

FACULDADE DE COMPUTAÇÃO E INFORMÁTICA  
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO  
ÁLGEBRA BOOLEANA E CIRCUITOS DIGITAIS – Aula 10 – 1º SEMESTRE/2022  
Prof. Jamil Kalil Naufal Júnior

## CIRCUITOS SEQUENCIAIS

---



Nossos **objetivos** nesta aula são:

- Conhecer circuito sequenciais.
- Conhecer as aplicações de circuitos sequenciais.
- Praticar a implementação de circuitos sequenciais.



Para esta aula, utiliza-se como referência o **Capítulo 5 (Flip-flops e Circuitos Correlatos)** do livro-texto:

TOCCI, Ronald J.; WIDMER, Neal S.; MOSS, Gregory L. **Sistemas Digitais: princípios e aplicações**. 12ª Ed. Editora Pearson, 2019.

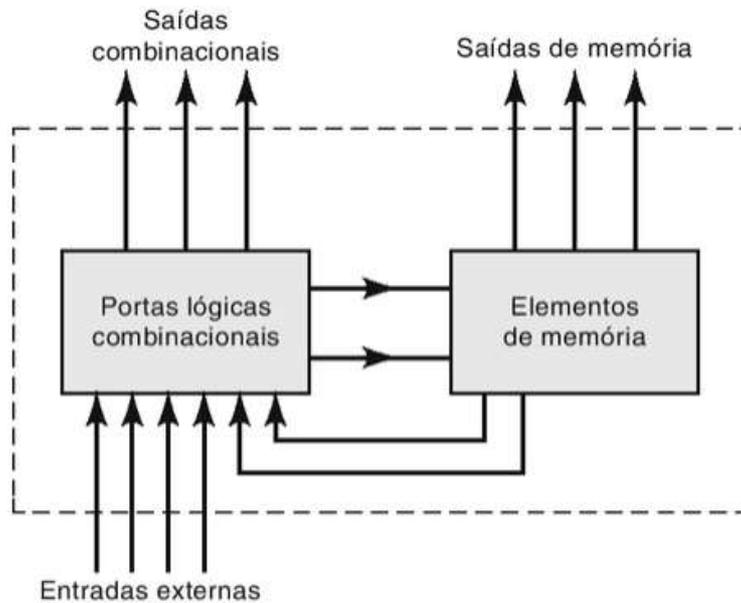
*Não deixem de ler a indicação depois desta aula!*

---

## RELEMBRANDO: CIRCUITO SEQUENCIAIS

---

- Os circuitos digitais podem ser classificados em combinacional e sequencial.
- O circuito combinacional depende unicamente das entradas para determinar a sua saída.
- O circuito sequencial possui uma **realimentação da saída para a entrada**.
- Denomina-se de estado interno a realimentação do sistema.
- As condições atuais da entrada e do estado interno determinam a condição futura da saída.



## **LATCHES & FLIP-FLOPs**

---

- Tipos de dispositivos de armazenamento temporário (volátil).
- São circuitos que apresentam **dois estados estáveis** (0 e 1) na saída, portanto denominados de **biestáveis**.
- Existem dois tipos básicos de biestáveis:
  - **Latches**: permite a troca de estado de maneira assíncrona ou síncrona, ou seja, muda o seu estado por ação de um pulso de disparo de relógio interno (clock).
  - **Flip-Flops**: permite a troca de estado apenas de maneira síncrona por borda de subida ou descida, ou seja, somente durante a rampa de mudança de estado do relógio interno (clock).
- *Latches* são mais utilizados como memória e os Flip-Flops como contadores e registradores.
- Os circuitos **sequenciais** têm as saídas dependentes das **variáveis de entrada** e/ou de seus **estados anteriores que permanecem armazenados**, sendo geralmente, sistemas pulsados, ou seja, operam sob o comando de uma sequência de pulsos denominados de *clock*.
- Quando pela ação de um estímulo externo, em sua(s) entrada(s), passam de um estado a outro e lá permanecem até que outro estímulo seja dado novamente em sua(s) entrada(s).
- Devido a característica de manter um determinado estado, os biestáveis são denominados também de **elementos de memória**.

- Existem basicamente quatro tipos de biestáveis:

Tipo RS

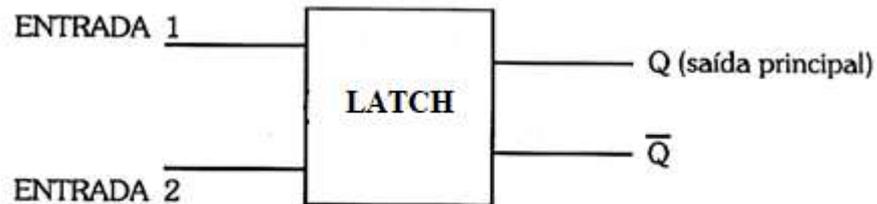
Tipo D

Tipo JK

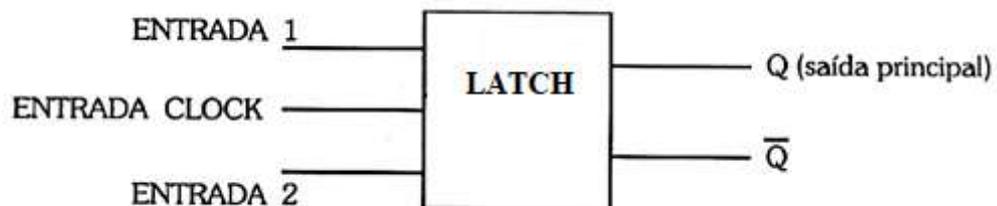
Tipo T

## Bloco lógico - Símbolo

### Assíncrono



### Síncrono



### ***LATCH SR (SET - RESET)***

---

- *Latch* é um tipo de dispositivo lógico biestável ou multivibrador.
- Entrada é ativa em nível alto (nível 1).
- Pode ser composto com duas portas NAND tendo um acoplamento cruzado (retroalimentação).

## Circuito Lógico baseado em portas NAND

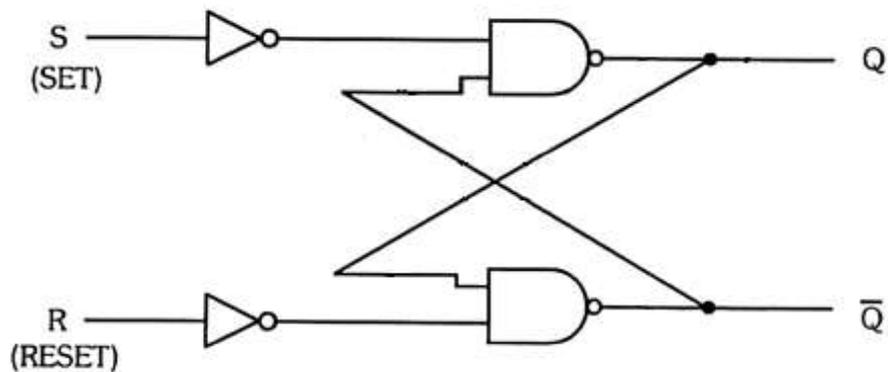
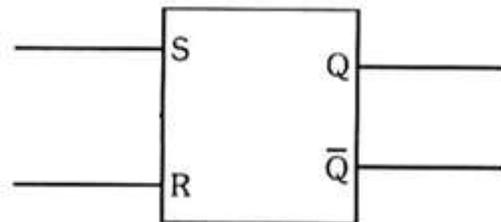


Tabela Verdade

S	R	Q
0	0	Mantém
0	1	0
1	0	1
1	1	Proibido

Bloco lógico



**Vantagem:** simplicidade

**Desvantagens:**

- Não há clock.
- Condição proibida, na situação  $R = S = 1$  denomina-se estado proibido, porque não tem sentido lógico querer resetar (R) e setar (S) ao mesmo tempo.

