14:332:231
DIGITAL LOGIC DESIGN

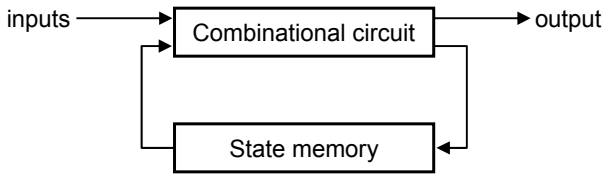
Ivan Marsic, Rutgers University
Electrical & Computer Engineering
Fall 2013

Lecture #15: Sequential Circuits: Latches

Sequential Circuits

- *Combinational circuits*: current input → output
- *Sequential circuit*: current and past inputs → output
- Sequential circuits
  - The information about the previous inputs history is called the "**state**" of the system
  - "State" is needed to predict the current and future behavior
    - *State variables* are bits of information *stored* in a memory (flip-flop) device *n* bits → 2^n states
    - A finite-state machine
  - Output depends on the current input *and* the past history represented by the states
  - Since *n* is always finite, sequential circuits are also called *finite state machines* (FSM)

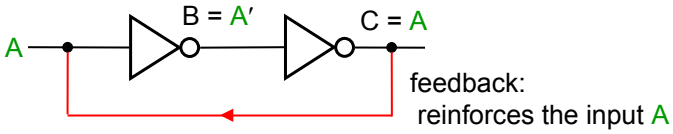
Describing Sequential Circuits

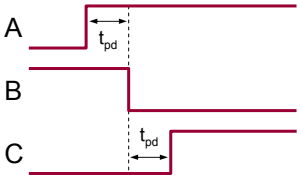


- State table
  - For each current-state, specify next-states as function of inputs and current state
- State diagram
  - Graphical version of state table

Bistable Element

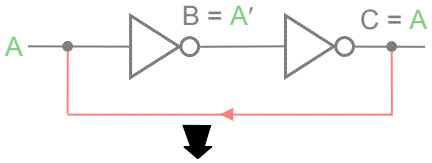
- The simplest sequential circuit
- Signal B = A′ appears after a short delay
  t_{pd} = propagation delay



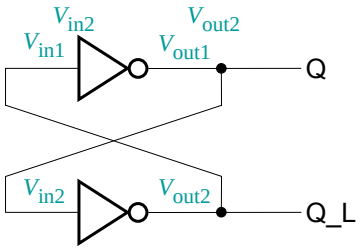


Bistable Element

- The simplest sequential circuit
- No input… for the moment
- Two states = one Boolean state variable, say, “Q”



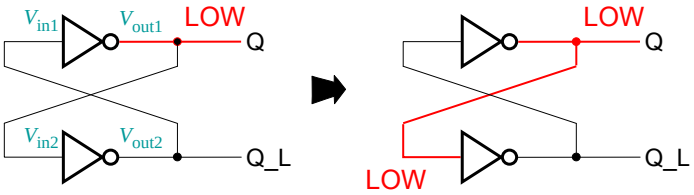
“twisted” representation:



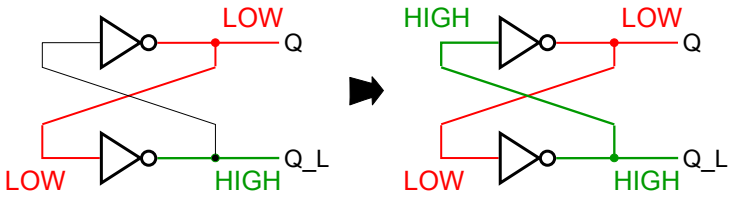
(Q_L is active low)

Bistable Element

- Assume Q is equal “0”



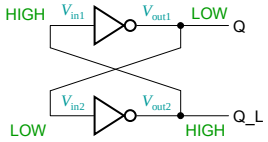
- Bottom inverter’s HIGH output → top inverter’s input



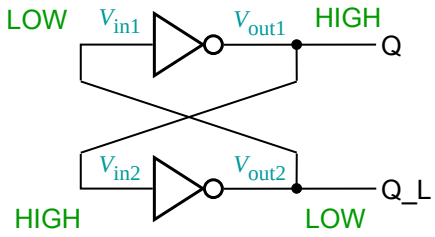
→ Top inverter’s output is forced LOW

Bistable Element

Assume Q is equal “0”



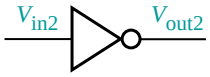
Now assume Q is equal “1”

