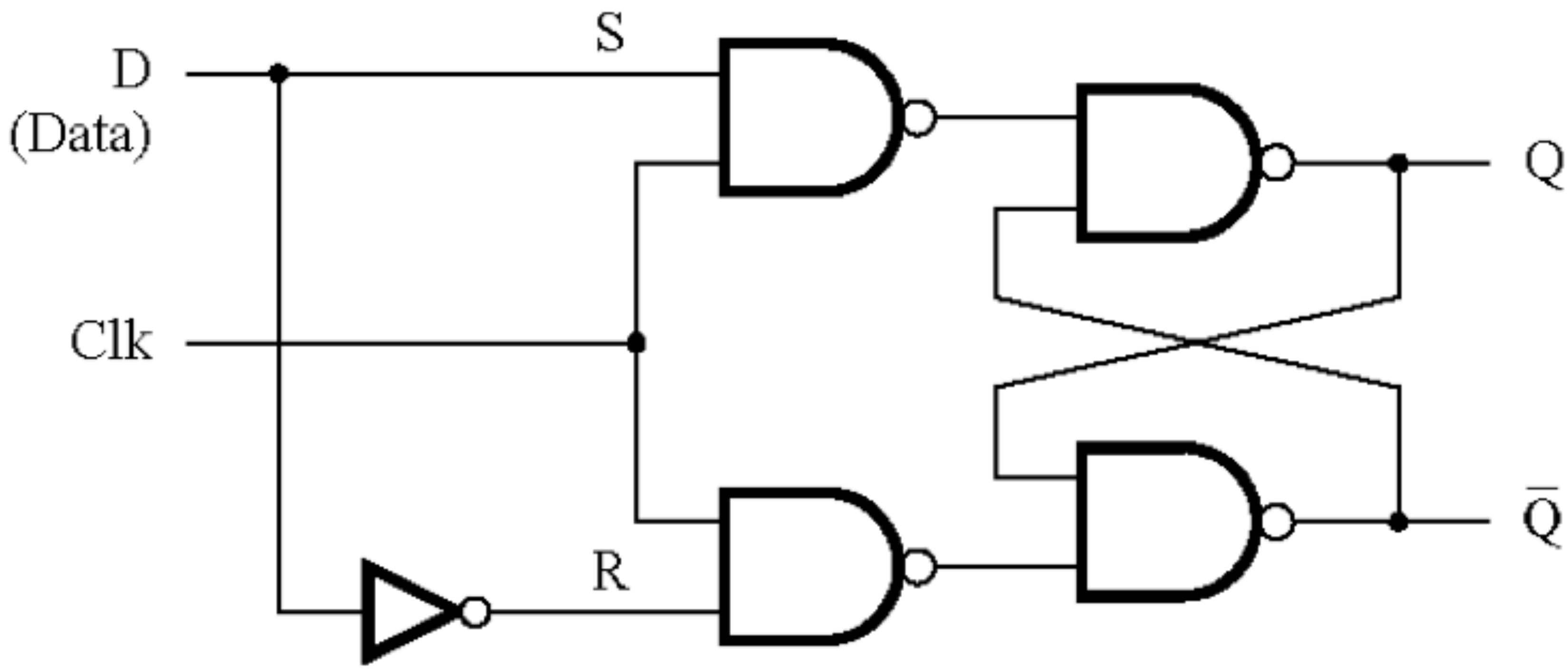
Characteristic tables are the same

Gated D Latch

Motivation

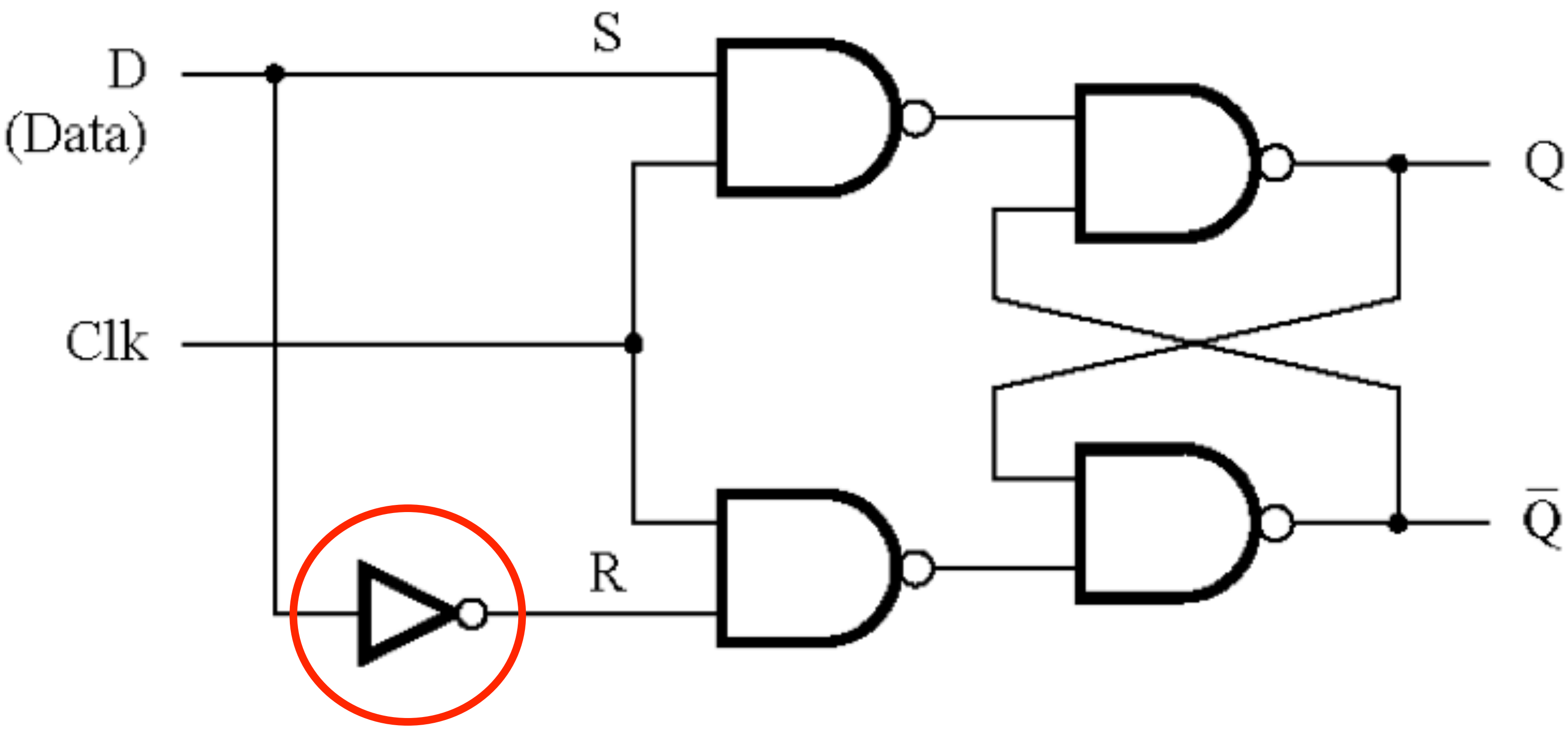
- Dealing with two inputs (S and R) could be messy. For example, we may have to reset the latch before some operations in order to store a specific value but the reset may not be necessary depending on the current state of the latch.
- Why not just have one input and call it D.
- The D latch can be constructed using a simple modification of the SR latch.