### ANÁLISE DO COMPORTAMENTO DO CIRCUITO

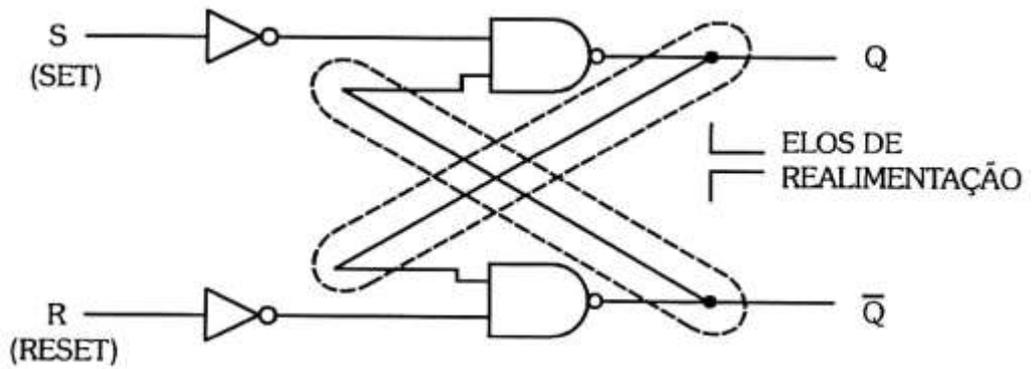
---

- Para analisarmos o comportamento do circuito, deve-se construir a tabela verdade levando em consideração as 2 variáveis de entrada (S e R) e a **saída anterior (Qa)** à aplicação das entradas.

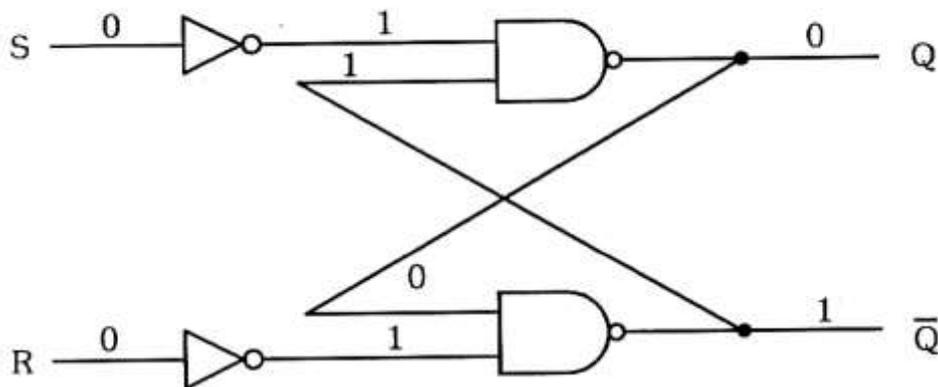
	S	R	Qa	Qf
0	0	0	0	
1	0	0	1	
2	0	1	0	
3	0	1	1	
4	1	0	0	
5	1	0	1	
6	1	1	0	
7	1	1	1	

estado anterior da saída Q.

estado que a saída deve assumir (estado futuro) após a aplicação das entradas.

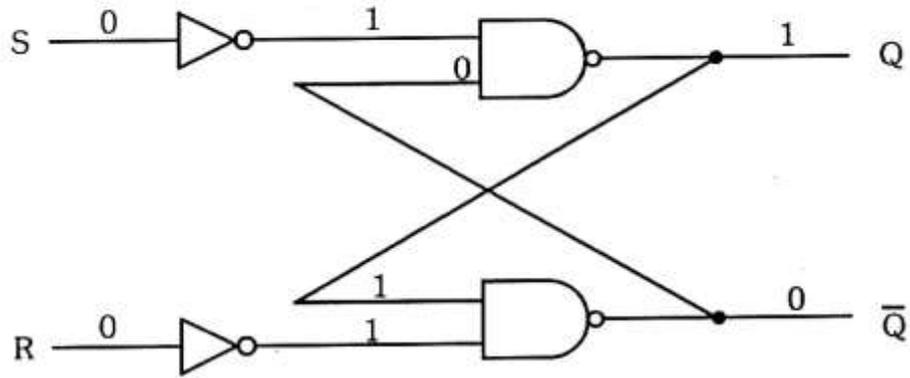


**Caso 0:**  $S = 0$ ,  $R = 0$  e  $Qa = 0 \rightarrow \bar{Q}a = 1$



Estado estável onde se obtém  $Qf = Qa = 0$ .

**Caso 1:**  $S = 0$ ,  $R = 0$  e  $Q_a = 1 \rightarrow \bar{Q}_a = 0$

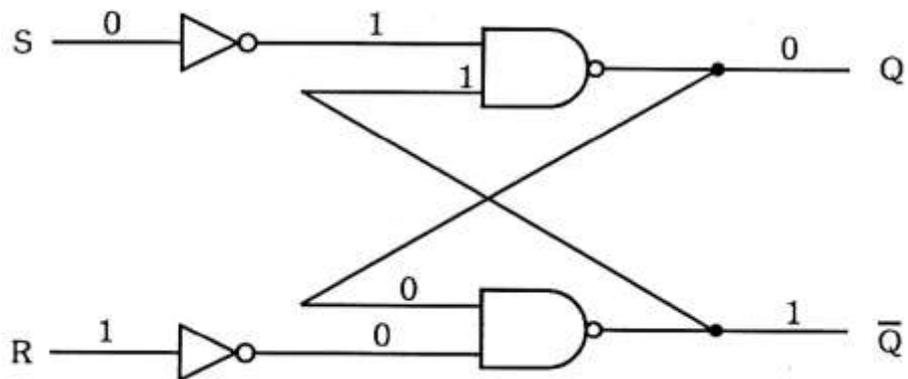


Estado estável onde se obtém  $Q_f = Q_a = 1$ .

Portanto, para  $S = R = 0$  obtém  $Q_f = Q_a$

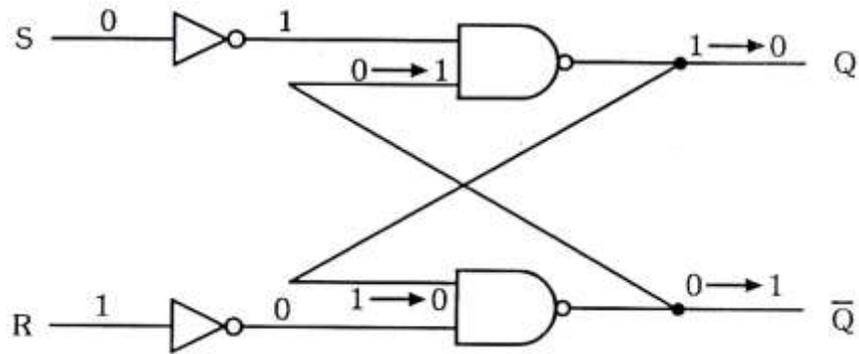
**(Mantém o estado anterior).**

**Caso 2:**  $S = 0$ ,  $R = 1$  e  $Q_a = 0 \rightarrow \bar{Q}_a = 1$



Estado estável onde se obtém  $Q_f = 0$ .

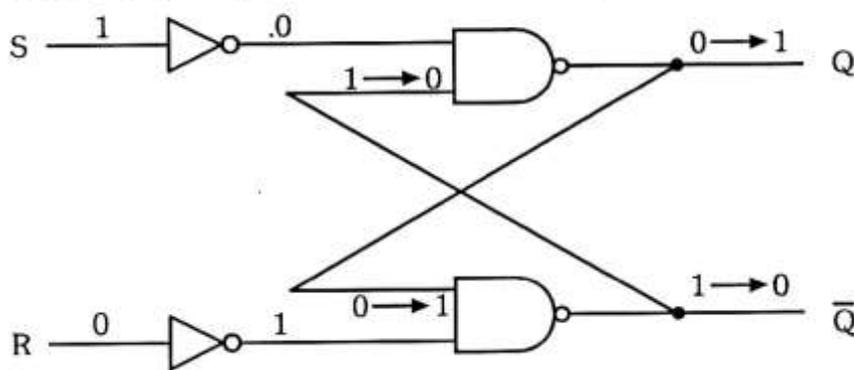
**Caso 3:**  $S = 0$ ,  $R = 1$  e  $Q_a = 1 \rightarrow \bar{Q}_a = 0$



Estado instável, pois irá mudar para 1, forçando assim que Q assumira valor 0 e aí sim, obtermos um estado estável **Qf = 0**.