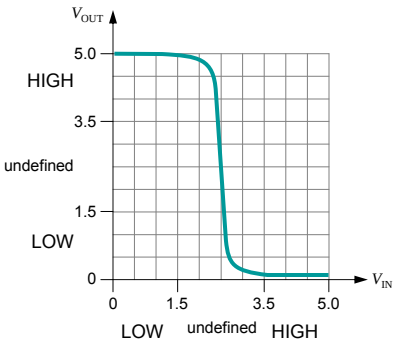


7 of 29

Analog Analysis (1)

- Assume pure CMOS thresholds, 5V is the V_{CC}
- Theoretical *threshold* center is 2.5V
- In principle, any TTL/CMOS have the same behavior, but different constants …



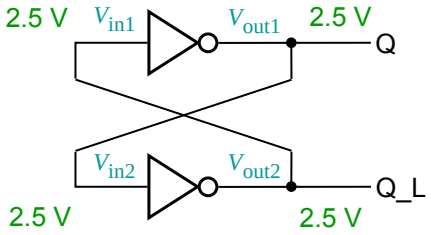


[Recall Lecture #9]

8 of 29

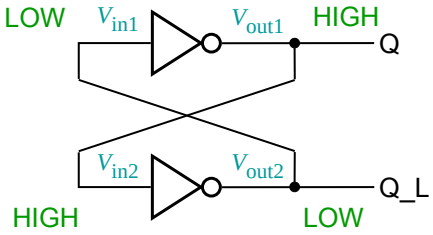
Analog Analysis (2)

Assume threshold inputs at 2.5 V:



If nothing moves …

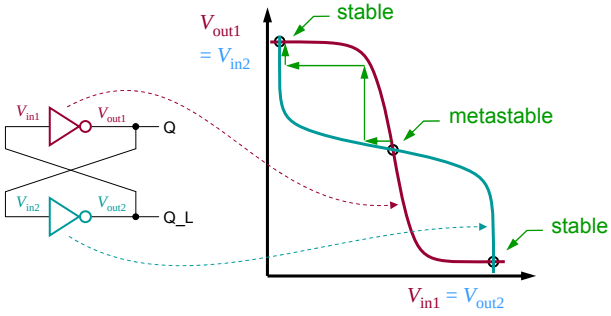
but a little input noise moves Q for example to …



… Q is equal "1"

Metastability

- Metastability is inherent in any bistable circuit



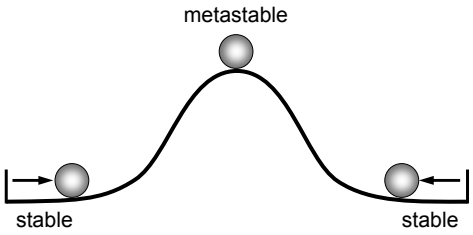
Transfer function:

$$V_{in1} = V_{out2}$$
$$= T(V_{in2})$$
$$= T(V_{out1})$$
$$= T(T(V_{in1}))$$

$$V_{in2} = T(T(V_{in2}))$$

- Two stable points, one metastable point

Another Look at Metastability



- If the ball sits *exactly* on the top of the hill, the bistable circuit can be in metastable state until random noise *nondeterministically* chooses one of the stable states.
- Can appear in *any* sequential circuit, as we will see …

Latches and Flip-Flops

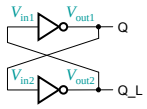
- The two most popular varieties of elements used to build sequential circuits are: latches and flip-flops

  Latch: **level sensitive** storage element

  Flip-Flop: **edge triggered** storage element

- Common examples of latches:
  S-R latch, $\overline{S}$-$\overline{R}$ latch, D latch (= gated D latch)

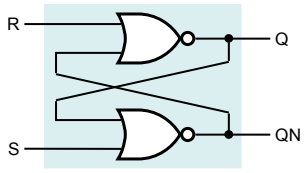
- Common examples of flip-flops (FF):
  D-FF, D-FF with enable, Scan-FF, JK-FF, T-FF



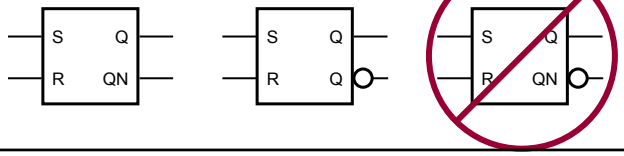
S-R Latch

X	Y	X NOR Y
0	0	1
0	1	0
1	0	0
1	1	0

- S-R (Set-Reset) latch with NOR
  - similar to inverter pair, with capability to force output to "0" (Reset=1) or "1" (Set=1)