Circuit Diagram for the Gated D Latch



[Figure 5.7a from the textbook]

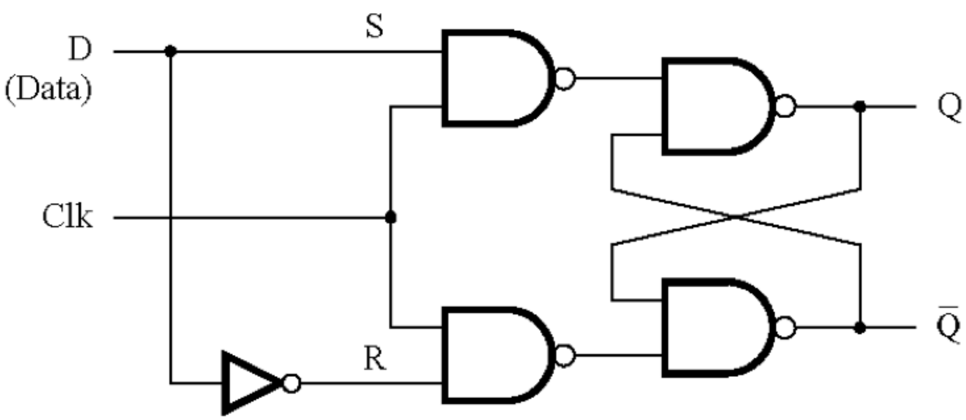
Circuit Diagram for the Gated D Latch



This is the only
new thing here.

[Figure 5.7a from the textbook]

Circuit Diagram and Characteristic Table for the Gated D Latch

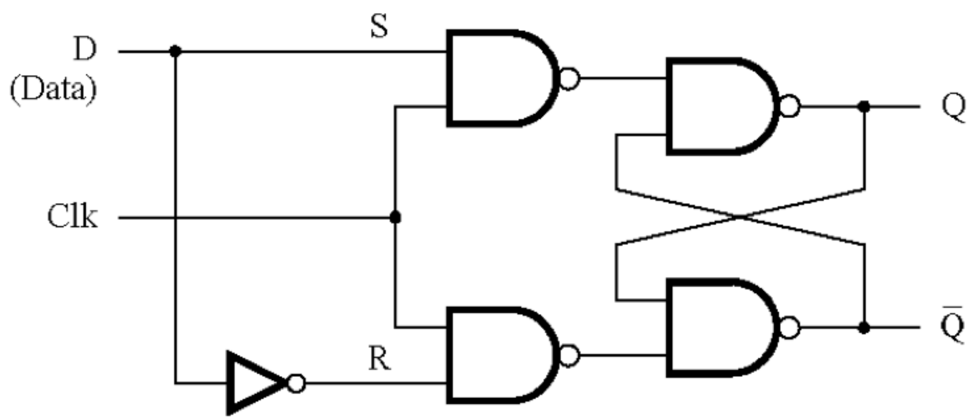


Clk	D	$Q(t+1)$
0	x	$Q(t)$
1	0	0
1	1	1

Note that it is now impossible to have S=R=1.

[Figure 5.7a,b from the textbook]

Circuit Diagram and Characteristic Table for the Gated D Latch

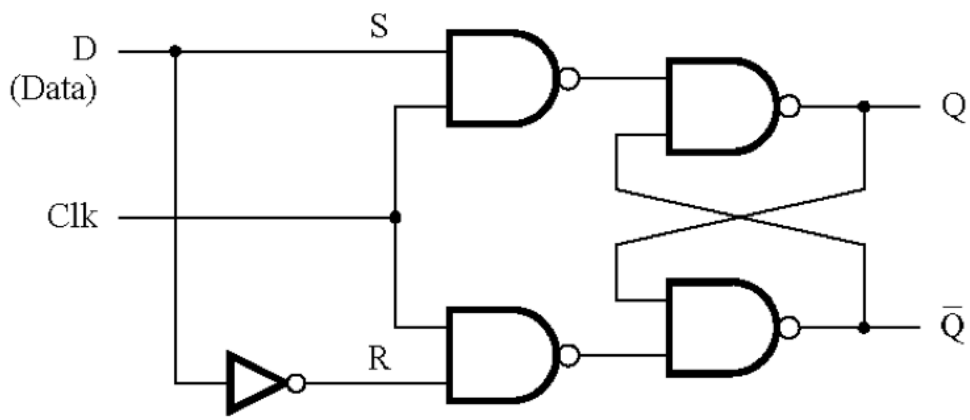


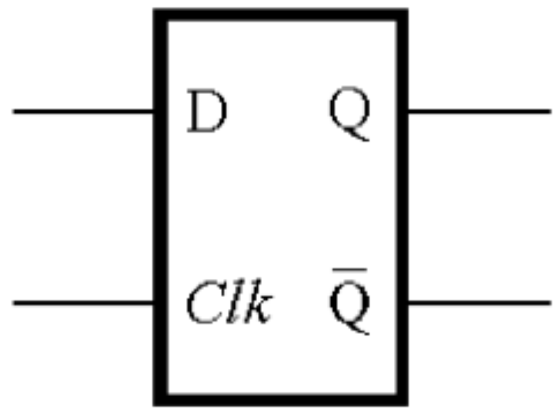
Clk	D	$Q(t+1)$
0	x	$Q(t)$
1	0	0
1	1	1

When Clk=1 the output follows the D input.
When Clk=0 the output cannot be changed.

