

XMUT 202

Digital Electronics

Felix Yan

School of Engineering and Computer Science
Victoria University of Wellington

Victoria

UNIVERSITY OF WELLINGTON

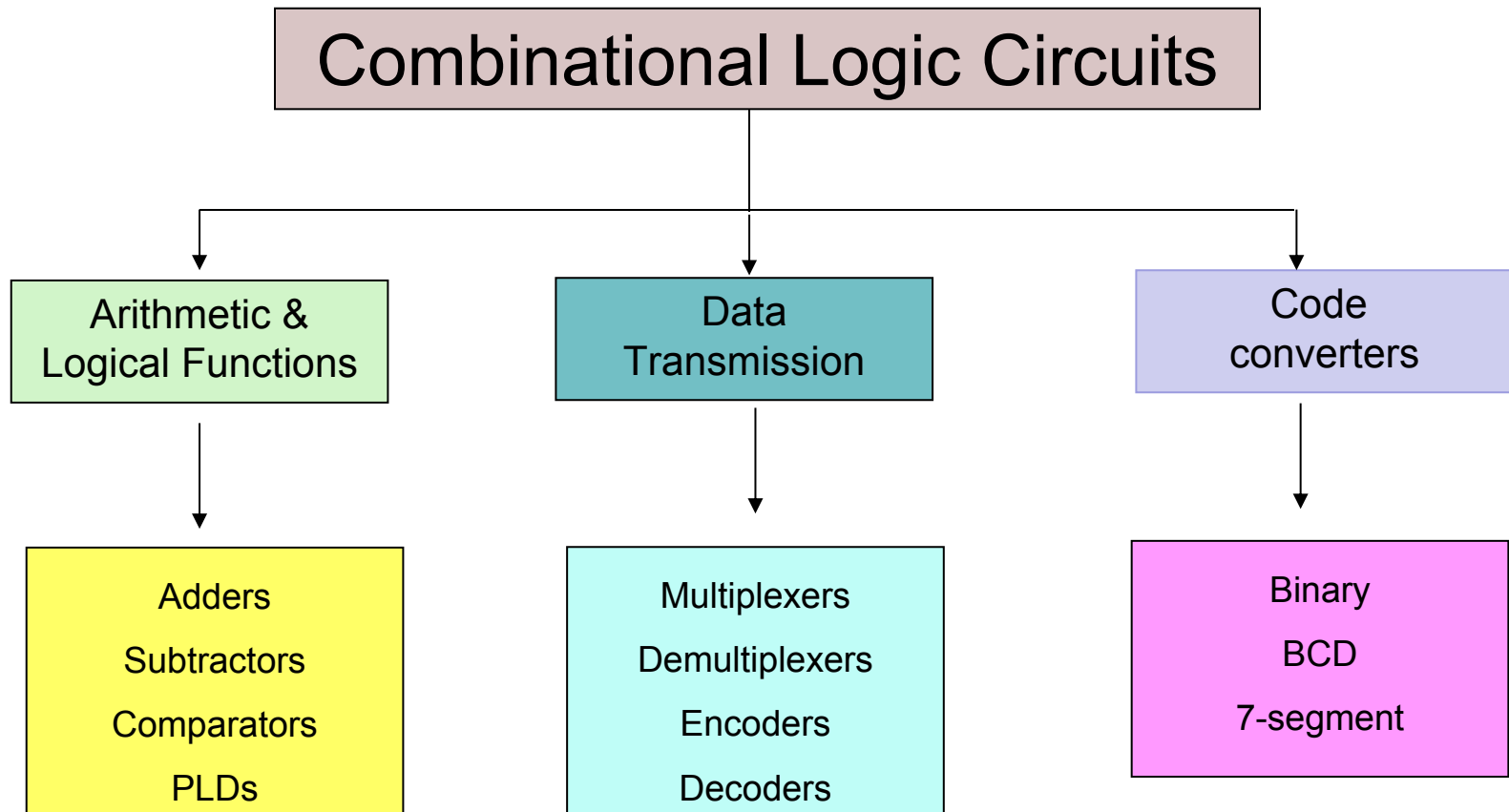
*Te Whare Wānanga
o te Ūpoko o te Ika a Māui*