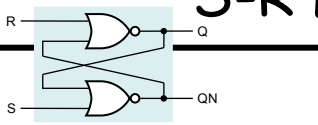


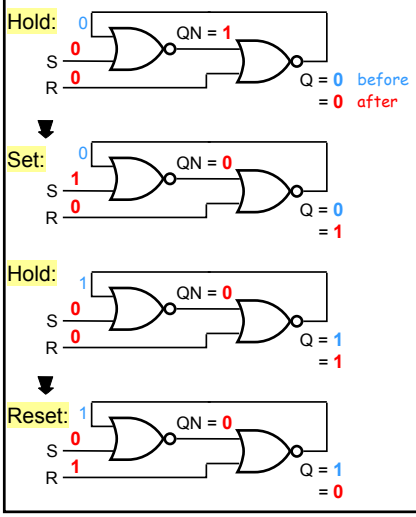
Inputs		*Outputs*		
S	**R**	**Q**	**QN**	
0	0	last Q	last QN	(hold)
0	1	0	1	(reset)
1	0	1	0	(set)
1	1	0	0	(forbidden)

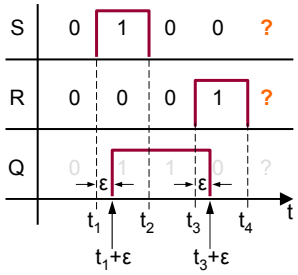


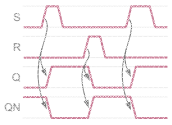
S-R Latch Operation

X	Y	X NOR Y
0	0	1
0	1	0
1	0	0
1	1	0







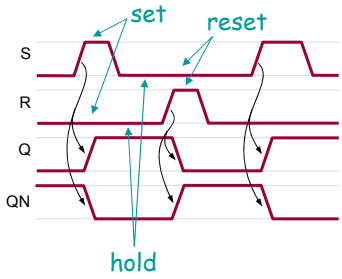


S(t)	R(t)	Q(t)	Q(t+ε)	QN(t+ε)
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	0	1
1	0	0	1	0
1	0	1	1	0
1	1	0	Still to analyze	
1	1	1		

S-R Latch Operation

- Typical operation of an S-R Latch
- (a) "normal" inputs

(a)

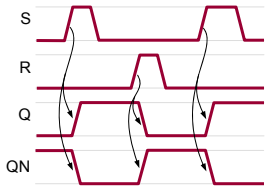


Inputs		Outputs		
S	R	Q	QN	
0	0	last Q	last QN	(hold)
0	1	0	1	(reset)
1	0	1	0	(set)
1	1	0	0	(forbidden)

S-R Latch Operation

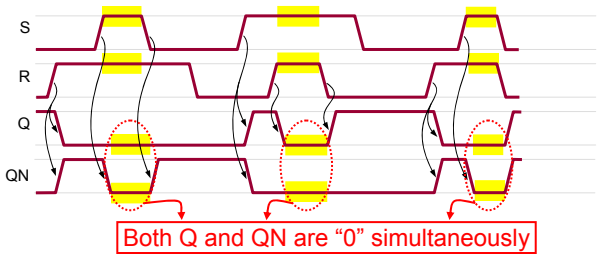
- Typical operation of an S-R Latch
- (a) "normal" inputs
- (b) S and R asserted simultaneously

(a)



Inputs		Outputs		
S	R	Q	QN	
0	0	last Q	last QN	(hold)
0	1	0	1	(reset)
1	0	1	0	(set)
1	1	0	0	(forbidden)

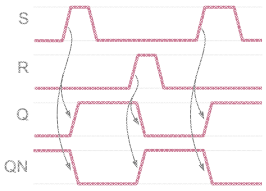
(b)



S-R Latch Operation

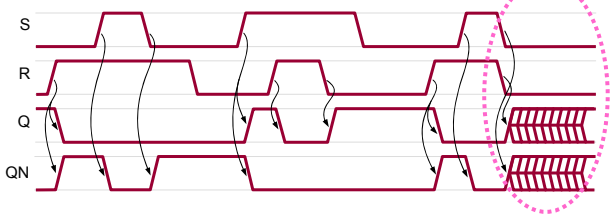
- Typical operation of an S-R Latch
- (a) "normal" inputs (b) S and R asserted simultaneously

