Module 4:FLIP-FLOP

Quote of the day

"Never think you are nothing, never think you are everything, but think you are something and achieve anything".

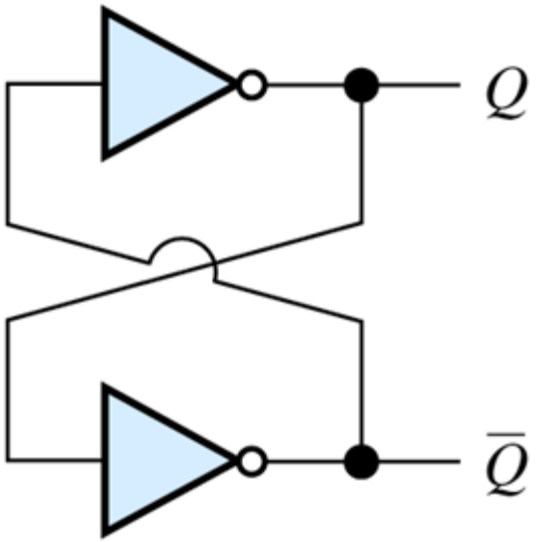
— Albert Einstein

Sequential and combinational circuits

- A digital circuit whose output depends on only applied input is combinational circuit.
- Digital circuits studied until now are combinational circuits
- A digital circuit whose output not only depends on applied input but also depends on the present state of output is Sequential circuit.
- Sequential circuit has memory.
- Flip-flop is a basic element of sequential circuit.

FLIP-FLOP

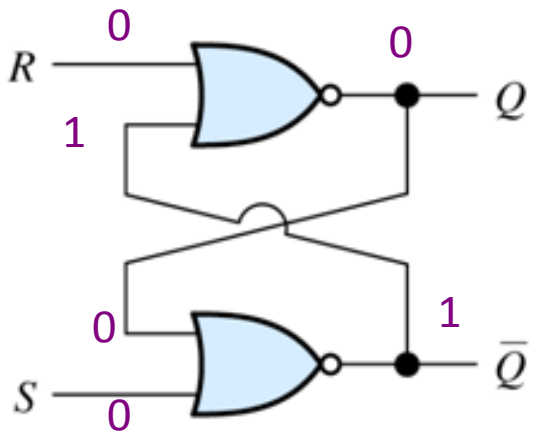
- A basic sequential circuit is a flip-flop
- Flip-flop has two stable states of complementary output values



Simple flip-flop.

NOR Gate Latch\SR Flip-flop

- SR (set-reset) flip-flop based on two nor gates.
- If R=0,S=0 then output of both NOR gates depend on other inputs ,i.e Q and $\bar{Q}$
- Assume Q=0 and $\bar{Q}$=1 initially
- Observe that out put will not change

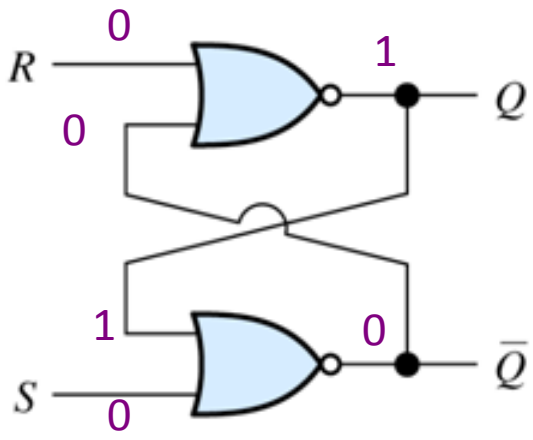


An SR flip-flop can be implemented by cross coupling two NOR gates.

SR-FLIP-FLOP			
R	S	Q	$\bar{Q}$
0	0		
0	1		
1	0		
1	1		

NOR Gate Latch\SR Flip-flop

- SR (set-reset) flip-flop based on two nor gates.
- If R=0,S=0 then output of both NOR gates depend on other inputs ,i.e Q and $\bar{Q}$
- Assume Q=1 and $\bar{Q}$=0 initially
- Observe that out put will not change

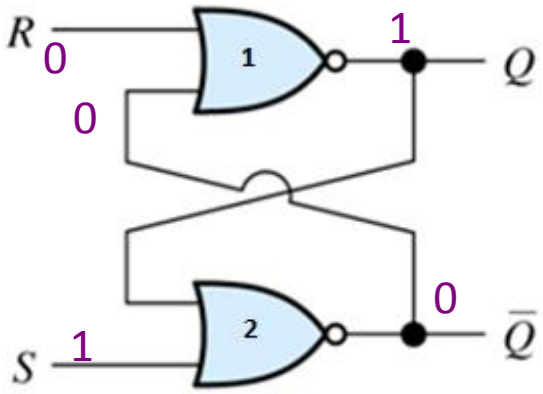


An SR flip-flop can be implemented by cross coupling two NOR gates.