[Figure 5.7a,b from the textbook]

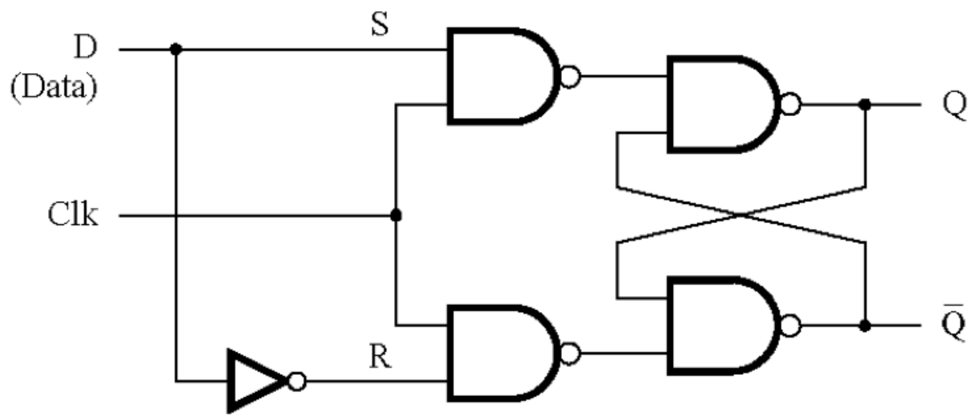
Circuit Diagram and Graphical Symbol for the Gated D Latch

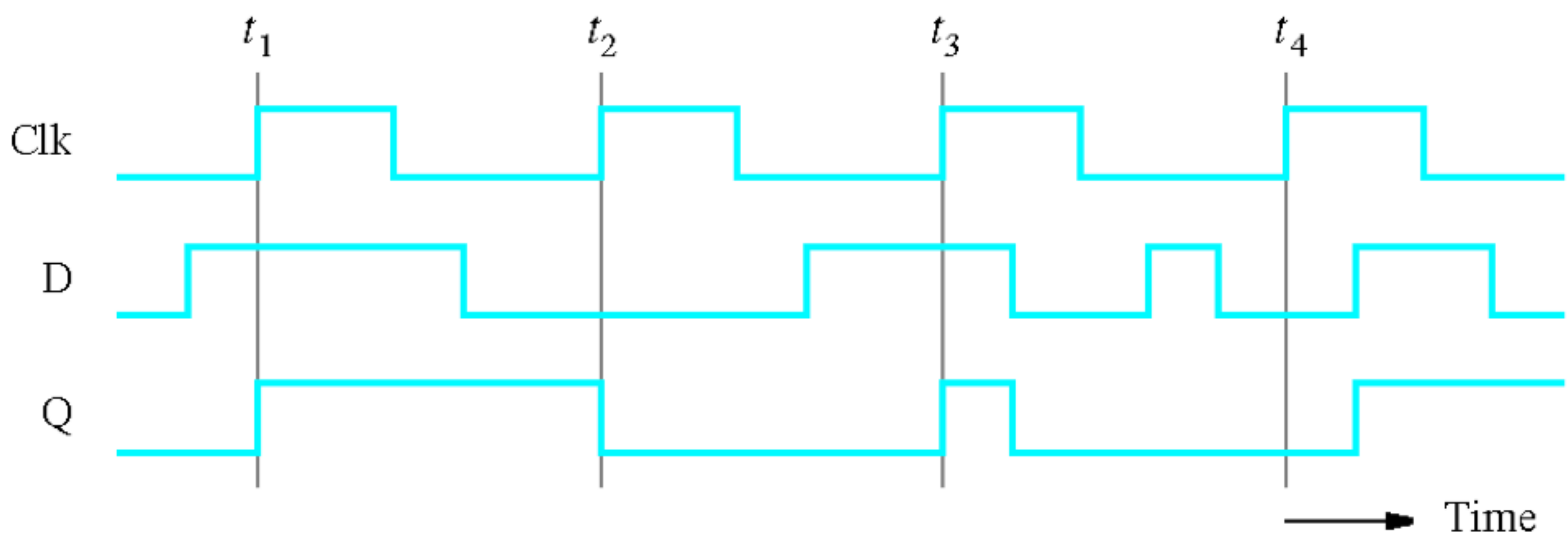




[Figure 5.7a,c from the textbook]