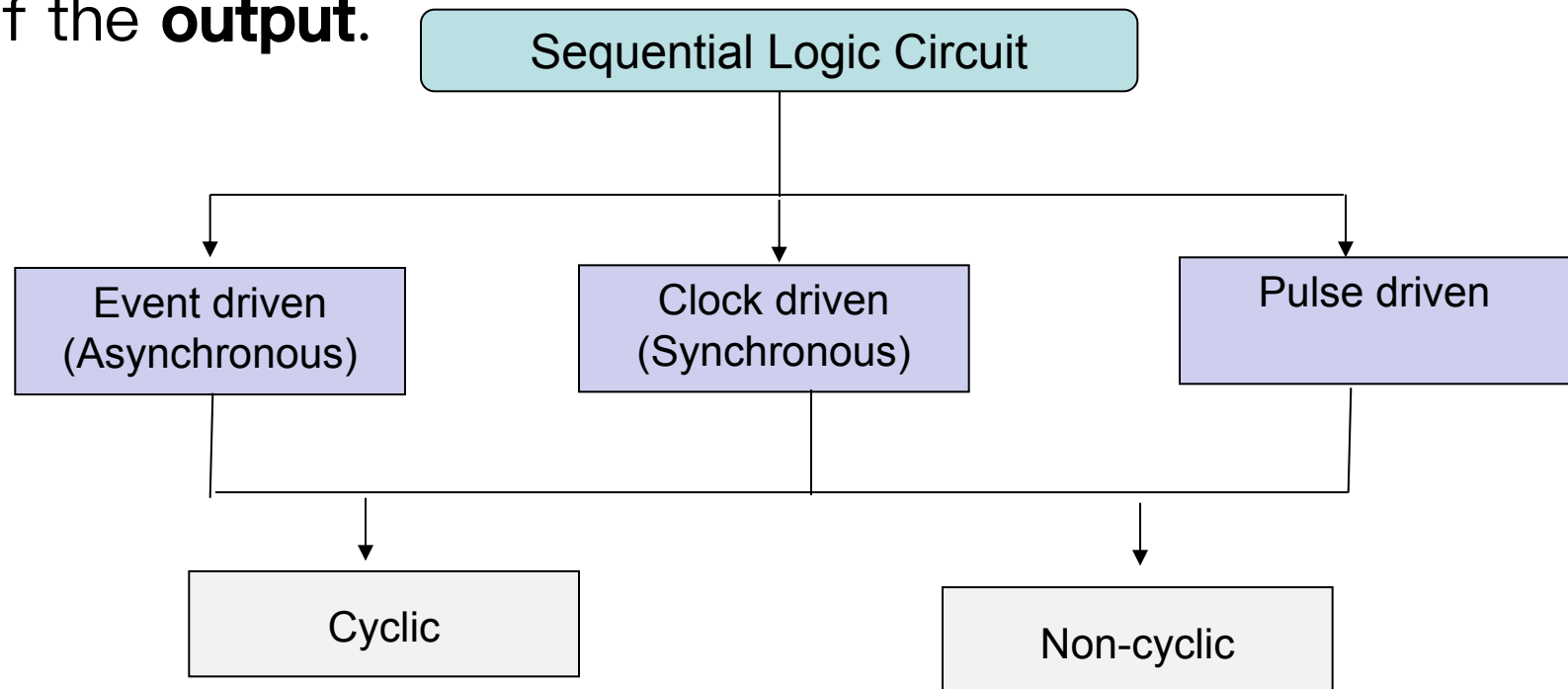
CAPITAL CITY UNIVERSITY

Combinational logic

- Outputs that respond immediately to inputs at some instant in time. Output is only determined by the logic functions and state of the inputs.

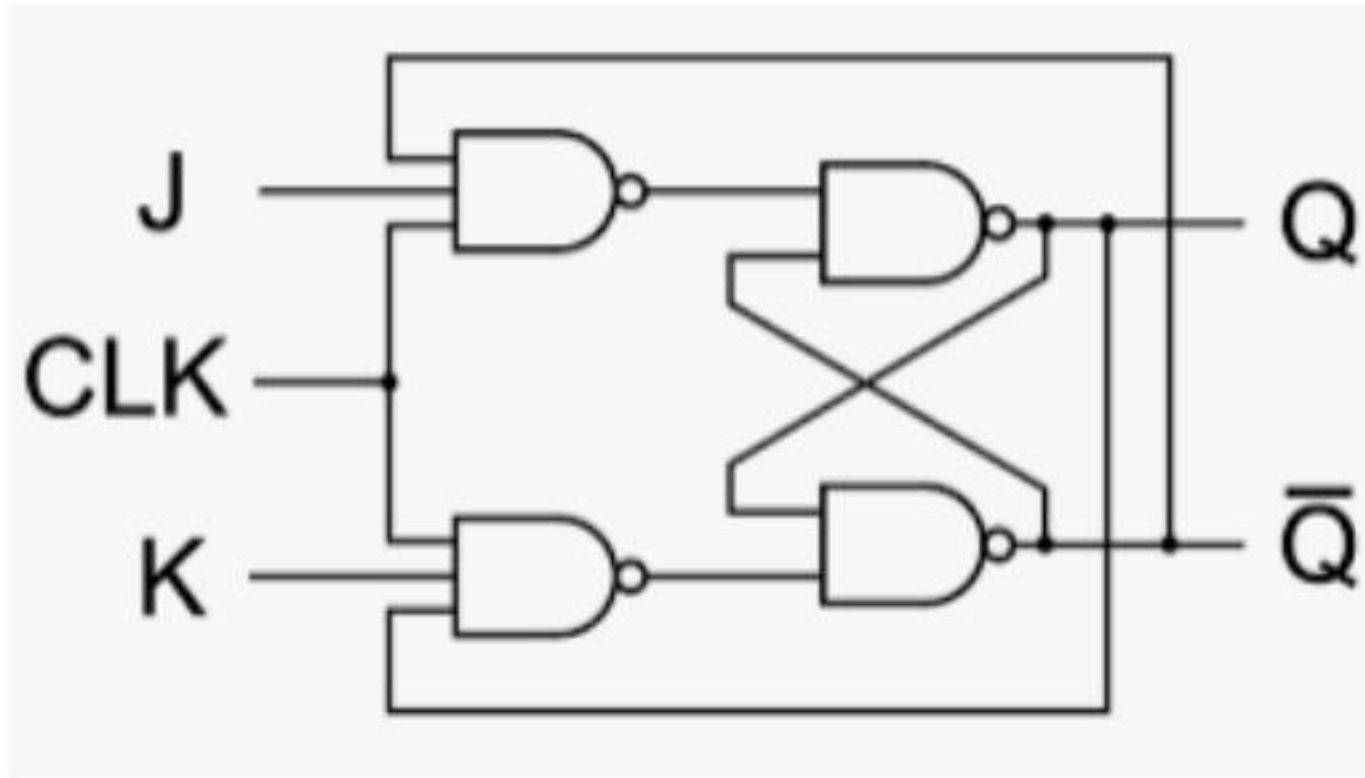


- So far — **combinational logic** – outputs that respond immediately to inputs at some instant in time. Output is only determined by the logic functions and state of the **inputs**.
- **Sequential Logic**: The output is determined not only by the state of the **inputs**, but also by the previous state of the **output**.