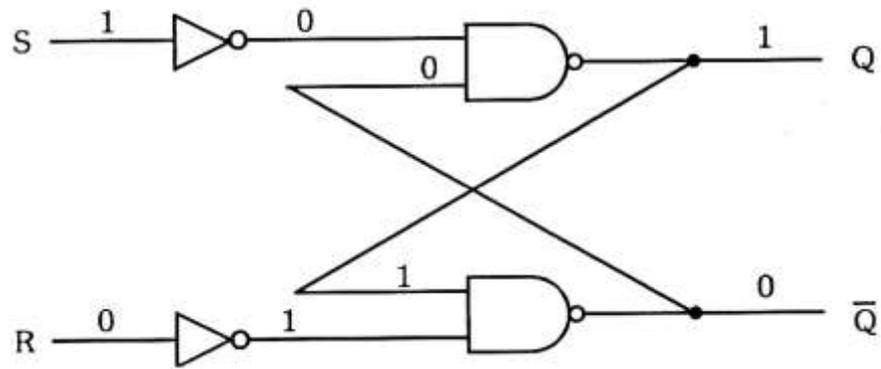
Portanto, para  $S = 0$  e  $R = 1$  obtém **Qf = 0**

**Caso 4:**  $S = 1$ ,  $R = 0$  e  $Q_a = 0 \rightarrow \bar{Q}_a = 1$



Estado instável, pois Q irá assumir valor 1 e consequentemente  $\bar{Q}$  assumirá valor 0. Obtém **Qf = 1**.

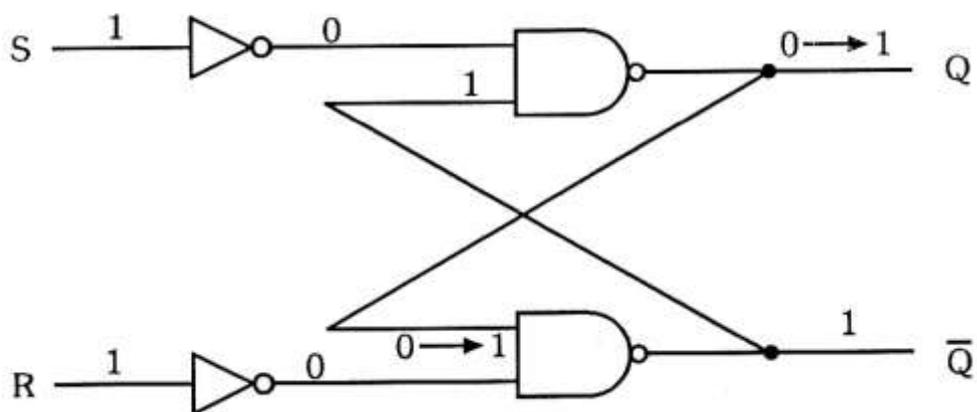
**Caso 5:**  $S = 1$ ,  $R = 0$  e  $Q_a = 1 \rightarrow \bar{Q}_a = 0$



Estado estável onde se obtém  $Q_f = 1$ .

Portanto, para  $S = 1$  e  $R = 0$  obtém  $Q_f = 1$

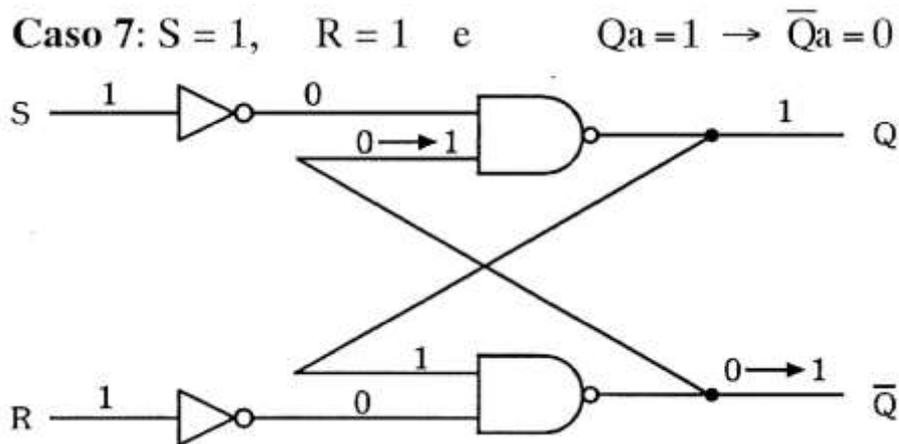
**Caso 6:**  $S = 1$ ,  $R = 1$  e  $Q_a = 0 \rightarrow \bar{Q}_a = 1$



Estado instável, pois Q irá assumir valor 1.

Note que  $Q'$  também assumirá valor 1.

**Situação não permitida**



Situação instável análoga ao Caso 6.

**Situação não permitida**

## RESUMO

### Latch RS Assíncrono Baseado em Portas NE e NOT (Tabela)

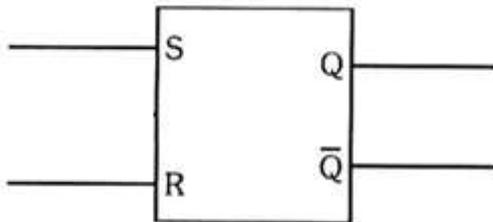
S	R	$Q_a$	$Q_f$	$\bar{Q}_f$	
0	0	0	0	1	} fixa $Q_f = Q_a$
0	0	1	1	0	
0	1	0	0	1	} fixa $Q_f$ em 0
0	1	1	0	1	
1	0	0	1	0	} fixa $Q_f$ em 1
1	0	1	1	0	
1	1	0	1	1	} não permitido
1	1	1	1	1	

S	R	$Q_f$
0	0	$Q_a$
0	1	0
1	0	1
1	1	<del>X</del>

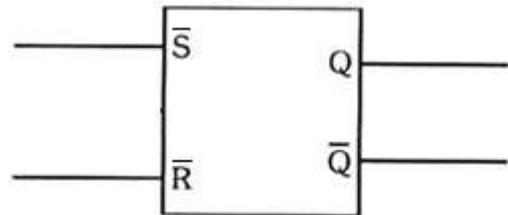
- Símbolos lógicos para os *latches* com entradas ativas em nível ALTO e entradas ativas em nível BAIXO.

## Bloco lógico

LATCH NÍVEL ALTO



LATCH NÍVEL BAIXO



LATCH NÍVEL ALTO

Entradas		Saídas		Comentários
S	R	Q	$\bar{Q}$	
0	0	Mantém	Mantém	Repouso! O <i>latch</i> permanece no estado atual.
0	1	0	1	<i>Latch</i> no estado RESET.
1	0	1	0	<i>Latch</i> no estado SET.
1	1	1	1	Condição inválida.

LATCH NÍVEL BAIXO

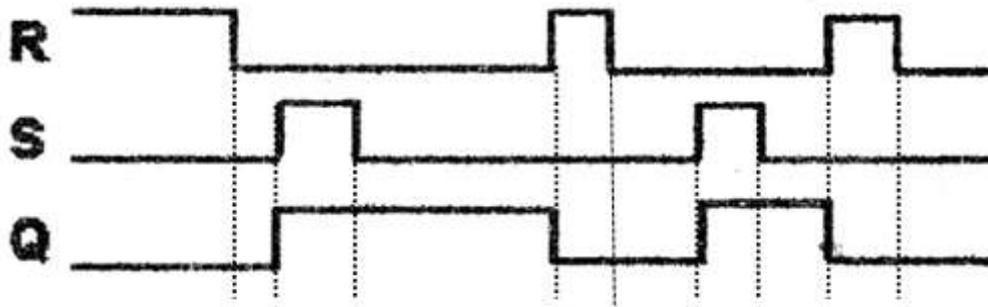
Entradas		Saídas		Comentários
$\bar{S}$	$\bar{R}$	Q	$\bar{Q}$	
0	0	1	1	Condição inválida.
0	1	1	0	<i>Latch</i> no estado SET.
1	0	0	1	<i>Latch</i> no estado RESET.
1	1	Mantém	Mantém	Repouso! O <i>latch</i> permanece no estado atual.