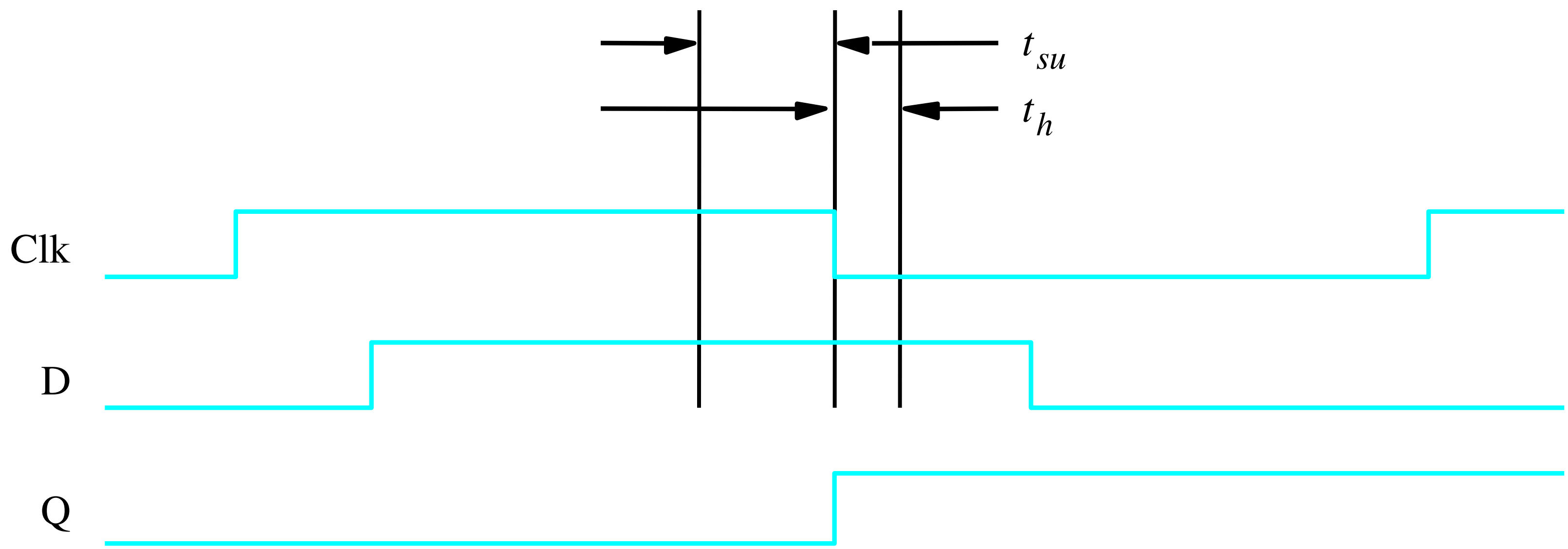
Timing Diagram for the Gated D Latch





[Figure 5.7d from the textbook]

Setup and hold times



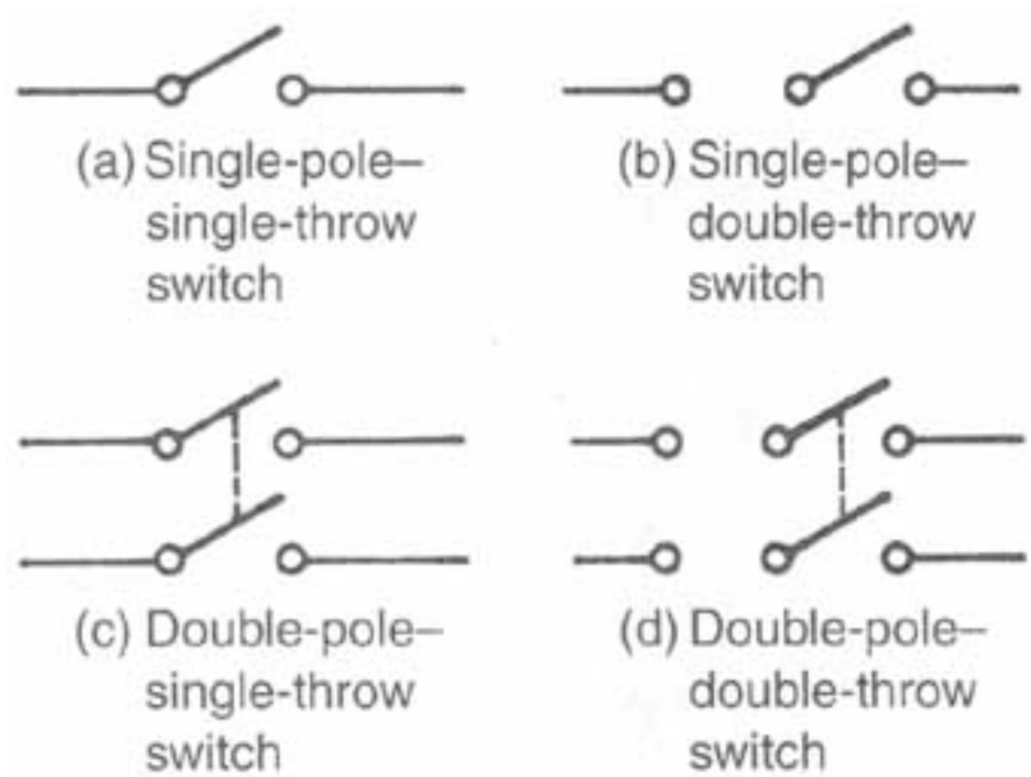
Setup time (t_{su}) – the minimum time that the D signal must be stable prior to the the negative edge of the Clock signal

Hold time (t_h) – the minimum time that the D signal must remain stable after the the negative edge of the Clock signal

[Figure 5.8 from the textbook]