(a)



Metastability is possible if S and R are *negated* simultaneously

(b)



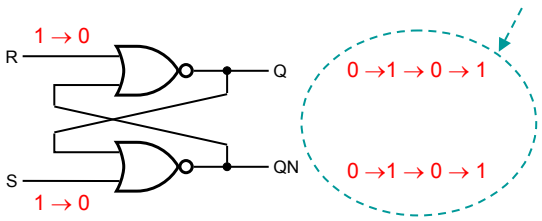
"race condition"

say, less than 20ns

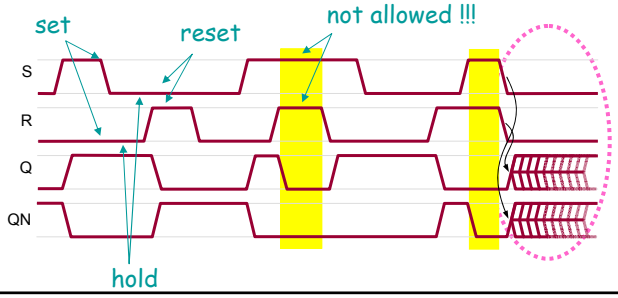
17 of 29

Improper S-R Latch Operation

- Metastability may occur if S and R are negated simultaneously: the circuit starts to oscillate



(b)

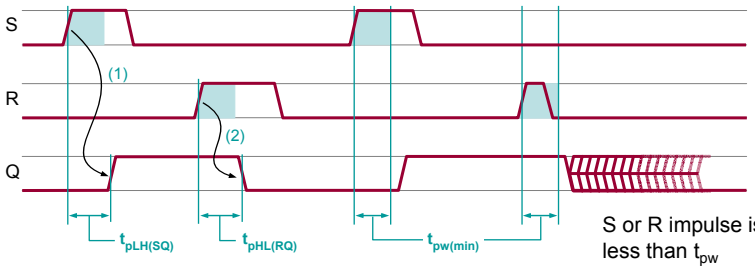


"race condition"

18 of 29

S-R Latch Timing Parameters

- **Propagation delay** (t_p) for an input transition to produce an output
- **Minimum pulse width** (t_{pw}) needed for deterministic transitions

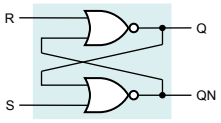


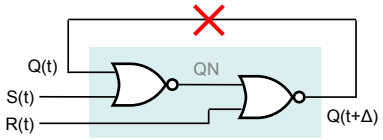
S or R impulse is less than t_{pw} also leads to metastability

- **Recovery time** (t_{rec}) = minimum delay between negating S and R for them *not* to be considered simultaneous
- t_{rec} and t_{pw} are related: both measure how long it takes for the latch feedback loop to stabilize
- Violations of t_{pw} and t_{rec} cause *metastability*

R-S Latch Analysis

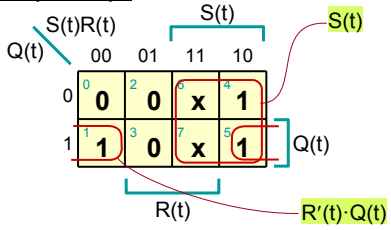
- Break the feedback path





Inputs			*Output*	
S(t)	R(t)	Q(t)	Q(t+Δ)	
0	0	0	0	(hold)
0	0	1	1	
0	1	0	0	(reset)
0	1	1	0	
1	0	0	1	(set)
1	0	1	1	
1	1	0	x	(not allowed)
1	1	1	x	

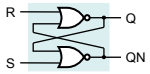
Q(t+Δ):



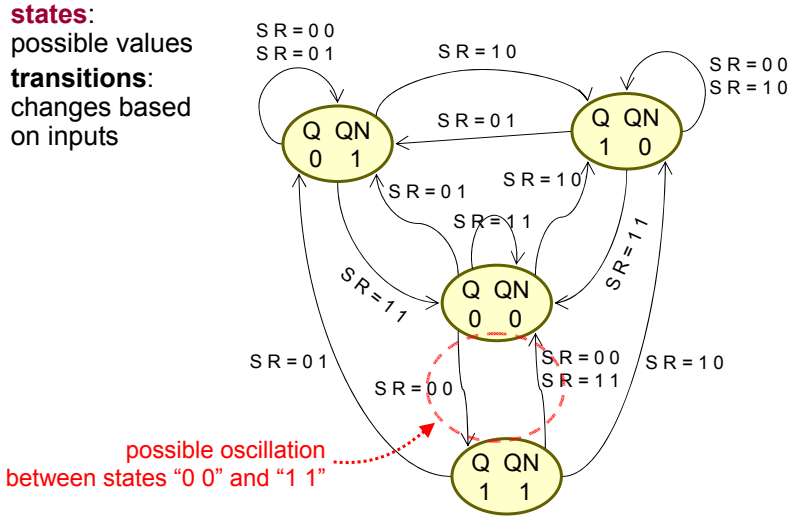
- Next state equation, a.k.a. *characteristic equation*:

$Q(t+\Delta) = S(t) + R'(t)\cdot Q(t) \quad \Leftrightarrow \quad Q^+ = Q^* = S + R'\cdot Q$

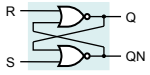
Theoretical R-S Latch Behavior



- State diagram
  - **states**: possible values
  - **transitions**: changes based on inputs

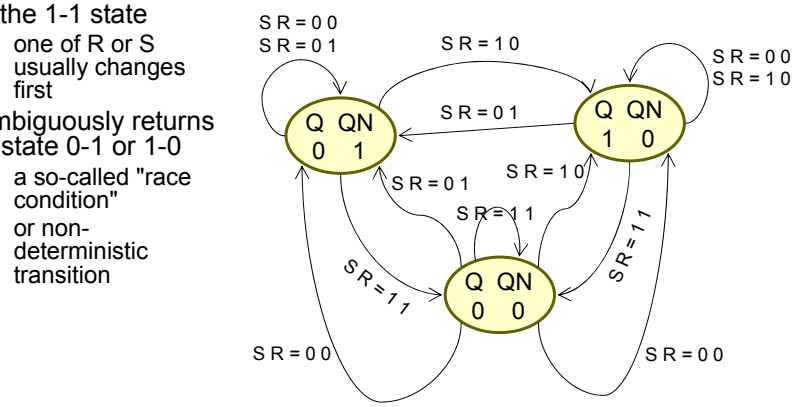


Observed R-S Latch Behavior



- Very difficult to observe R-S latch in the 1-1 state
  - one of R or S usually changes first
- Ambiguously returns to state 0-1 or 1-0
  - a so-called "race condition"
  - or non-deterministic transition

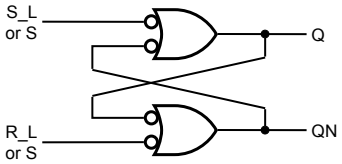
metastability

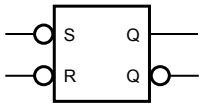


$\bar{S}$-$\bar{R}$ Latch using NAND gates

- Used more than S-R latch with NOR

… because NAND gates are preferred over NOR gates





- Timing and metastability similar as for S-R

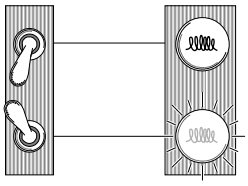
Inputs		*Outputs*	
S_L	**R_L**	**Q**	**QN**
0	0	1	1
0	1	1	0
1	0	0	1
1	1	last Q	last QN

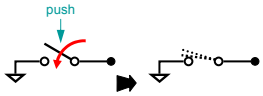
- Next state equation (characteristic equation):

$Q(t+\Delta) = S'(t) + R(t)\cdot Q(t) \quad \Leftrightarrow \quad Q^{+} = Q^{*} = S' + R\cdot Q$

Bistable Application: Switch Debouncing

- Mitigating oscillations in mechanical switches when the wiper makes contact with the terminal



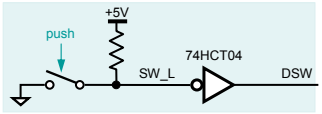


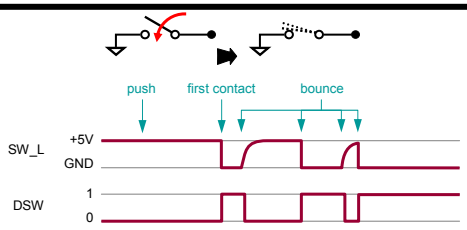
- Problem if the switch is used for counting the number of pushes

Wakerly, Section 8.2.3

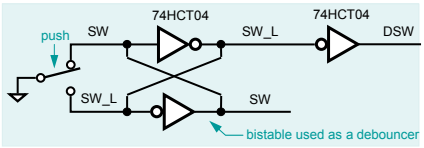
Switch Debouncer [Wakerly, Section 8.2.3]