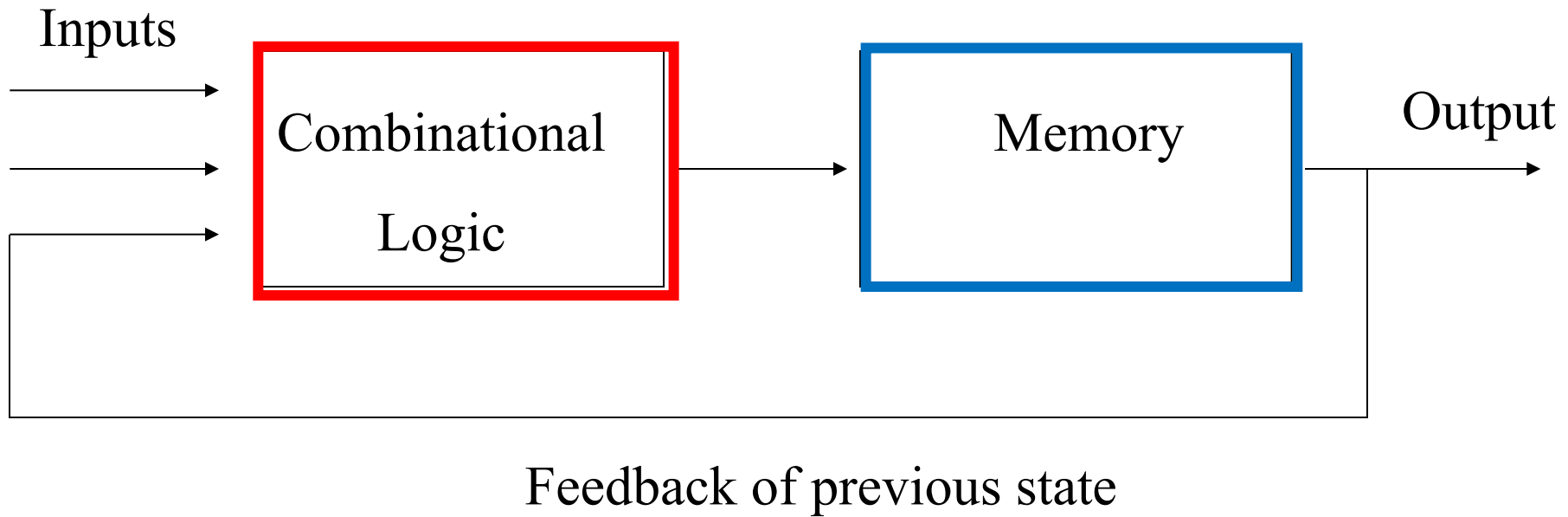
- So far — **combinational logic** – outputs that respond immediately to inputs at some instant in time. Output is only determined by the logic functions and state of the inputs.
- **Sequential Logic**: The output is determined not only by the state of the inputs, but also by the previous state of the output.
- **Sequential circuit** is thus a combinations of the combinational logic elements as well as memory elements — the output is a function of both the input as well as the information stored in memory.

- Look at the flip-flop as a building block for sequential logic (the memory element).



- Flip-flops are the core elements in counters and memory systems.

Sequential Logic

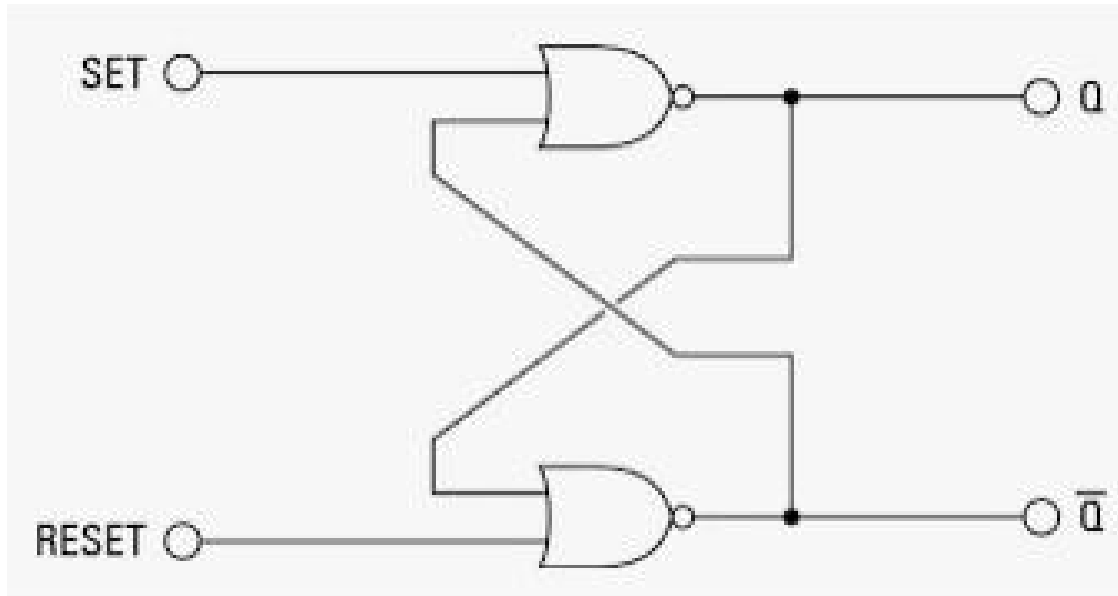


Latches, Flip-Flops, multivibrators – what is what ?

All have two stable output states – can switch between these using sequential logic

Latch is: Asynchronous

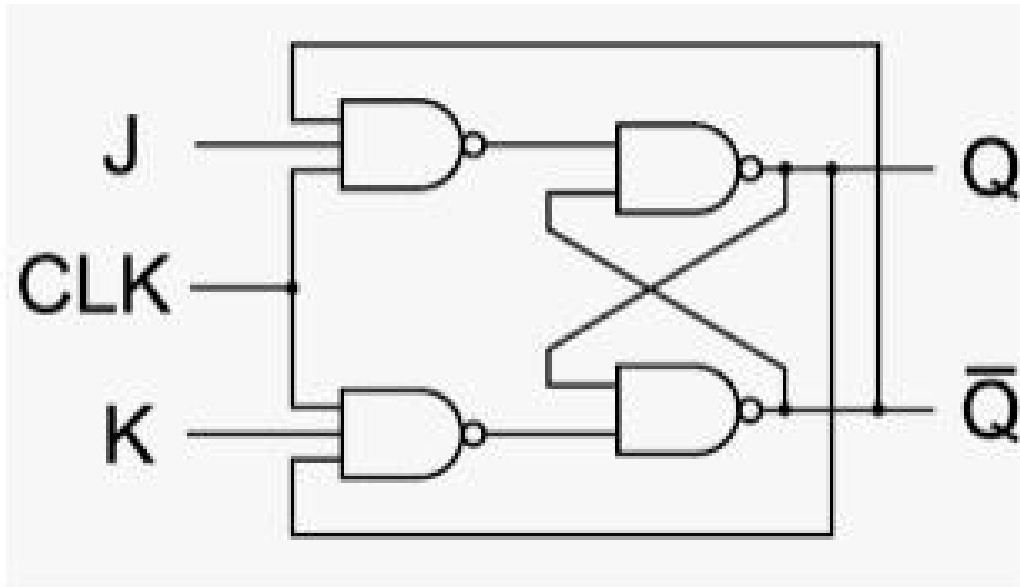
Output is triggered by input signals



Latches, Flip-Flops, multivibrators – what is what ?