Some Practical Examples

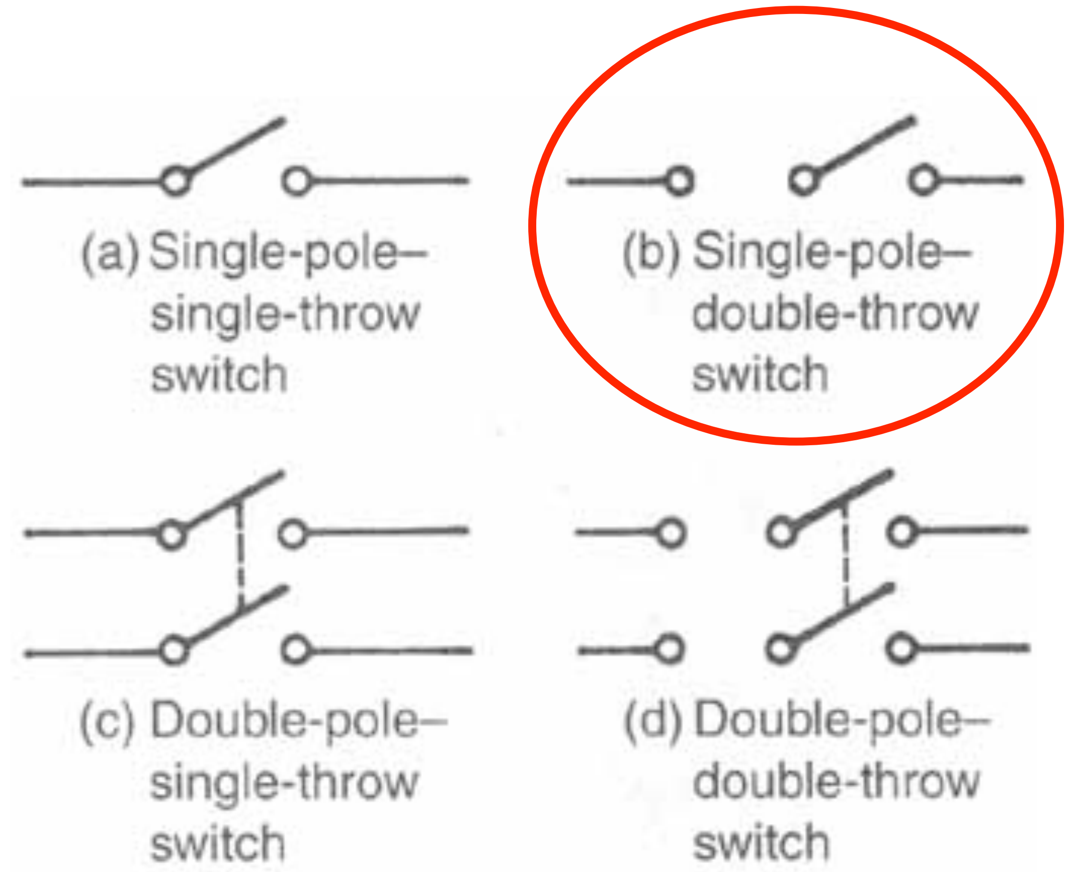
Different Types of Switches



http://www.industrial-electronics.com/Electricity-Refrigeration-Heating-Air-Conditioning_5b.html

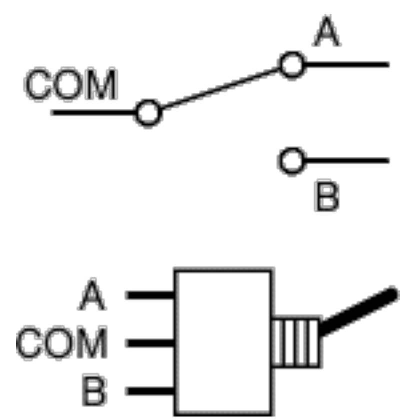
Different Types of Switches

If you are building a circuit with latches you'll need to use this type of switch.



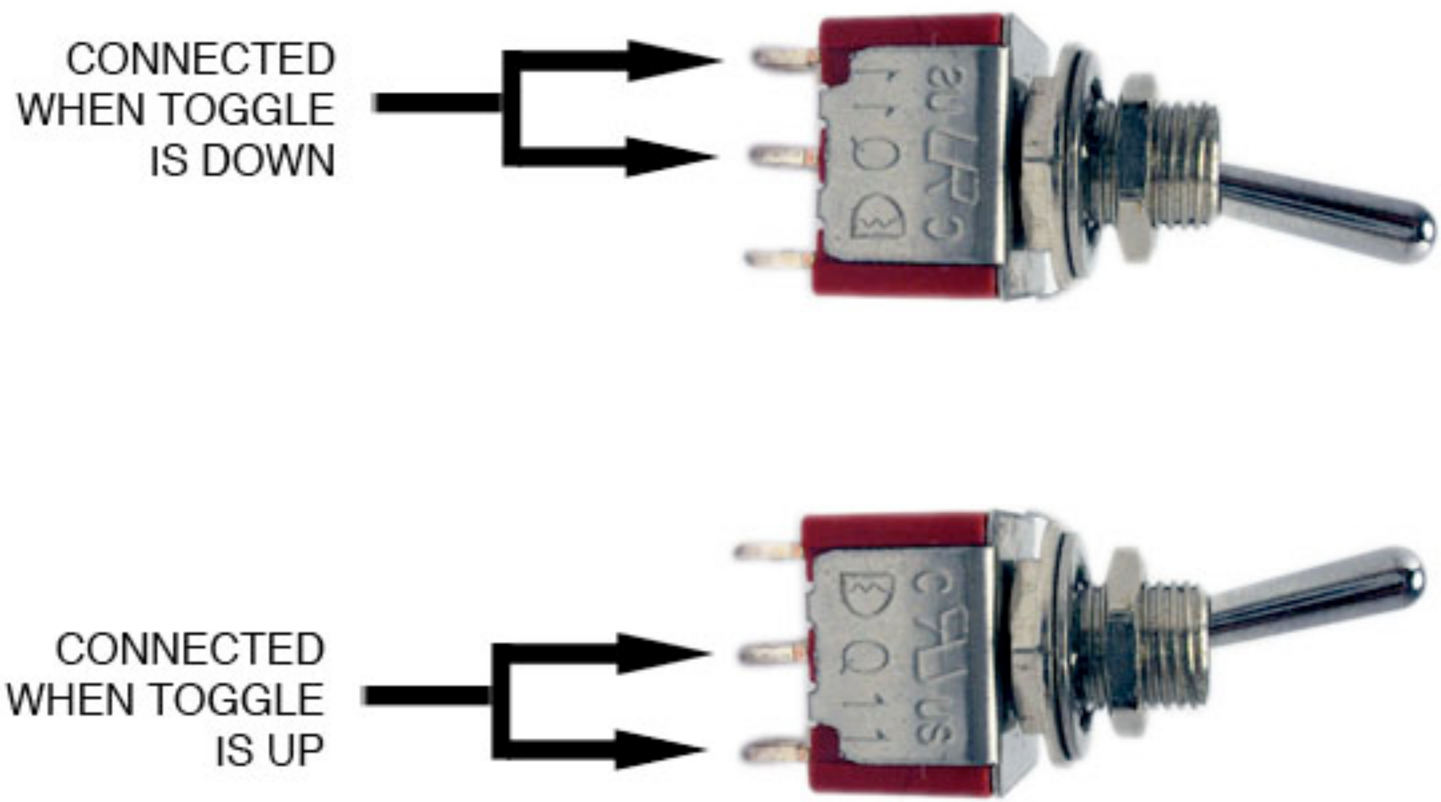
http://www.industrial-electronics.com/Electricity-Refrigeration-Heating-Air-Conditioning_5b.html