SR-FLIP-FLOP			
R	S	Q	$\bar{Q}$
0	0	Last state	
0	1		
1	0		
1	1		

NOR Gate Latch\SR Flip-flop

- SR (set-reset) flip-flop based on two nor gates.
- If R=0,S=1 then output of Gate 2 becomes one, i.e. $\bar{Q}$ = 0 and output of gate 1 which depends on $\bar{Q}$ becomes 1.
- Observe that out put set to one whatever may be previous state of O/P.

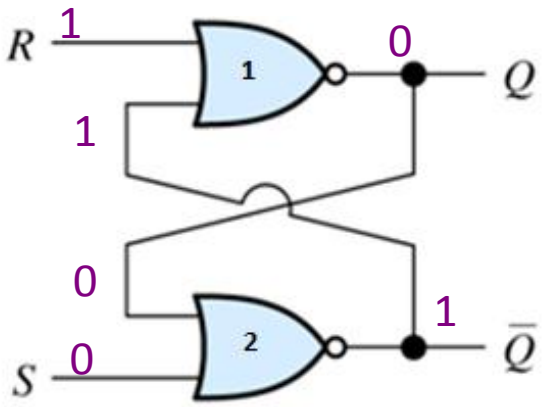


An SR flip-flop can be implemented by cross coupling two NOR gates.

SR-FLIP-FLOP			
R	S	Q	$\bar{Q}$
0	0	Last state	
0	1	1	0
1	0		
1	1		

NOR Gate Latch\SR Flip-flop

- SR (set-reset) flip-flop based on two nor gates.
- If R=1,S=0 then output of Gate 1 becomes zero, i.e Q= 0 and output of gate 2 which depends on Q becomes 1.
- Observe that out put reset to zero whatever may be previous state of O/P.

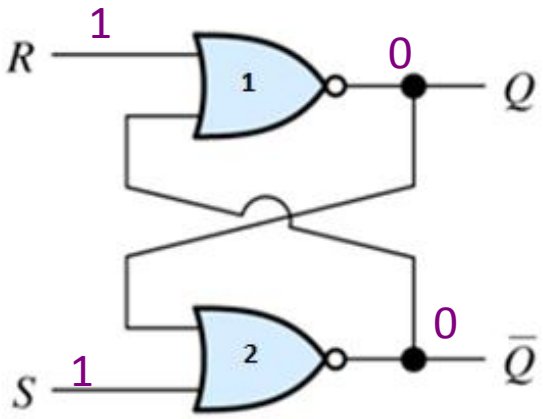


An SR flip-flop can be implemented by cross coupling two NOR gates.

SR-FLIP-FLOP			
R	S	Q	$\bar{Q}$
0	0	Last state	
0	1	1	0
1	0	0	1
1	1		

NOR Gate Latch\SR Flip-flop

- SR (set-reset) flip-flop based on two nor gates.
- If R=1,S=1 then output of both Gates becomes zero, i.e. Q= 0 and $\bar{Q}$=0.
- This is not allowed state of Flip Flop.
- Is Illegal state or Forbidden state of Flip-flop

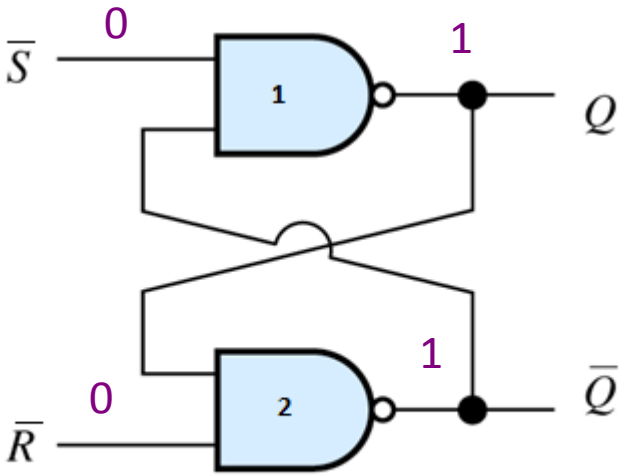


An SR flip-flop can be implemented by cross coupling two NOR gates.