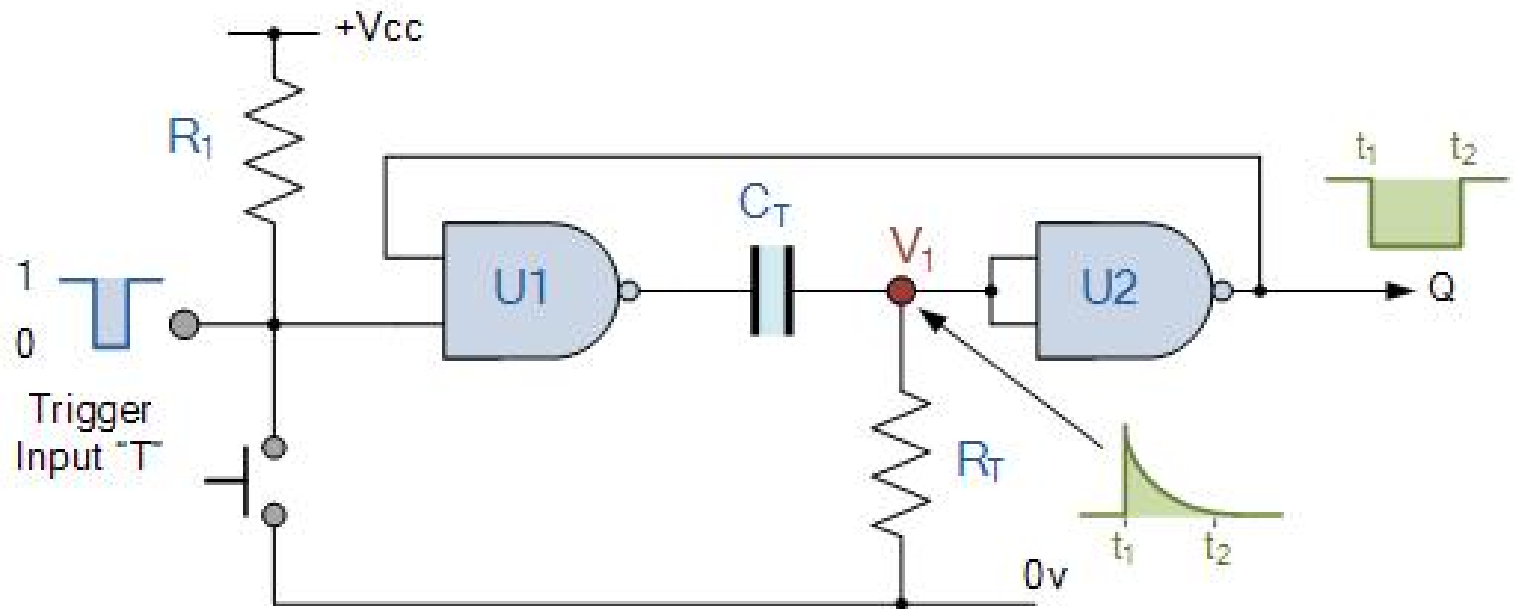
Flip-flop is: Synchronous

Edge triggered by clock signal



Latches, Flip-Flops, multivibrators – what is what ?

Simple NAND Gate Monostable Circuit

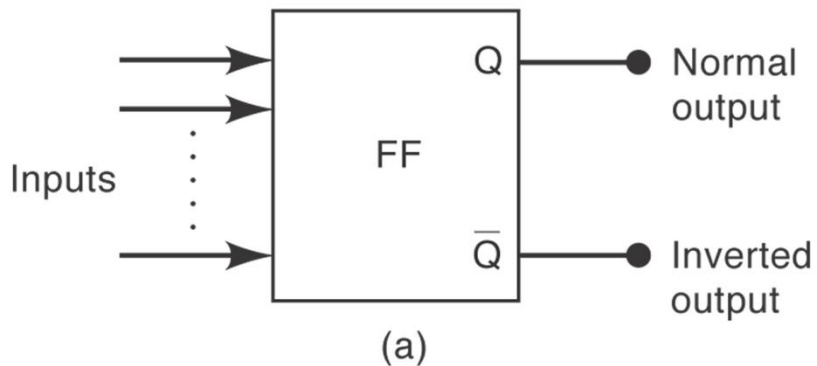


https://www.electronics-tutorials.ws/sequential/seq_3.html

Multivibrators – impressive name – ignore for now (more technical name for a flip-flop)

- Latches and flip-flops (FF) are key elements of counters and memory systems.
- The outputs of a latch/(FF) are Q and \overline{Q} .
- Q is understood to be the normal output, \overline{Q} is always the opposite.
- When the normal output (Q) is placed in the high or 1 state we say the latch/FF has been set.
- When the normal output (Q) is placed in the low or 0 state we say the latch/FF has been cleared or reset.

- When the normal output (Q) is placed in the high or 1 state we say the latch/FF has been set.
- When the normal output (Q) is placed in the low or 0 state we say the latch/FF has been cleared or reset.