Single Pole, Double Throw = SPDT



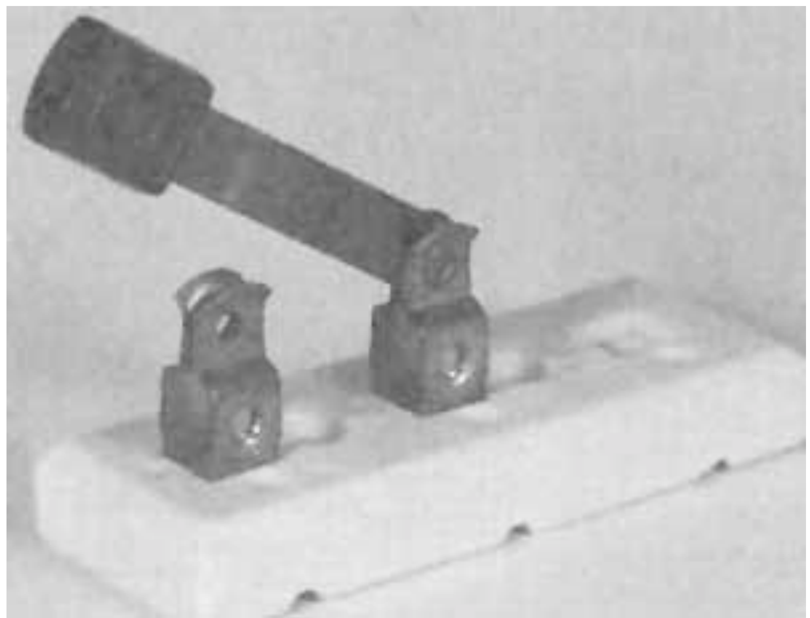
[http://electronicsclub.info/images/swabc.gif]

Single Pole, Double Throw = SPDT



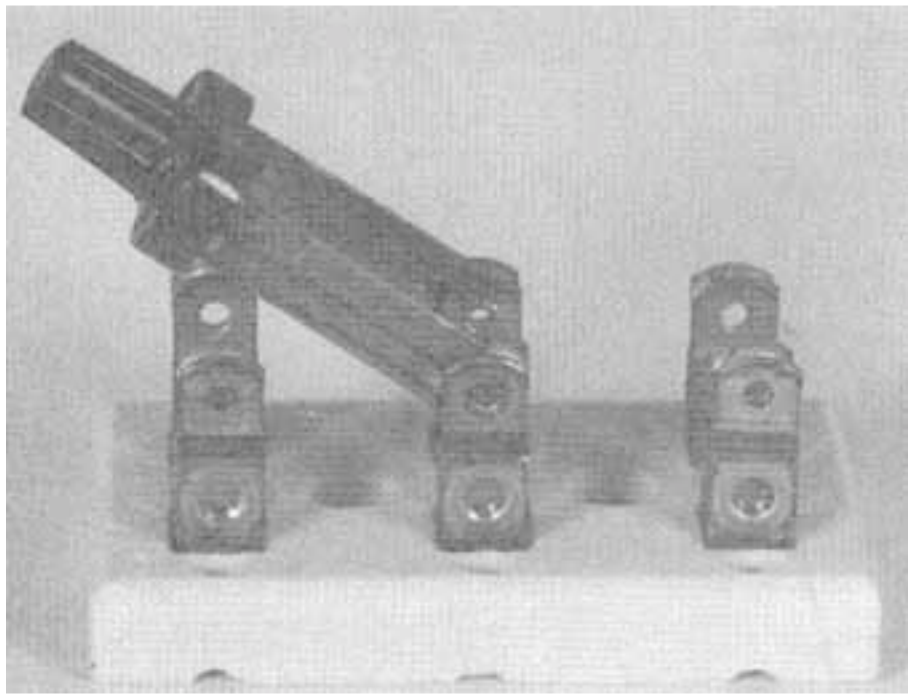
[http://www.cubisteffects.com/images/MIY/Pt3_Throw.jpg]