SR-FLIP-FLOP			
R	S	Q	$\bar{Q}$
0	0	Last state	
0	1	1	0
1	0	0	1
1	1	Illegal	

NAND Gate Latch\S̄R̄ Flip-flop

- S̄R̄ (set-reset) flip-flop based on two nor gates.
- If R̄=0,S̄=0 then output of both Gates becomes one, i.e. Q= 1 and Q̄=1.
- This is not allowed state of Flip Flop.
- Is Illegal state or Forbidden state of Flip-flop



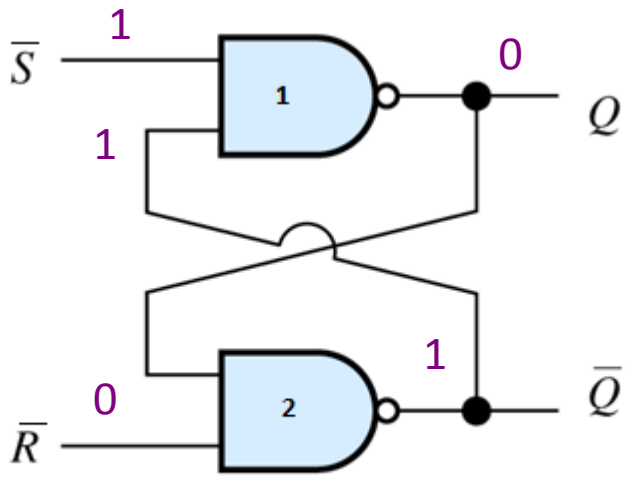
A flip-flop implemented with NAND gates.

S̄R̄-FLIP-FLOP			
R̄	S̄	Q	Q̄
0	0	Illegal	
0	1		
1	0		
1	1		

NAND Gate Latch\S̄R̄ Flip-flop

S̄R̄ (set-reset) flip-flop based on two nor gates.

- If R̄=0,S̄=1 then output of Gate-2 becomes one, i.e Q̄=1 and output of gate 1 which depends on Q̄ becomes 0.
- Observe that out put reset to zero whatever may be previous state of O/P.

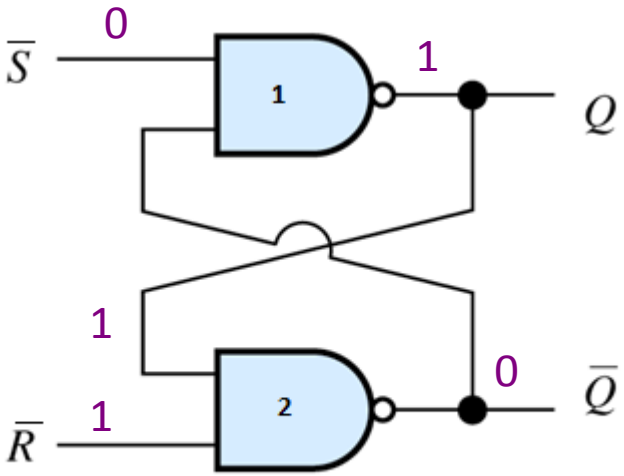


A flip-flop implemented with NAND gates.

S̄R̄-FLIP-FLOP			
R̄	S̄	Q	Q̄
0	0	Illegal	
0	1	0	1
1	0		
1	1		

NAND Gate Latch\S̄R̄ Flip-flop

- S̄R̄ (set-reset) flip-flop based on two nor gates.
- If R̄=1,S̄=0 then output of Gate-1 becomes one, i.e Q=1 and output of gate 2 which depends on Q becomes 0.
- Observe that out put set to one whatever may be previous state of O/P.