## DIAGRAMAS DE TEMPO

---

- Diagramas ou Cartas de tempo apresentam as alterações das saídas do circuito lógico em função das alterações das entradas ao longo do tempo.
- Para o *latch* SR temos o seguinte Diagrama de Tempo:

## DIAGRAMA DE TEMPO

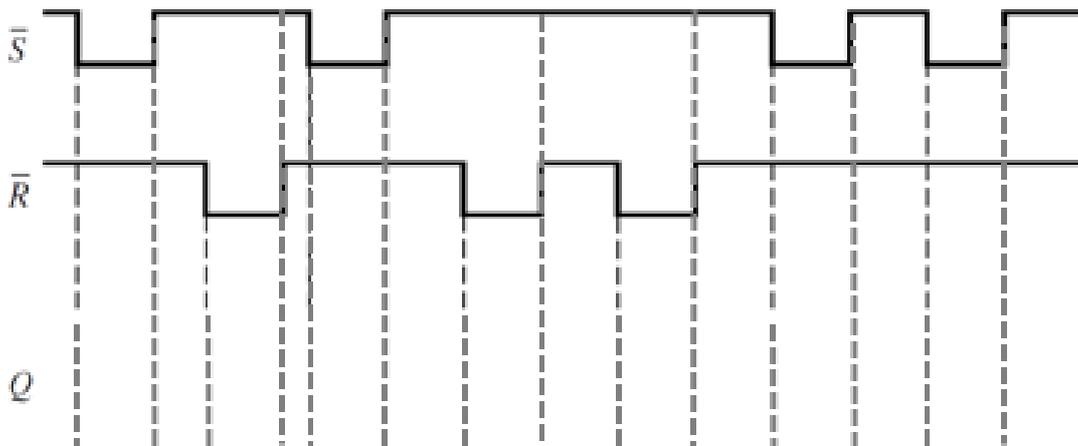


- Observe que o circuito do *latch* SR irá muda apenas no instante em que ocorrem mudanças nas variáveis de entrada.

## EXERCÍCIO TUTORIADO

---

1. As formas de onda  $\bar{S}\bar{R}$  apresentadas a seguir são aplicadas nas entradas do *latch*  $\bar{S}\bar{R}$  com as entradas ativas em nível BAIXO, determine a forma de onda observada na saída Q. Considere que Q está inicialmente em nível BAIXO.



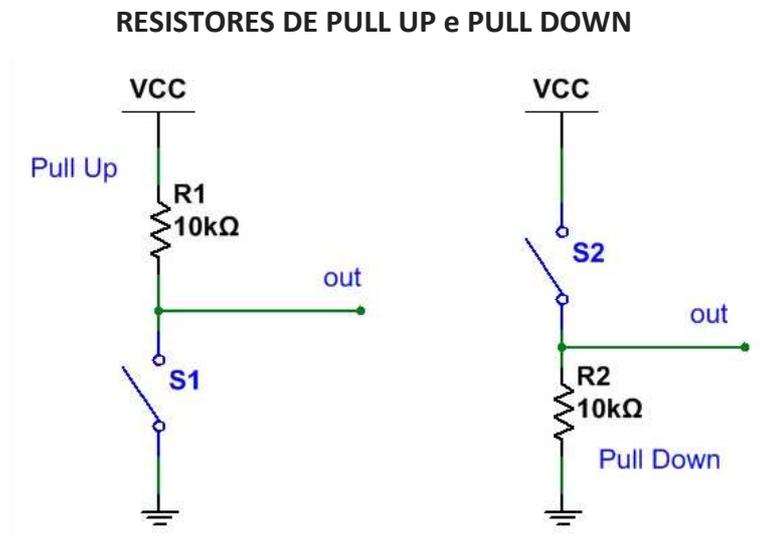
## APLICAÇÃO – EFEITOS DE FLUTUAÇÃO E TREPIDAÇÃO

---

### Efeito de Flutuação (Floating)

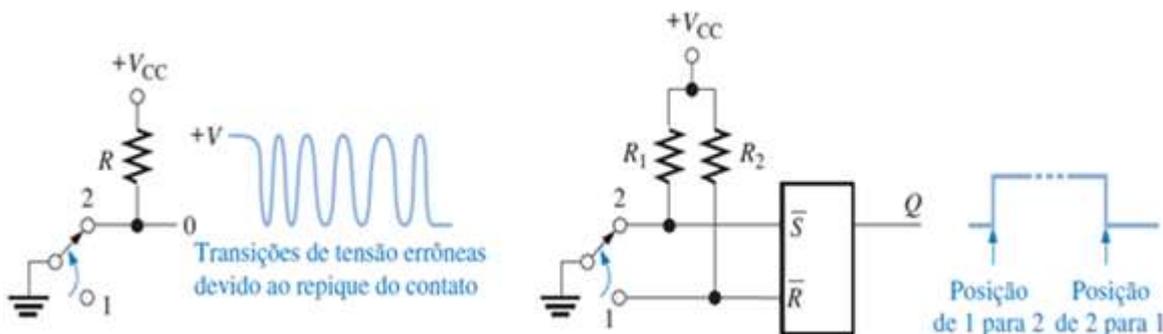
- Em circuitos lógicos digitais como TTL uma entrada não conectada a uma porta é chamada de entrada flutuante (*floating input*), pois a entrada flutua na tensão limite do dispositivo.
- Uma entrada TTL flutuante geralmente atua como uma entrada HIGH. No entanto, as entradas abertas são suscetíveis a ruídos, obtidos normalmente por entradas com cabos metálicos de interligação, tais como *jumpers*, que são susceptíveis a ruídos.

- Uma solução possível em um circuito digital típico envolve uma chave mecânica e um resistor interligados a VCC e GND.
- O resistor é conhecido como resistor de **Pull-Down** ou **Pull-Up**.
- Se o resistor estiver conectado ao GND, então é chamado de resistor de **Pull-Down**.
- Se o resistor estiver conectado ao VCC, então é chamado de resistor de **Pull-Up**.

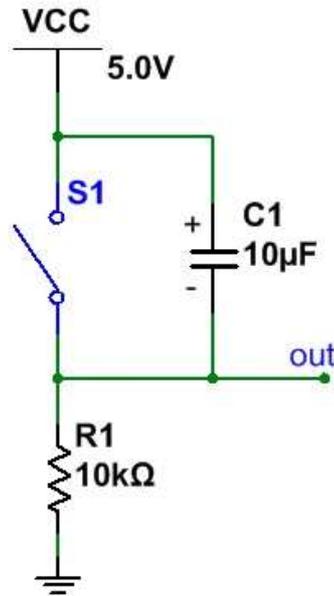


### Efeito de Trepidação (*Bouncing*)

- Quando o contato de uma chave comuta fechando o contato, este **contato pode vibrar** (trepidar) fisicamente várias vezes antes de finalmente estabelecer um contato firme e permitir um sinal elétrico estável.
- Esta trepidação (repique) é chamada de efeito *bouncing* de uma chave mecânica.

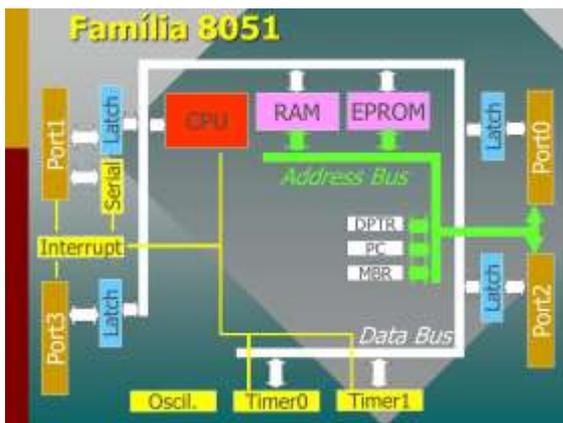


- Para eliminar a trepidação de contato pode ser utilizada um *latch* RS.
- Uma outra alternativa é adicionar um capacitor em paralelo a chave. O capacitor irá atenuar o sinal, sendo que a malha resistor/capacitor irá gerar um tempo de atraso no circuito. Este tempo de atraso fará com que rápidas oscilações indesejadas sejam atenuadas para o GND através do resistor.



## APLICAÇÃO DE LATCHES EM COMPUTAÇÃO

Os *latches* são algumas vezes usados em sistemas de computador para a **multiplexação de dados em um barramento**. Por exemplo, os dados que entram no computador a partir de uma fonte externa têm que **compartilhar o barramento de dados** com os dados de outras fontes.