Single-pole—single-throw manual switch



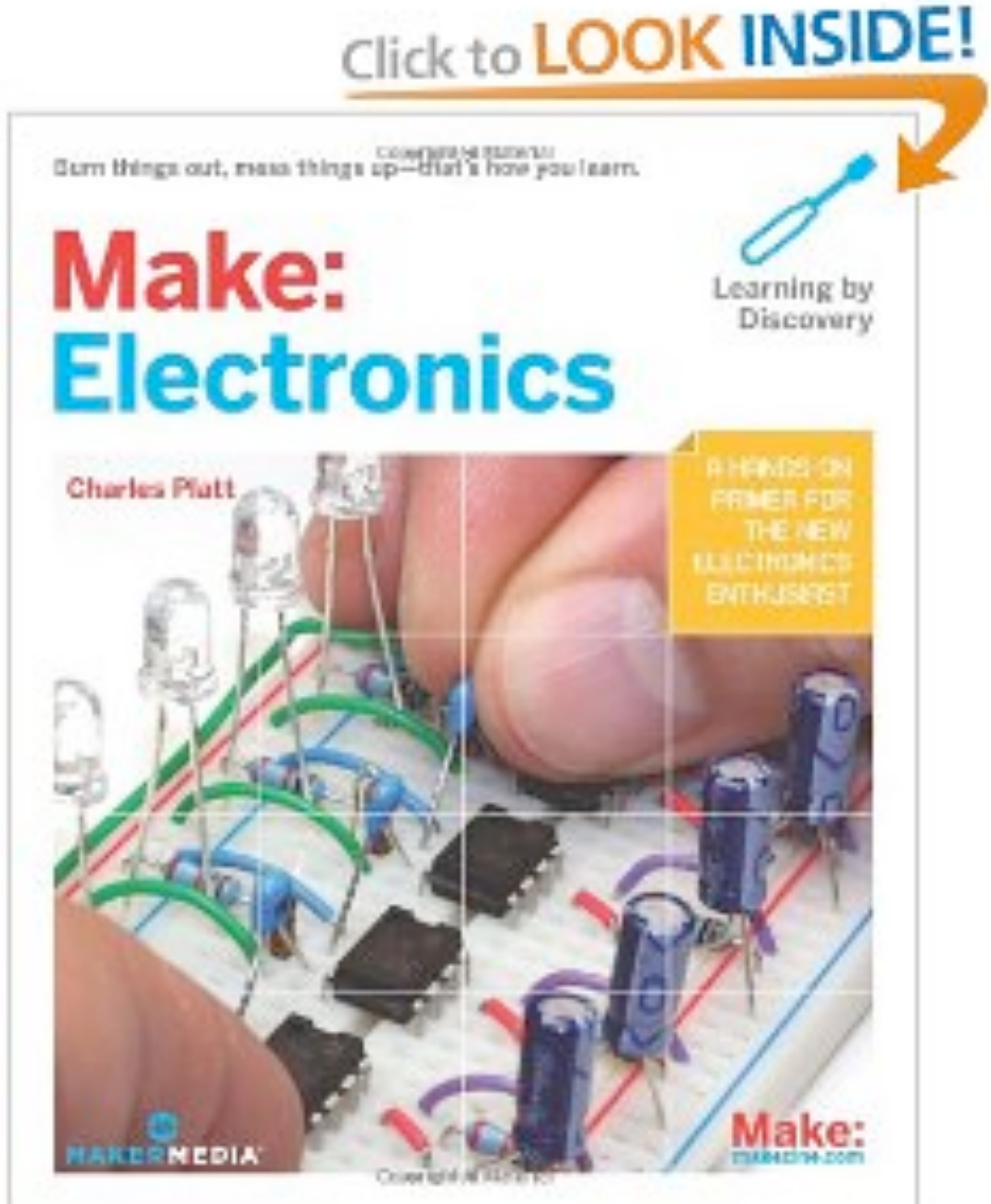
http://www.industrial-electronics.com/Electricity-Refrigeration-Heating-Air-Conditioning_5b.html

Double-pole—double-throw manual switch

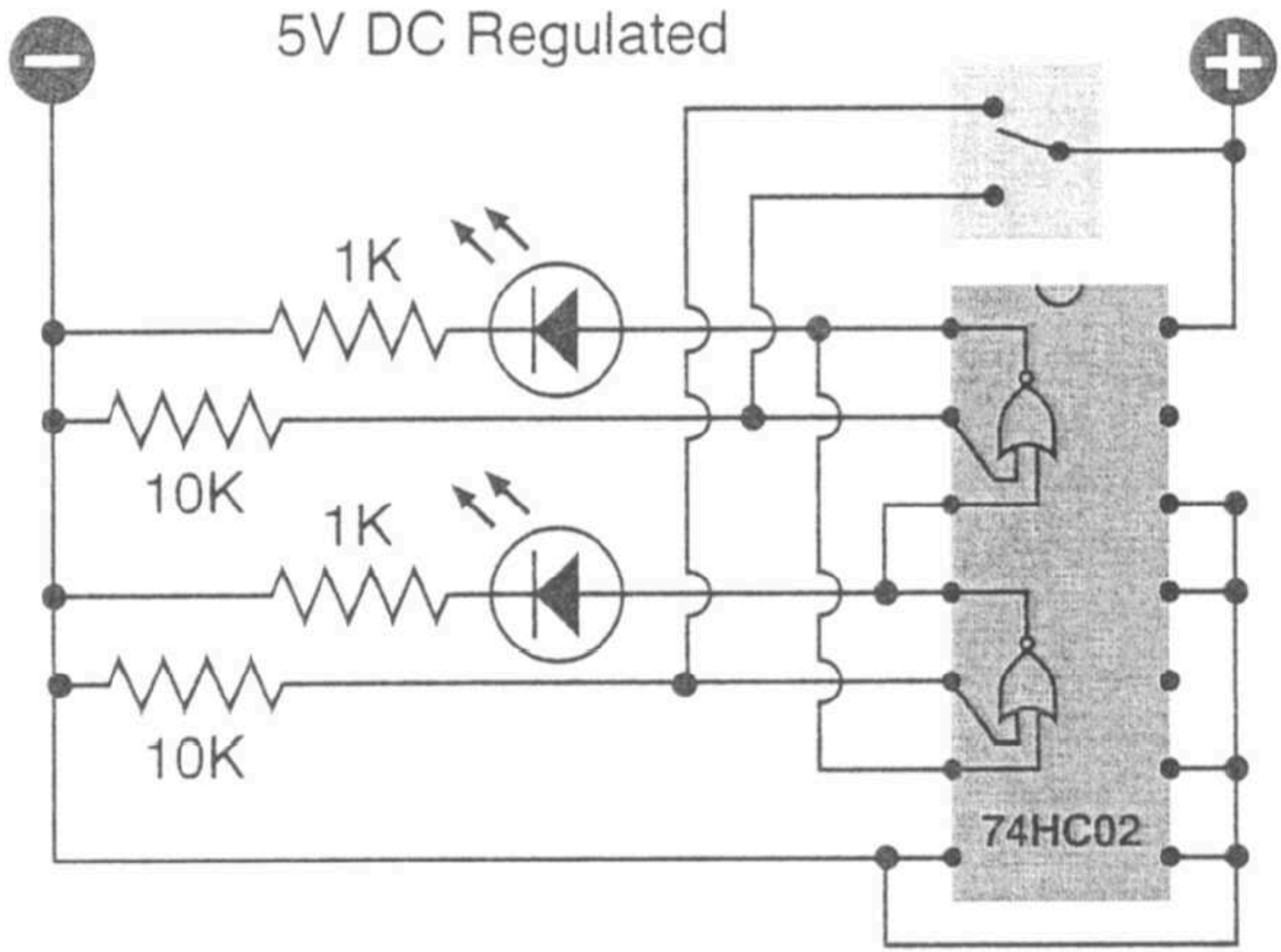


http://www.industrial-electronics.com/Electricity-Refrigeration-Heating-Air-Conditioning_5b.html