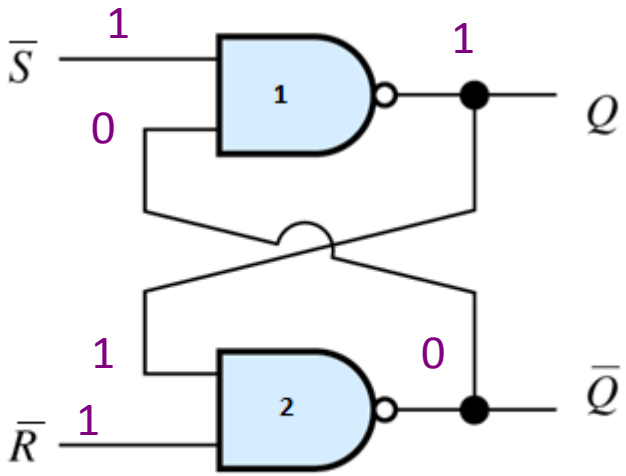


A flip-flop implemented with NAND gates.

S̄R̄-FLIP-FLOP			
R̄	S̄	Q	Q̄
0	0	Illegal	
0	1	0	1
1	0	1	0
1	1		

NAND Gate Latch\S̄R̄ Flip-flop

- S̄R̄ (set-reset) flip-flop based on two nor gates.
- If R̄=1,S̄=1 then output of both NOR gates depend on other inputs ,i.e Q and Q̄ .
- Assume Q=1 and Q̄=0 initially.
- Observe that out put will not change.

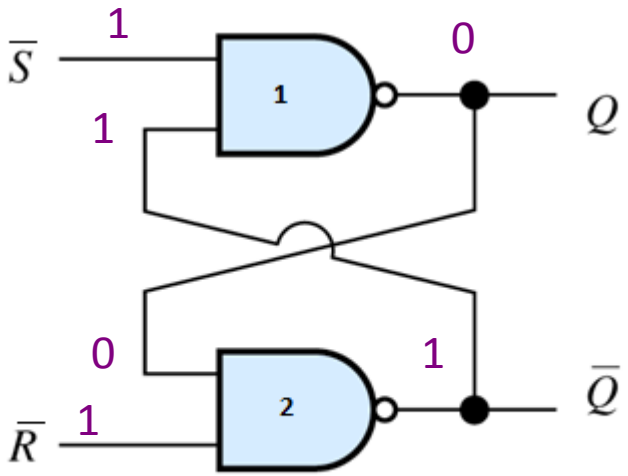


A flip-flop implemented with NAND gates.

S̄R̄-FLIP-FLOP			
R̄	S̄	Q	Q̄
0	0	Illegal	
0	1	0	1
1	0	1	0
1	1		

NAND Gate Latch\S̄R̄ Flip-flop

- S̄R̄ (set-reset) flip-flop based on two nor gates.
- If R̄=1,S̄=1 then output of both NOR gates depend on other inputs ,i.e Q and Q̄ .
- Assume Q=0 and Q̄=1 initially.
- Observe that out put will not change.



A flip-flop implemented with NAND gates.

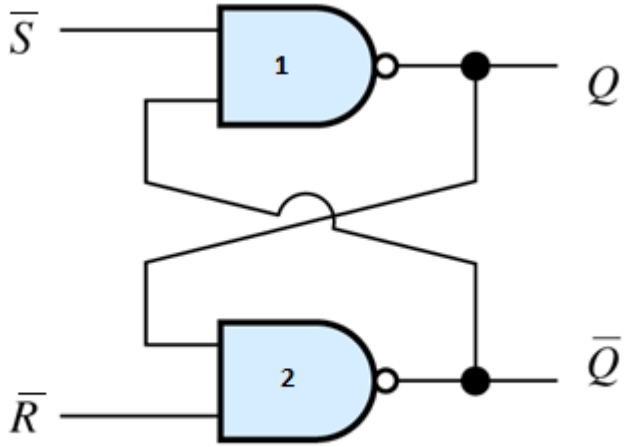
SR-FLIP-FLOP			
R̄	S̄	Q	Q̄
0	0	Illegal	
0	1	0	1
1	0	1	0
1	1	Last state	

Truth table					
$\bar{R}$	$\bar{S}$	R	S	Q	$\bar{Q}$
0	0	1	1	**Illegal**	
0	1	1	0	**0**	**1**
1	0	0	1	**1**	**0**
1	1	0	0	**Last state**	

SR-FLIP-FLOP				
R	S	Q	$\bar{Q}$	Output(Q)
0	0	**Last state**		**Last state**
0	1	**1**	**0**	**Set**
1	0	**0**	**1**	**Reset**
1	1	**Illegal**		**Illegal**

Flip-Flop

- The previous slides described you the NAND/NOR Gate Latch.
- You observed that a latch using NAND is referred as $\bar{S}\bar{R}$ latch.