Output states

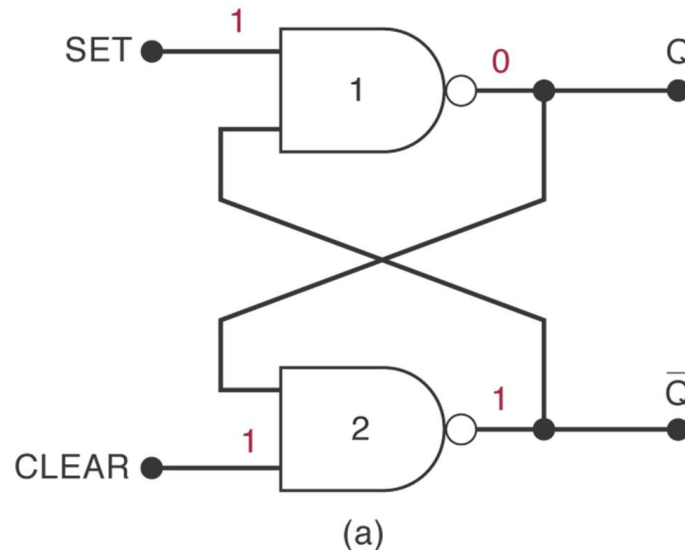
$Q = 1, \bar{Q} = 0$: called HIGH or 1 state; also called SET state

$Q = 0, \bar{Q} = 1$: called LOW or 0 state; also called CLEAR or RESET state

(b)

The NAND Gate Latch

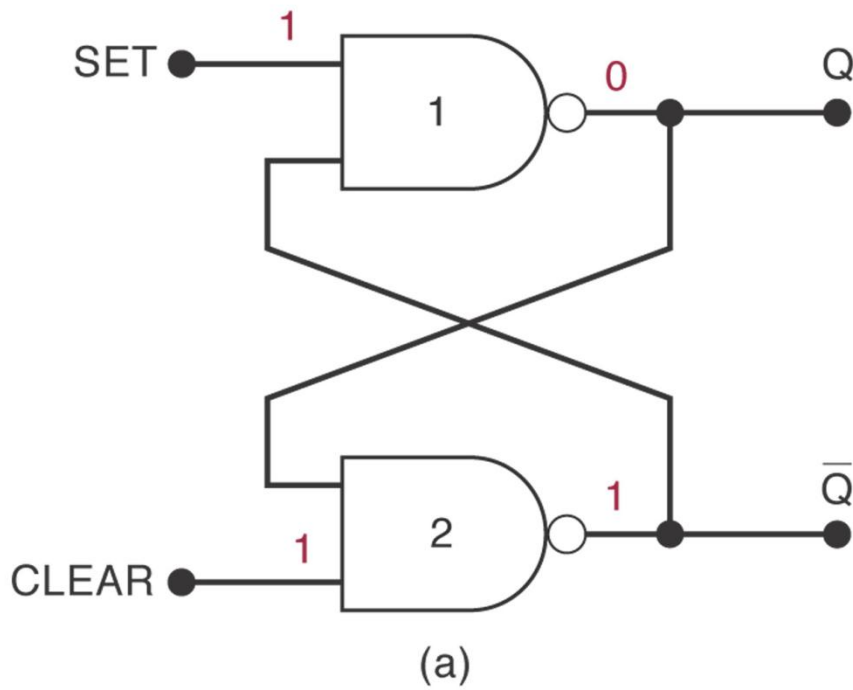
- The NAND gate latch is constructed from two cross-coupled NAND gates.
- The inputs are set (S) and clear/reset (C/R).



The NAND Gate Latch

- The NAND gate latch is constructed from two cross-coupled NAND gates.
- The inputs are set (S) and clear/reset (C/R).
- **The inputs are active low** - the output will change when the input is pulsed low.
- When the latch is set: $Q = 1$ and $\bar{Q} = 0$
- When the latch is clear or reset: $Q = 0$ and $\bar{Q} = 1$

Resting state of the NAND latch

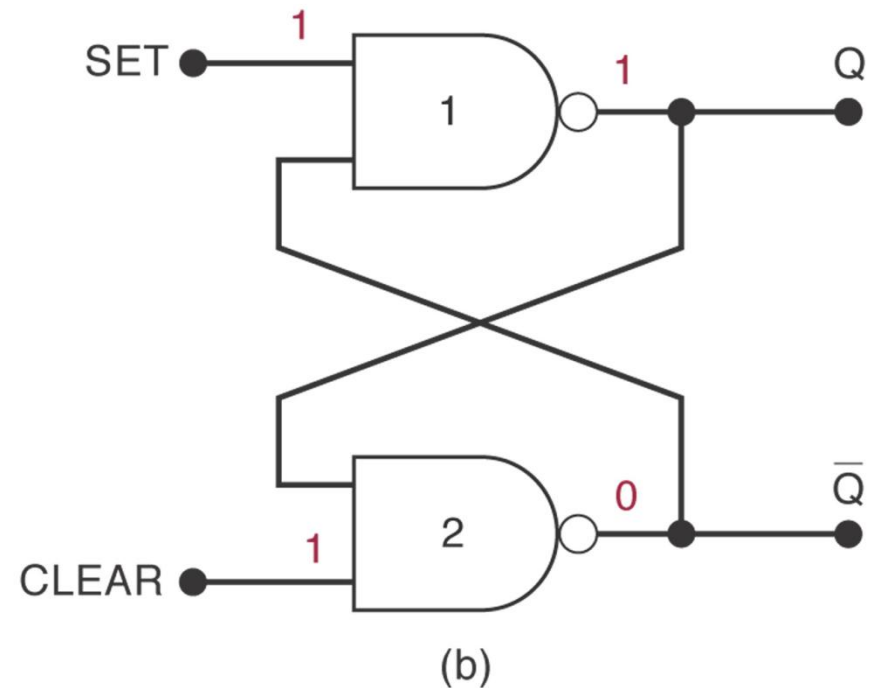
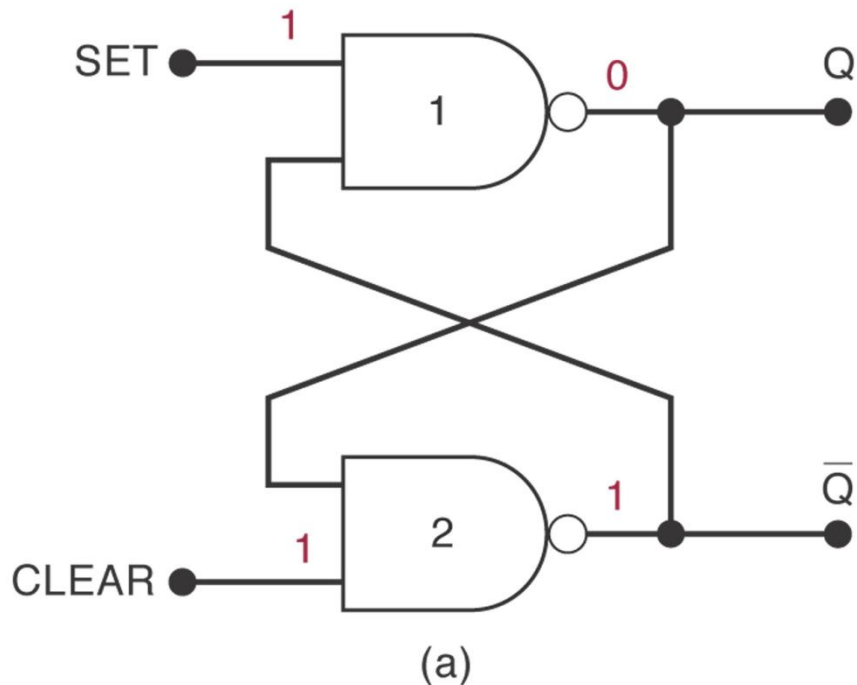