The following examples came from this book

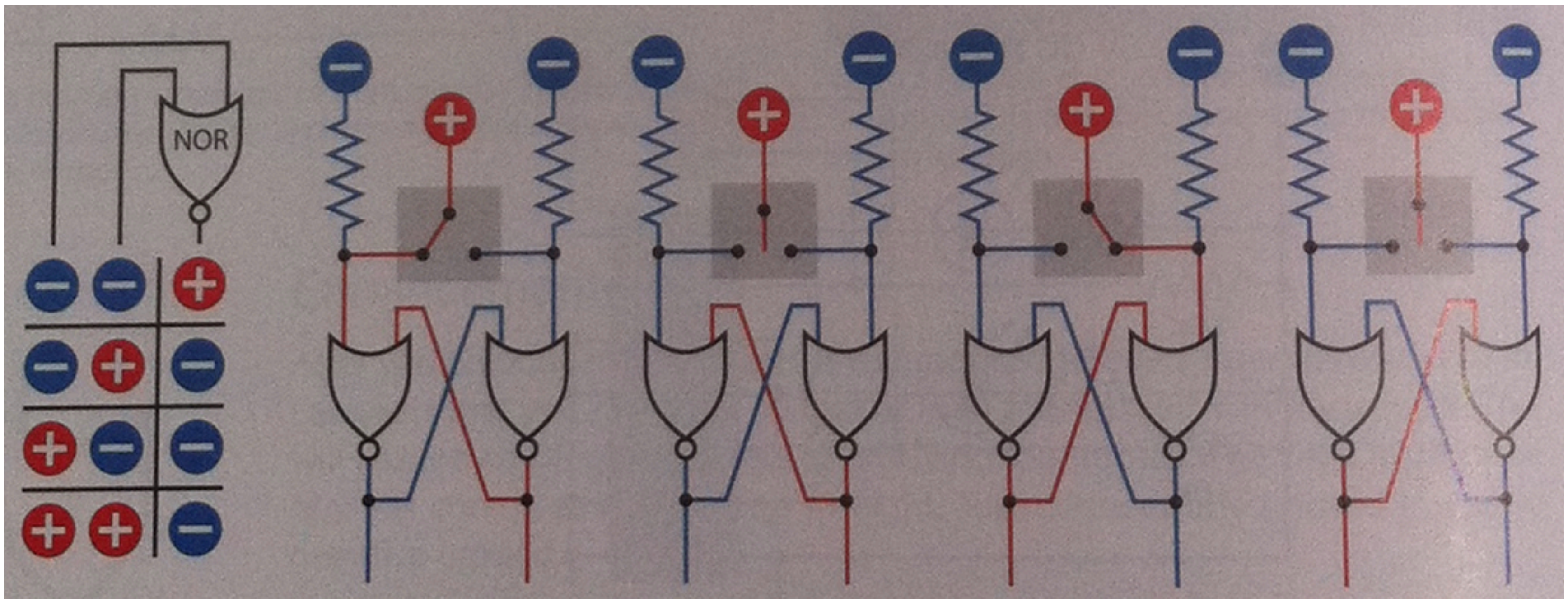


A Simple Circuit



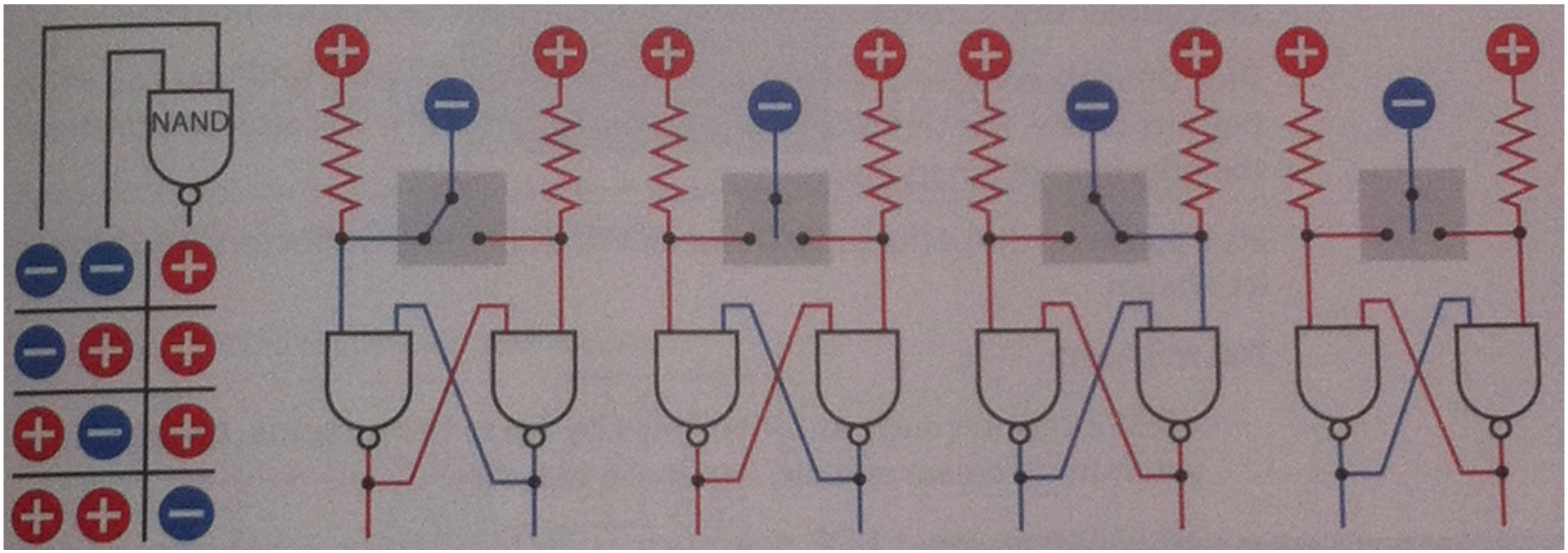
[Platt 2009]

Let’s Take a Closer Look at This



[Platt 2009]

A Similar Example with NAND Gates



[Platt 2009]

Questions?

THE END