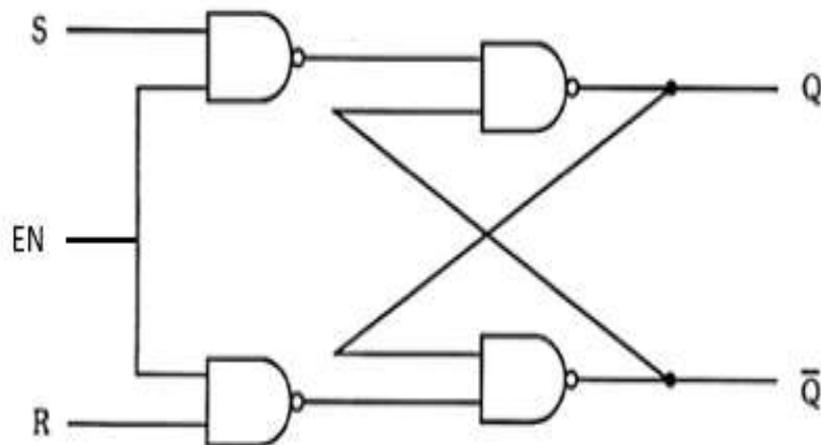
Quando o **barramento de dados se torna indisponível** para uma fonte externa, o dado existente tem que ser

**temporariamente armazenado e os latches** colocados entre a fonte externa e o barramento de dados podem ser usados para esse fim. Quando o barramento de dados está indisponível para a fonte externa, os *latches* têm que ser **desconectados do barramento** usando um método conhecido por **tristate**. Quando o barramento de dados se torna disponível, os dados externos passam através dos *latches*, originando assim a denominação de **latch transparente**. O *latch* D (será visto mais a frente) implementado com portas realiza essa função, porque quando ele está habilitado, o dado em sua entrada aparece na saída como se existisse uma conexão direta através dele. O dado na entrada é armazenado logo que o *latch* seja desabilitado.

## LATCH SR CONTROLADO

- Um *latch* controlado necessita de uma **entrada de habilitação**, Enable (EN), a letra G também é usada para indicar uma entrada de habilitação.
- As entradas S e R controlam o estado para o qual o *latch* irá quando um **nível ALTO** é aplicado a entrada EN.
- **Não ocorrerá mudança de estado** até que EN seja nível **ALTO**, porém enquanto essa entrada **permanecer em nível ALTO**, a **saída** é determinada pelos **estados das entradas S e R**.
- O estado **inválido** ocorre quando S e R forem **simultaneamente nível ALTO**.

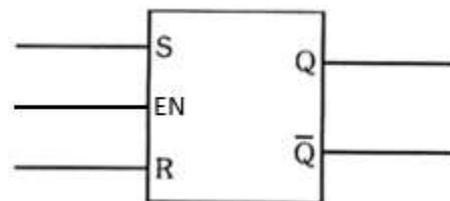
### Circuito Lógico baseado em portas NAND



### TABELA VERDADE

EN	S	R	Q	Q'
0	X	X	Não muda	
1	0	0	Não muda	
1	0	1	0	1
1	1	0	1	0
1	1	1	Proibido	

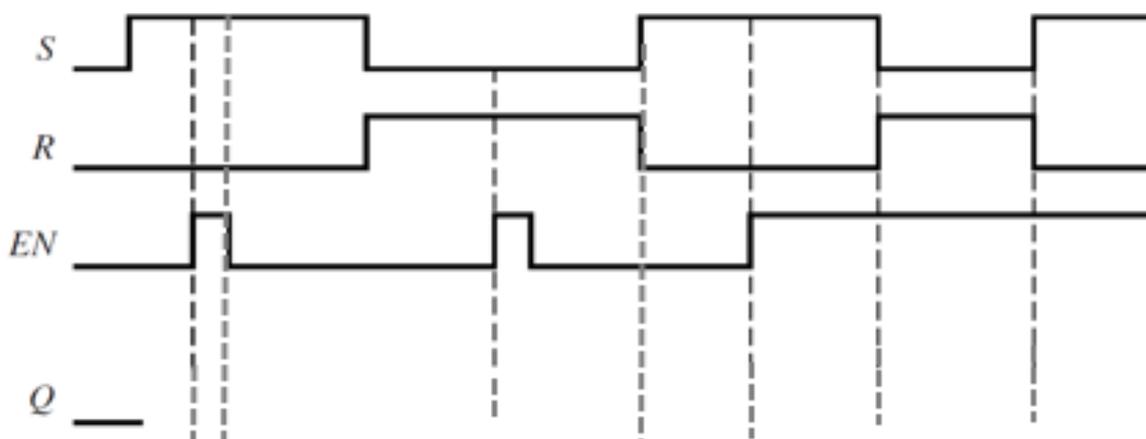
### BLOCO LÓGICO



## EXERCÍCIO TUTORIADO

---

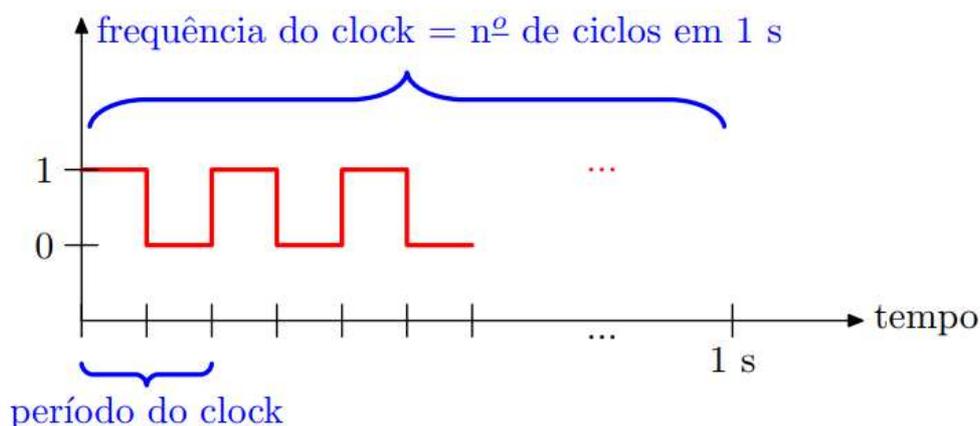
2. Determine a forma de onda da saída Q se as entradas abaixo forem aplicadas no *latch* SR controlado que está inicialmente resetado (nível BAIXO).



## RELÓGIO DE TEMPO (CLOCK)

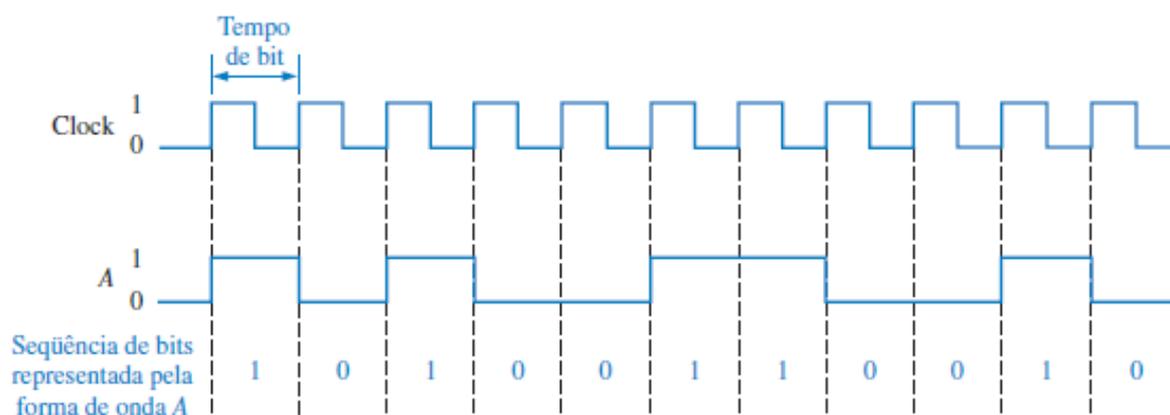
---

- **Relógio de tempo ou Clock:** forma de onda retangular contínua, utilizada para temporização (sincronismo) de circuitos lógicos.
- Sinal de relógio que determina quando determinados eventos devem acontecer no hardware computacional.
- Elementos básicos:
  - **Ciclo do *clock*:** ciclo repetitivo que engloba o nível lógico 0 e nível lógico em 1.
  - **Transição (borda) do *clock*:**
    - ✓ **Transição de subida:** ocorre na transição do nível lógico 0 para 1.
    - ✓ **Transição de descida:** ocorre na transição do nível lógico de 1 para 0.
- Período do clock ou tempo de ciclo do clock: duração de 1 ciclo do clock.
- Frequência do clock ou taxa do clock: número de ciclos em 1 segundo.