Start-up (default) state is $S=1=C$

Note that both output states (SET and CLEAR) are derived from the same input states: Both SET and CLEAR are HI – this is then also called the **resting** state of the latch.

NAND Gate		
A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

Resting state of the NAND latch



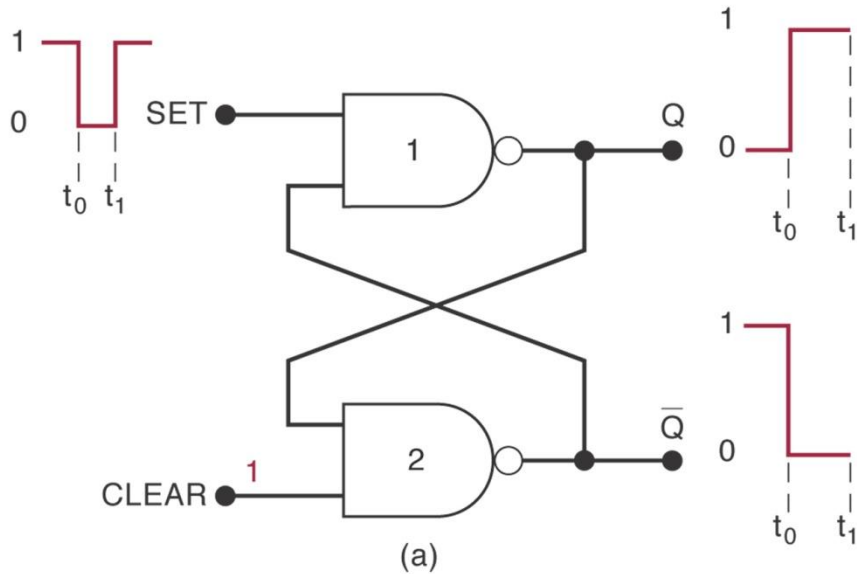
Start-up (default) state is $S=1=C$

Note that both output states (SET and CLEAR) are derived from the same input states: Both SET and CLEAR are HI – this is then also called the **resting state of the latch**.

NAND Gate		
A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

Setting the NAND latch

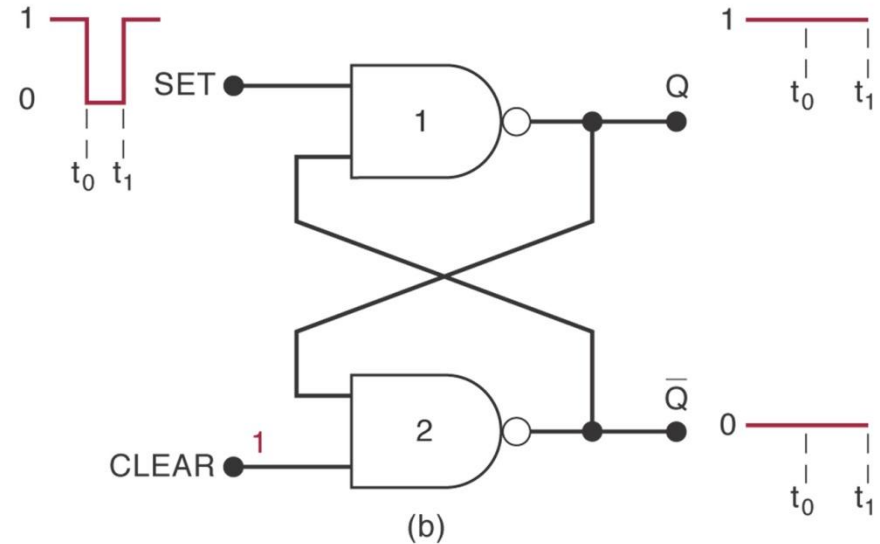
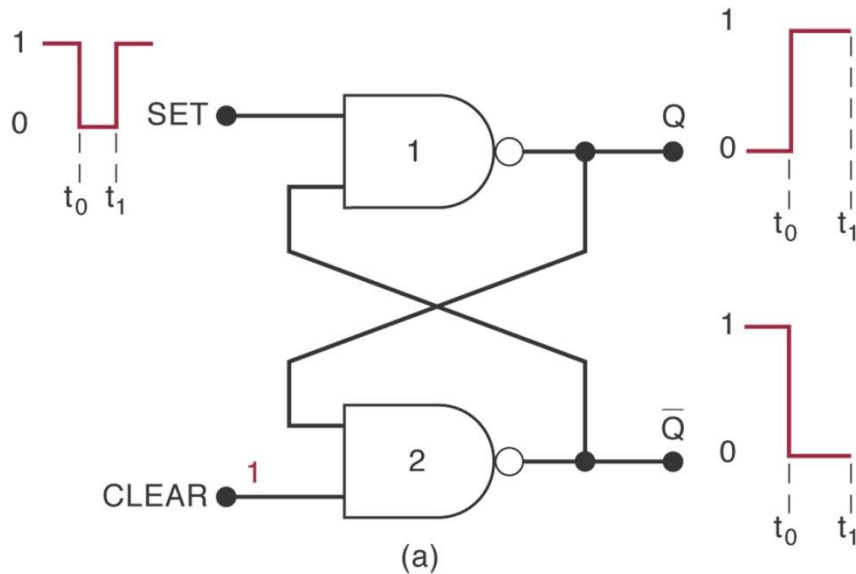
Active LO: Need to take SET LO to set.