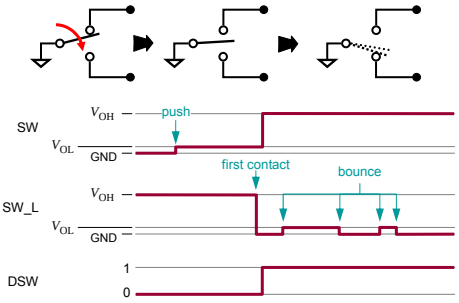
- No debouncing:





- Using a bistable for debouncing:



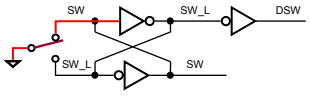


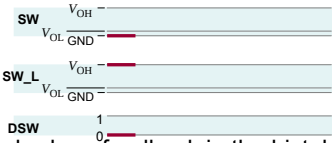
- SW trying to maintain 0 when the push starts but the first contact with SW_L moves it to 1. After say 30 ns, the switch changes, DSW=1

... see the next slide for details

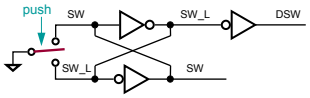
Using a Bistable for Debouncing

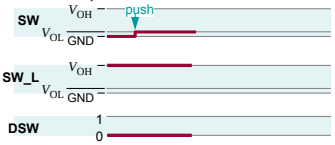
- Before the button is pushed, the top contact holds SW at 0 V and top inverter produces "1" on SW_L



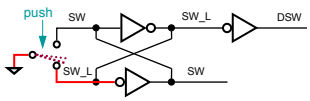


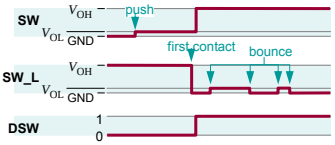
- When the button is pushed and contact is broken, feedback in the bistable holds SW at V_{OL}, i.e., logic "0"





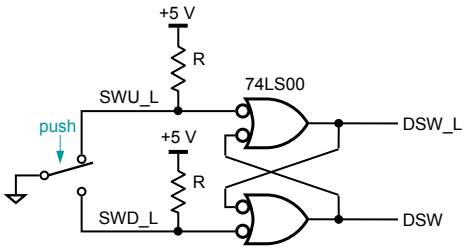
- When the wiper hits the bottom contact, SW_L is forced to logic "0" and the output of the top inverter also becomes "0"
- After this, feedback in the bistable maintains the logic "0" on SW even if the wiper bounces off the bottom contact





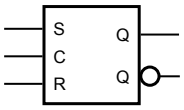
Using an S-R Latch for Debouncing

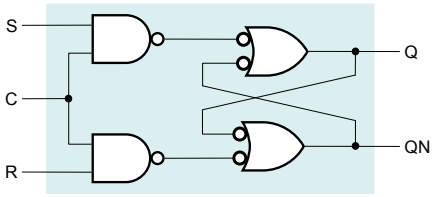
- The pull-up resistors avoid the momentarily *shorting* of the gate outputs,
  when the output of the top inverter is still HIGH
  and the SW_L becomes connected to the ground



S-R Latch with Enable

- Sensitive to S/R inputs only when an enabling input **C** is asserted ("C" stands for "clock")

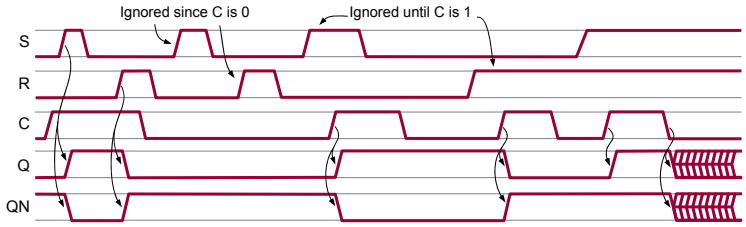




Inputs			*Outputs*	
S	**R**	**C**	**Q**	**QN**
0	0	1	last Q	last QN
0	1	1	0	1
1	0	1	1	0
1	1	1	1	1
x	x	0	last Q	last QN

S-R Latch with Enable

- Typical operation
- If both S and R are “1” when the enabling input C is turned from “1” to “0”, the circuit behaves like an S-R latch and the output can become metastable



29 of 29