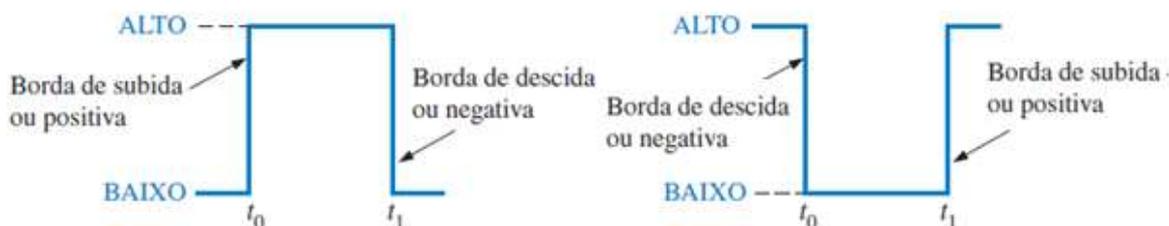


## SEQUÊNCIA DE BITS EM FORMATO TEMPORAL

- Formas de onda podem representar sequências de bits em um sistema digital.
- Em nível **ALTO** representa o número **um em binário**.
- Em nível **BAIXO** representa o número **zero em binário**.
- Tempo de bit** é o intervalo temporal finito existente em uma forma de onda.
- Em um *clock* cada intervalo entre os pulsos (período) é igual ao tempo de um bit.
- O clock não transporta informação.**

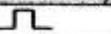


- Um único pulso positivo é gerado quando a tensão (ou corrente) passa do nível **BAIXO** normal para o nível **ALTO** e em seguida retorna para o nível **BAIXO**.



## BIESTÁVEIS SÍNCRONOS (LATCHES E FLIP-FLOPS)

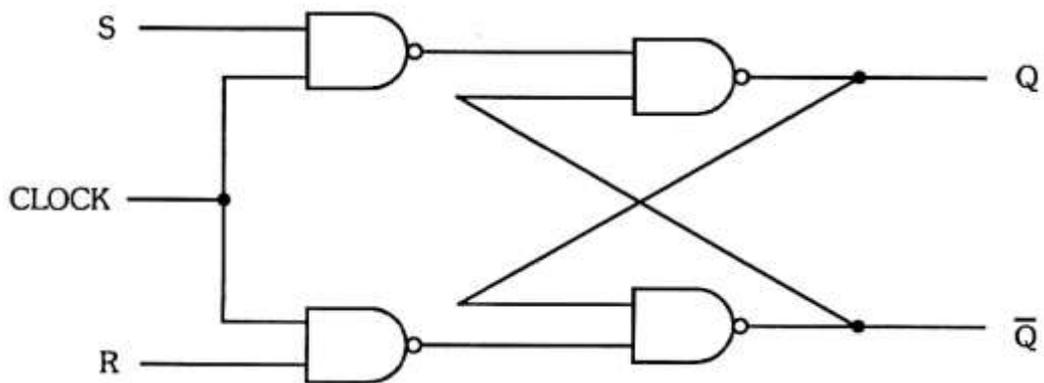
- Os biestáveis síncronos são aqueles que possuem dependência do relógio (*clock*), ou seja, há uma **entrada adicional** que receberá os **pulsos de clock**.
- O funcionamento pode ser de duas maneiras: **a nível** ou **a borda** (transição ou rampa).
- Os quadros a seguir apresentam a simbologia adotada para o funcionamento a nível alto e a nível baixo e, também para a borda de subida e de descida.

NÍVEL ALTO (positivo)	NÍVEL BAIXO (negativo)
1 ou 	0 ou 

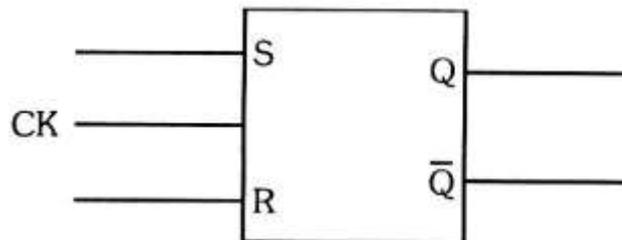
BORDA DE SUBIDA	BORDA DE DESCIDA
	

## LATCH SR: SÍNCRONO NÍVEL ALTO

### Circuito Lógico baseado em portas NAND



### Bloco lógico



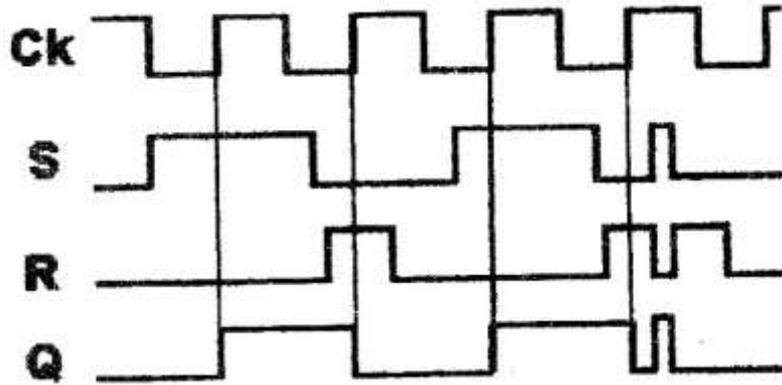
### Tabela Verdade

Ck	S	R	Q
1	0	0	Mantém
1	0	1	0
1	1	0	1
1	1	1	Proibido

### Tabela Verdade Simplificada

Ck	Qf
0	Qa
1	RS Básico

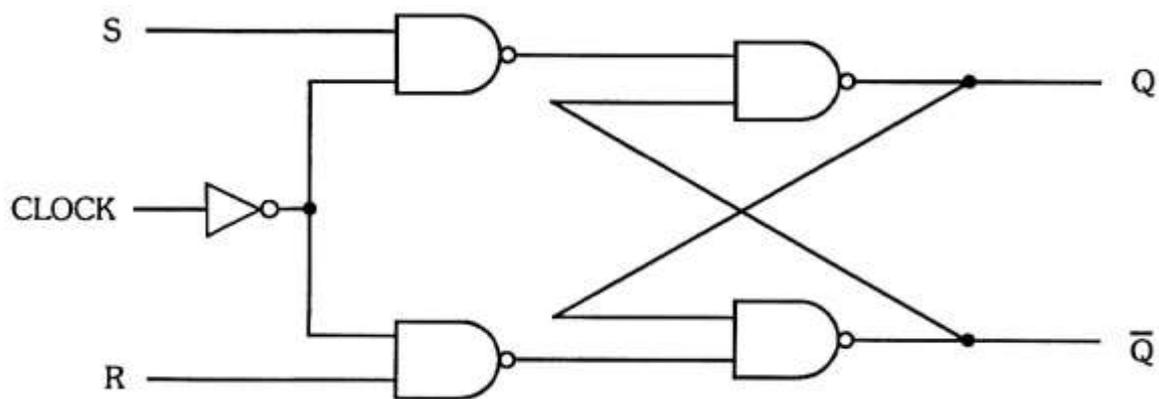
## Diagrama de Tempo



- A saída só é alterada enquanto o *clock* estiver em "1", respeitando a **tabela verdade** do tipo SR.

## LATCH RS: SÍNCRONO NÍVEL BAIXO

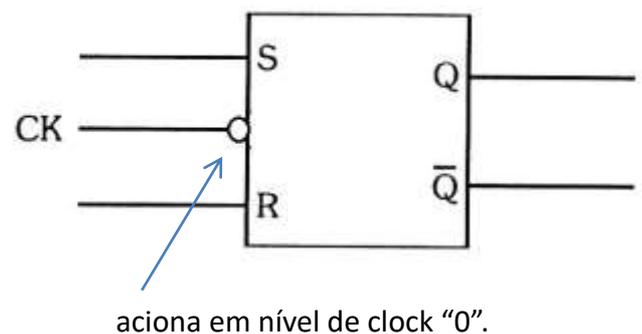
### Circuito Lógico baseado em portas NAND