If the latch is in RESET ($Q = 0$) and SET goes LO then we will see a change in output to the SET ($Q = 1$) output state.

Setting the NAND latch

Active LO: Need to take SET LO to set.



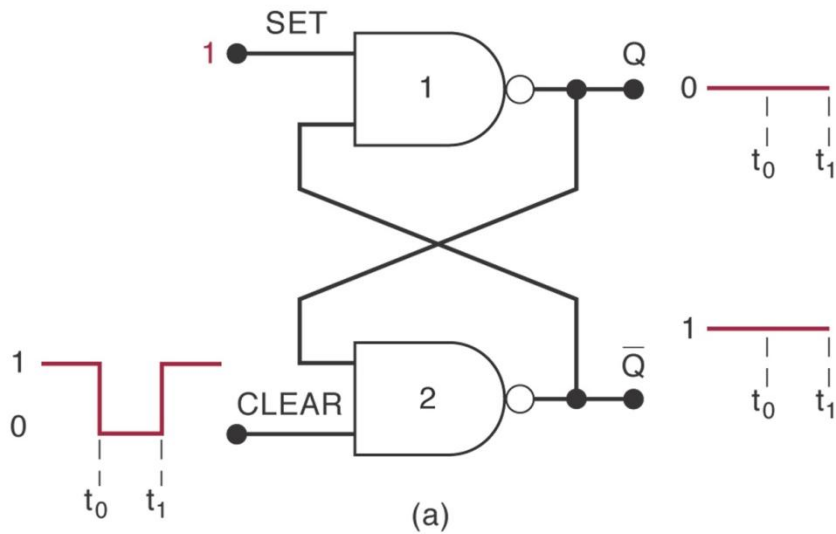
If the latch is in RESET ($Q = 0$) and SET goes LO then we will see a change in output to the SET ($Q = 1$) output state.

If the latch is in SET ($Q = 1$) and SET goes LO then we will see no change in output and $Q = 1$ will remain the output state.

The SET signal can thus go HI again after the LO pulse but the SET state ($Q = 1$) will remain. Recall: the resting state.

Clearing the NAND latch

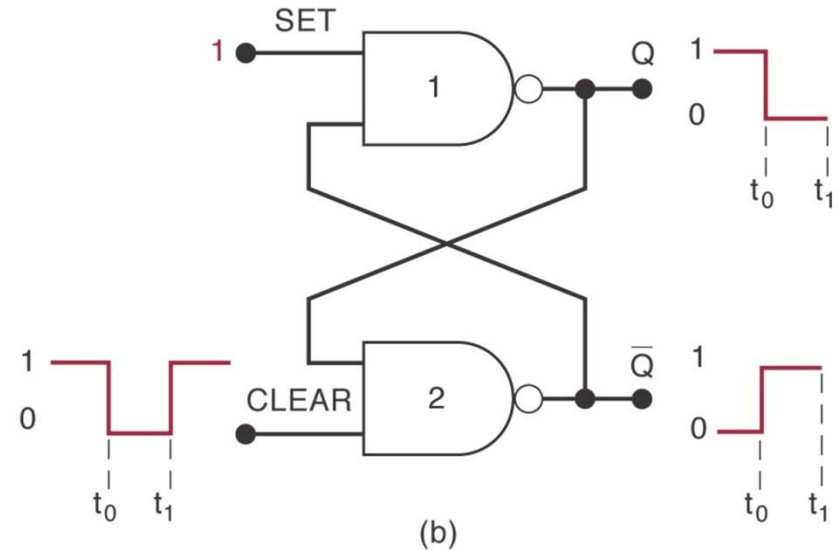
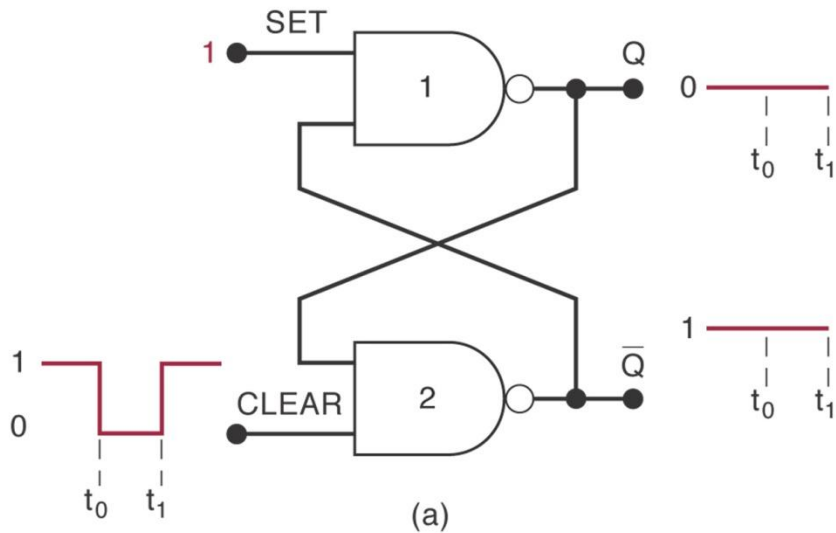
Active LO: Need to take CLEAR LO to reset.



If the latch is in SET ($Q = 1$) and RESET goes LO then we will see a change in output to the RESET ($Q = 0$) output state.

Clearing the NAND latch

Active LO: Need to take CLEAR LO to reset.