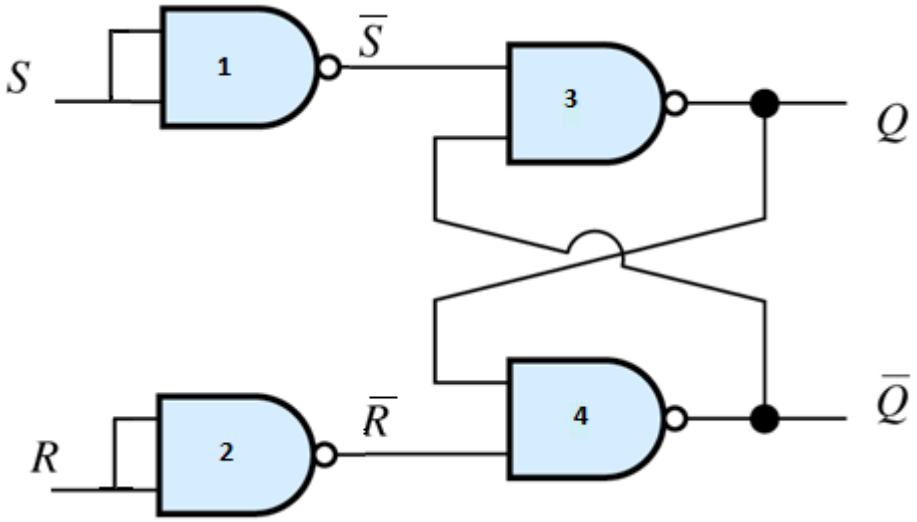


A flip-flop implemented with NAND gates.

SR-FLIP-FLOP			
$\bar{R}$	$\bar{S}$	Q	$\bar{Q}$
0	0	Illegal	
0	1	0	1
1	0	1	0
1	1	Last states	

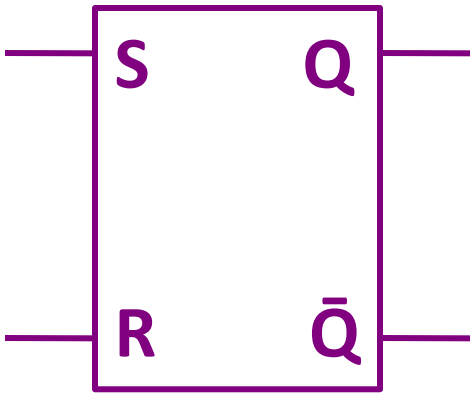
- To obtain SR latch using NAND gates, we are suppose to add two NOT gates implemented using NAND gates at the terminal of these inputs $\bar{S}\bar{R}$.

SR latch using NAND gates Flip-flop



Truth table					
R	S	$\bar{R}$	$\bar{S}$	Q^+	$\bar{Q}^+$
0	0	1	1	Last state	
0	1	1	0	1	0
1	0	0	1	0	1
1	1	0	0	Illegal	

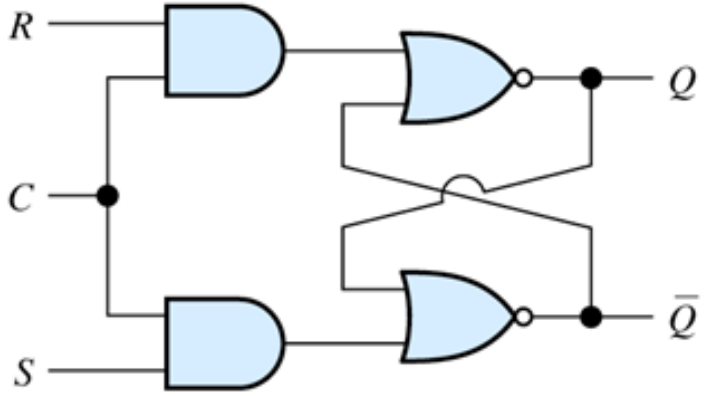
Logical Symbol



Gated Flip-Flops

- In the latches observed till now output responds immediately to the changes in input.
- In many digital systems it is required that the circuit responds only at some prescribed time, decided by another input called the enable or clock input.
- A Flip-Flop which immediately responds to the changes of input in spite of enable or clock input is referred as latch.
- Some flip-flops responds to the changes of input during the edges of the clock inputs.