### Tabela Verdade

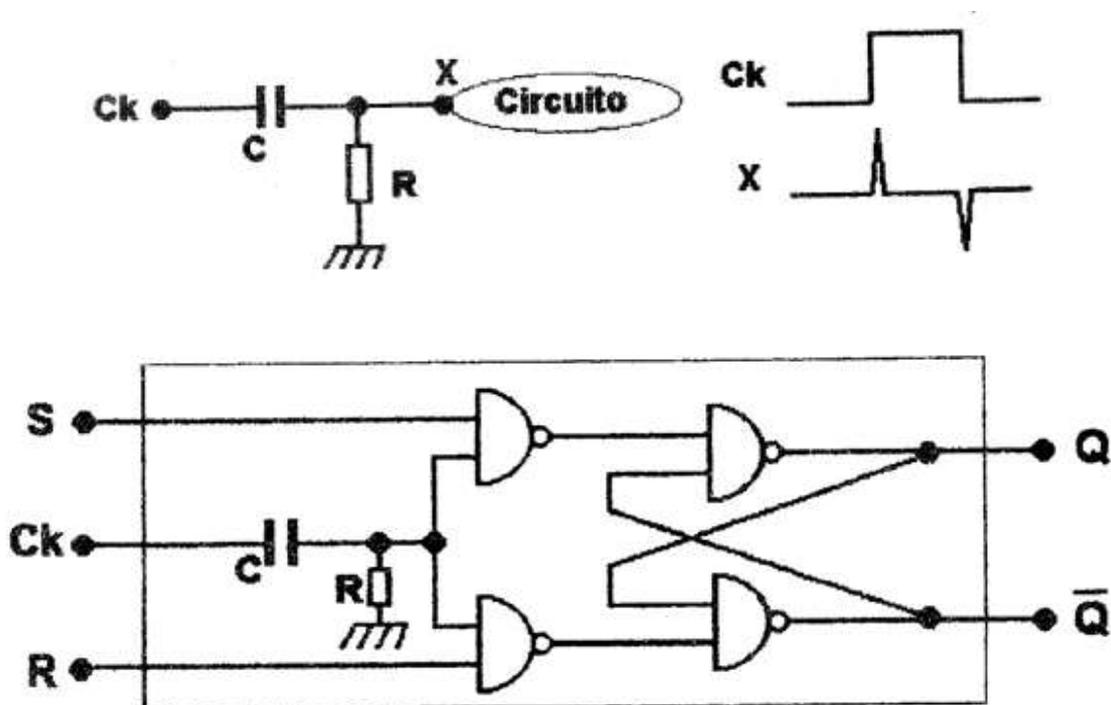
Ck	S	R	Q
0	0	0	Mantém
0	0	1	0
0	1	0	1
0	1	1	Proibido

### Bloco lógico



## FLIP-FLOP RS: SÍNCRONO BORDA DE SUBIDA

- O biestável de borda de subida só tem a sua **saída alterada** conforme as **condições de entrada** e na **transição do clock (de zero para um)**.
- Mesmo que o *clock* esteja em **nível lógico alto**, a **saída não se altera** independentemente do que foi colocado em suas entradas.
- A diferença do **circuito de nível alto** para o de **borda de subida** consiste no **acréscimo de um circuito diferenciador** colocado na entrada do *clock* fazendo com que o **funcionamento do mesmo após o diferenciador seja exatamente igual ao de nível alto**.



- No instante em que o *clock* vai a **nível lógico alto**, no **ponto X** o sinal vai de **0 para 1** e **instantaneamente retorna para 0**, permitindo apenas uma **leitura momentânea** das **entradas do biestável**.

Tabela Verdade

Ck	S	R	Q
↑	0	0	Mantém
↑	0	1	0
↑	1	0	1
↑	1	1	Proibido

Simbologia

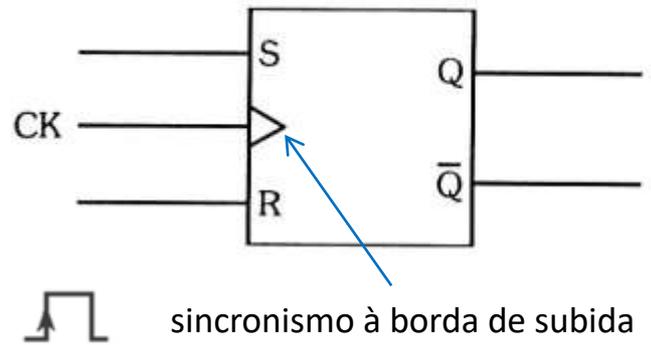
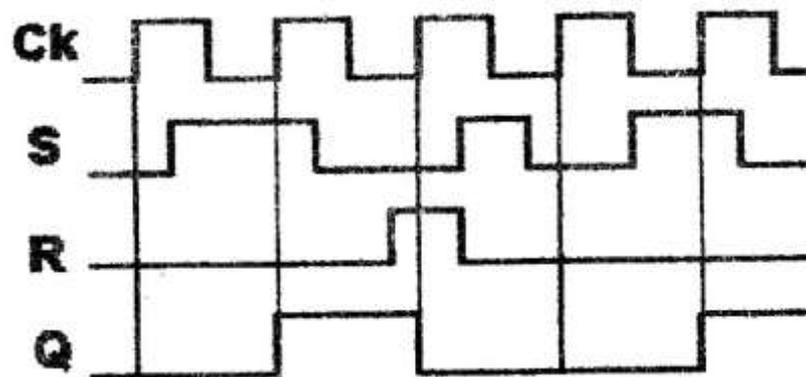


Diagrama de Tempos



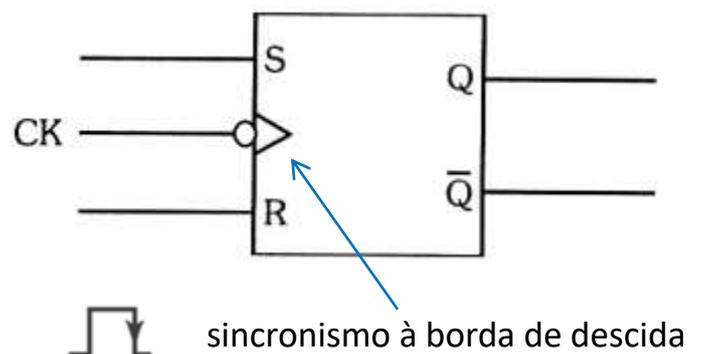
- A saída só tem possibilidade de ser alterada no instante em que o *clock* vai de 0 para 1.

**FLIP FLOP RS: SÍNCRONO BORDA DE DESCIDA**

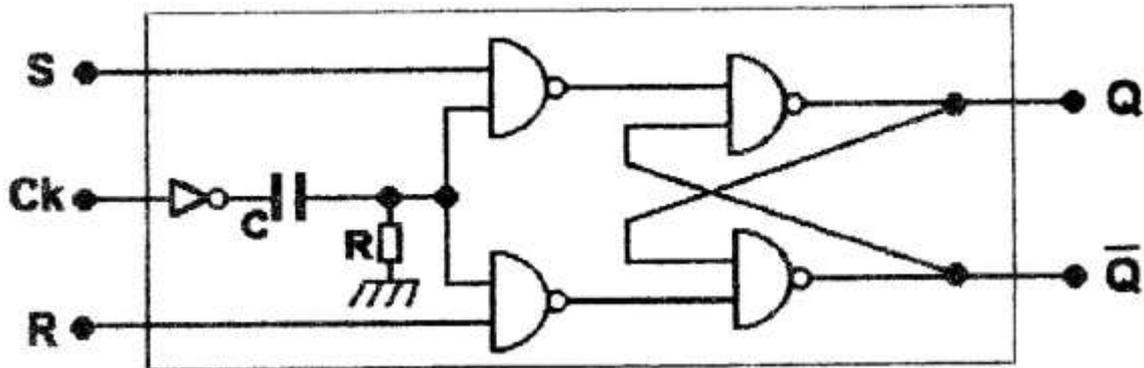
Tabela Verdade

Ck	S	R	Q
↓	0	0	Mantém
↓	0	1	0
↓	1	0	1
↓	1	1	Proibido

Simbologia

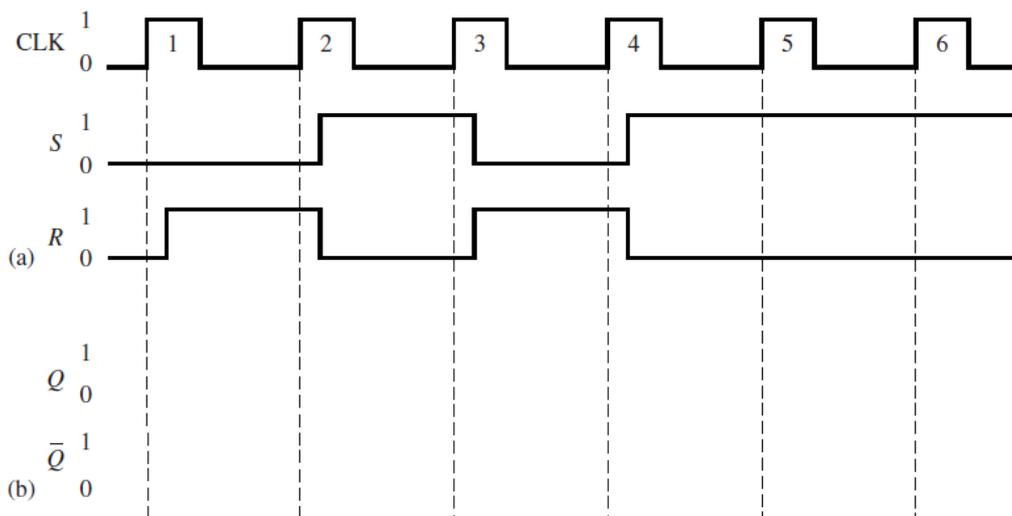
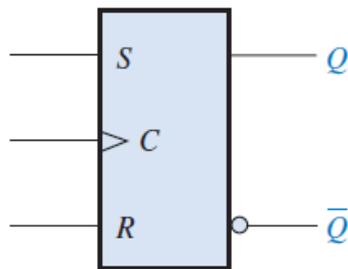