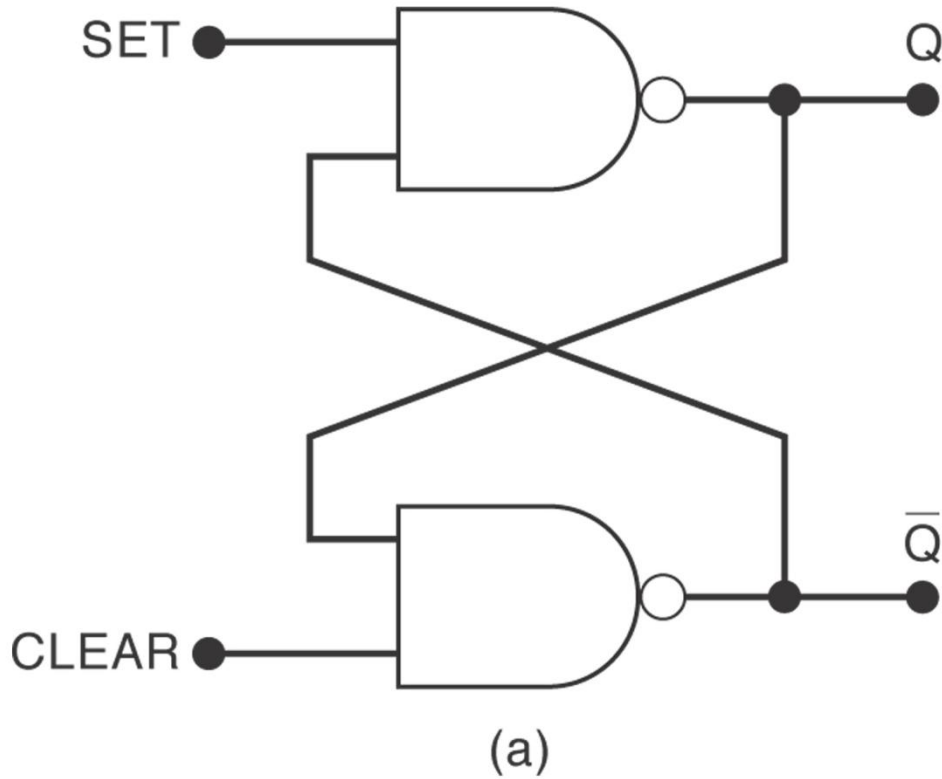


If the latch is in RESET ($Q = 0$) and RESET goes LO then we will see no change in output and $Q = 0$ will remain the output state.

If the latch is in SET ($Q = 1$) and RESET goes LO then we will see a change in output to the RESET ($Q = 0$) output state.

The RESET signal can thus go HI again after the LO pulse but the SET state ($Q = 1$) will remain. Recall: resting state

The NAND gate latch and truth table.



Set	Clear	Output
1	1	No change
0	1	Q = 1
1	0	Q = 0
0	0	Invalid*

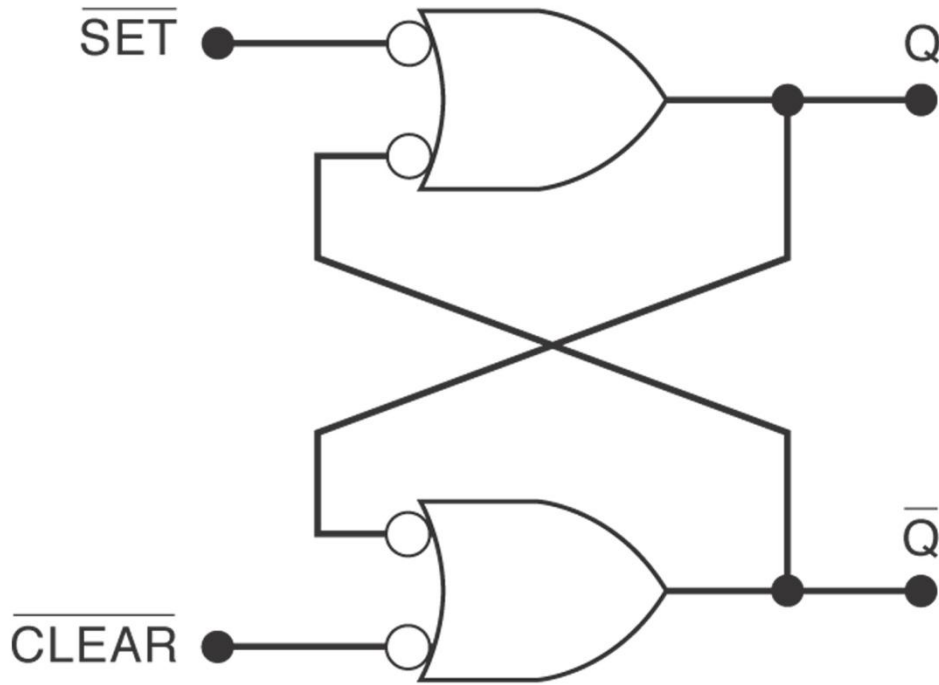
*Produces $Q = \bar{Q} = 1$.

(b)

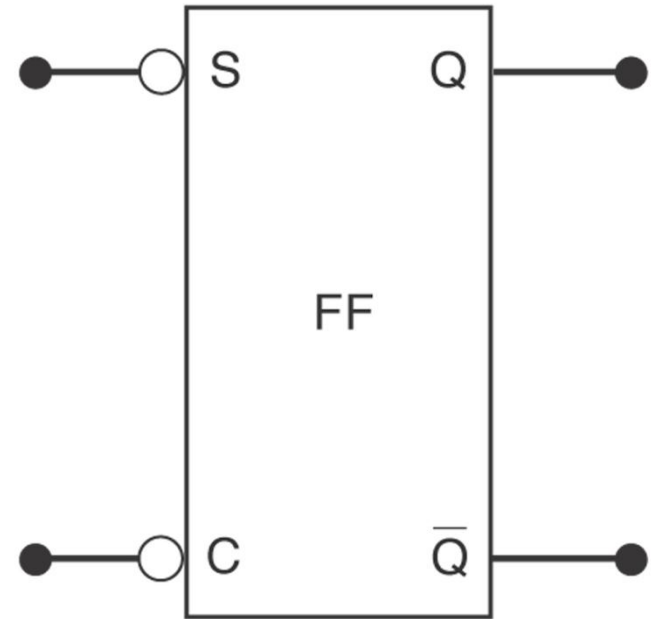
Summary of the NAND latch:

- **Set = Clear = 1.** Normal resting state, outputs remain in state prior to input.
- **Set = 0, Clear = 1.** Q will go high and remain high even if the Set input goes high.
- **Set = 1, Clear = 0.** Q will go low and remain low even if the Clear input goes high.
- **Set = Clear = 0.** Output is unpredictable because the latch is being set and cleared at the same time = undesired state

Alternative presentation of NAND gate latch



(a)



(b)