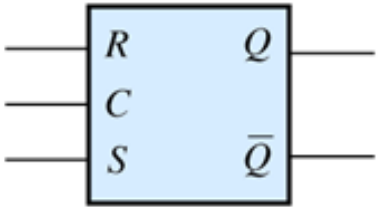
Gated Flip-Flops



(a) Circuit diagram

R	S	C	Q_n
0	0	×	Q_{n-1}
0	1	1	1
1	0	1	0
1	1	1	Not allowed
×	×	0	Q_{n-1}

(b) Truth table

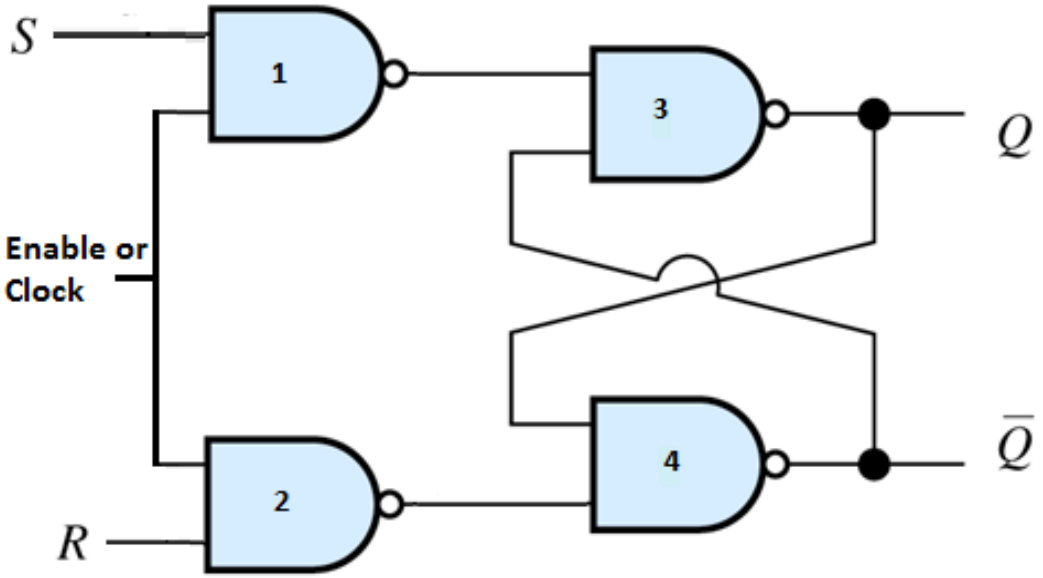


(b) Circuit symbol

A clocked SR flip-flop.

- **The latch will disable with C=0 and continue the last state. The output is said to be latched.**
- **It will enable with C=1 and function as normal SR F/F.**
- **Q_n is present state of output, Q_{n-1} last state of output state**

Gated Latch-Clocked RS Flip-flop

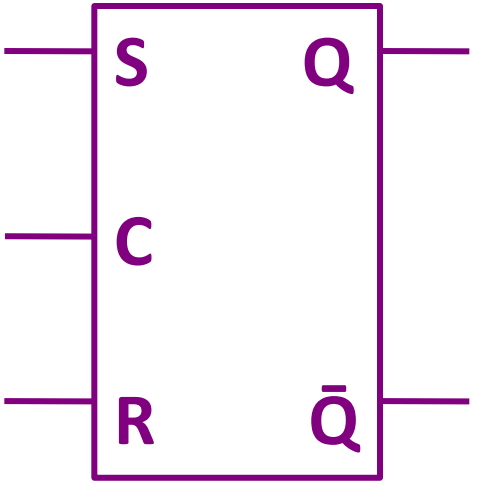


A clocked SR flip-flop.

R	S	Enable	Q_n
0	0	×	Q_{n-1}
0	1	1	1
1	0	1	0
1	1	1	Not allowed
×	×	0	Q_{n-1}

Truth table

Logical Symbol



Input output waveform