### Circuito Lógico baseado em portas NAND



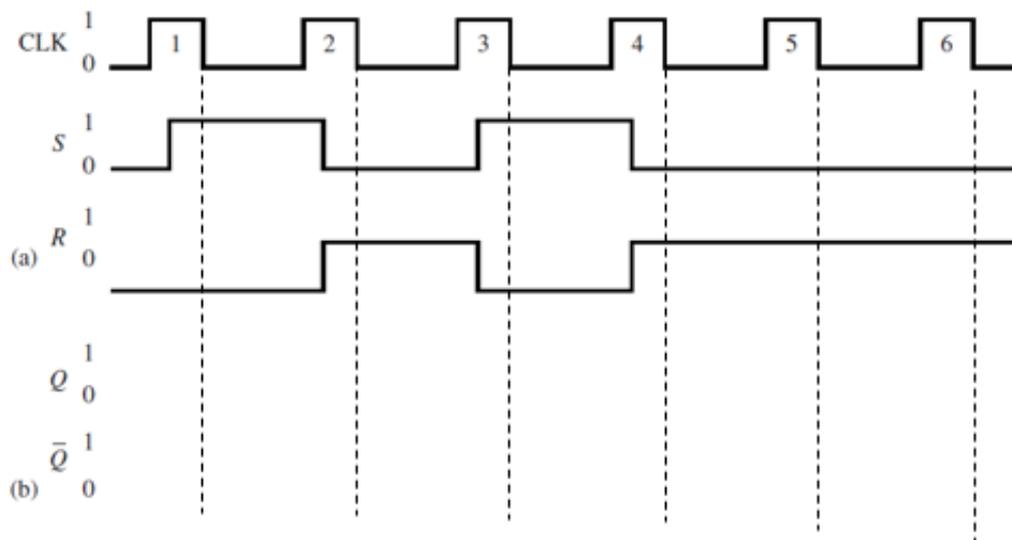
#### EXERCÍCIO TUTORIADO

3. Determine as formas de onda das saídas  $Q$  e  $\bar{Q}$  do flip-flop mostrado abaixo para as entradas  $S$ ,  $R$  e  $CLOCK$  ( $C$ ) mostradas na Figura. Considere que o flip-flop disparado por borda subida e esteja inicialmente resetado (saída  $Q$  em nível BAIXO).



#### EXERCÍCIO COM DISCUSSÃO EM DUPLAS

4. Determine as formas de onda das saídas  $Q$  e  $\bar{Q}$  do flip-flop SR para as entradas  $S$ ,  $R$  e  $CLOCK$  ( $CLK$ ). Considere que o flip-flop disparado por borda de descida e inicialmente resetado.



## APLICABILIDADE DE FLIP-FLOPS

---

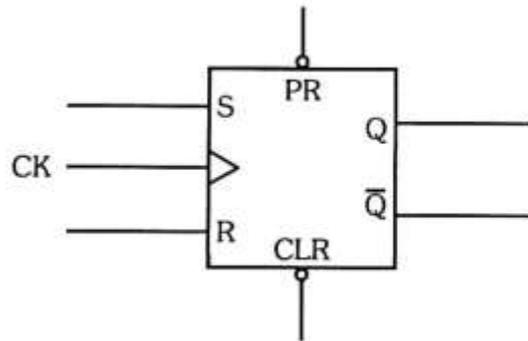
- As memórias semicondutoras nos computadores consistem de muitas células individuais. Cada célula de armazenamento mantém um nível 1 ou um nível 0. Um dos tipos de memória possível é a **Memória de Acesso Aleatório Estática (SRAM)** que usa **flip-flops para as células de armazenamento**, porque um flip-flop retém qualquer um dos dois estados indefinidamente enquanto a alimentação estiver aplicada, daí o termo denominado de memória estática.
- Esse tipo de memória é classificado como **memória volátil**, porque **todos os dados armazenados são perdidos quando a alimentação é desligada**.
- Um outro tipo de memória, a **Memória de Acesso Aleatório Dinâmica ou DRAM**, utiliza **capacitância em vez de flip-flops** como elemento básico de armazenamento e tem que ser **renovada (refresh) periodicamente** para manter o dado armazenado.

## FLIP FLOP TIPO RS: SÍNCRONO BORDA DE SUBIDA COM PRESET E CLEAR

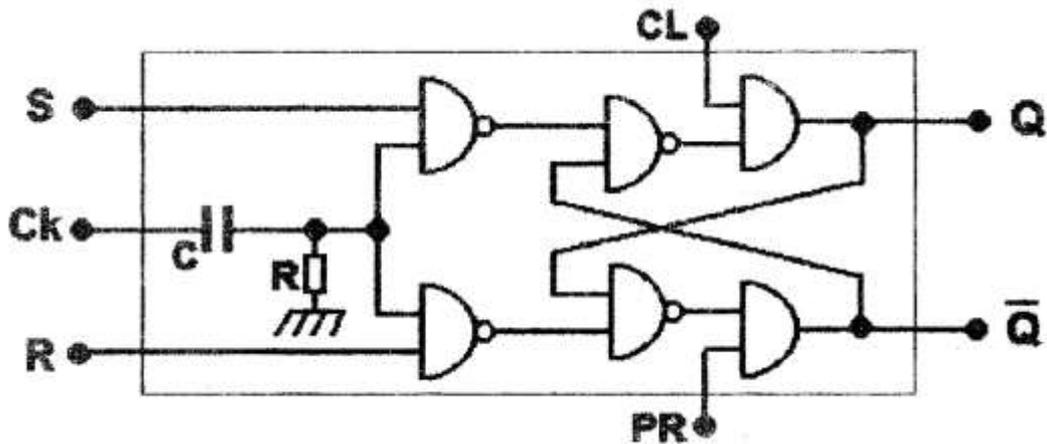
---

- Biestável que contém dois terminais adicionais (**PRESET** e **CLEAR**) que possibilitam ao usuário **forçar a saída ser 0 ou 1 independentemente das condições de entrada e do clock**.
- A função **PRESET** força a saída ser "1" quando **CLEAR = 1 e PRESET = 0**.
- A função **CLEAR** força a saída ser "0" quando **CLEAR = 0 e PRESET = 1**.

### Simbologia



### Circuito Lógico baseado em portas NAND



### TABELA VERDADE COM PRESET E CLEAR

- A tabela verdade neste caso apresenta 5 entradas (PRESET, CLEAR, clock, R e S).
- Quando a saída é forçada através do PRESET ou do CLEAR, independentemente do que se tenha nas entradas S ou R (ou mesmo no clock), a mesma fica inalterada.

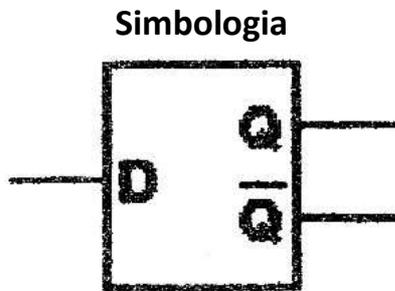
### TABELA VERDADE PARA FLIP-FLOP BORDA DE SUBIDA

PR	CL	CK	R	S	Q
0	0	X	X	X	proibido
0	1	X	X	X	1
1	0	X	X	X	0
1	1	0	X	X	mantém
1	1	1	X	X	mantém
1	1	↓	X	X	mantém
1	1	↑	0	0	mantém
1	1	↑	0	1	1
1	1	↑	1	0	0
1	1	↑	1	1	proibido

## LATCH D

---

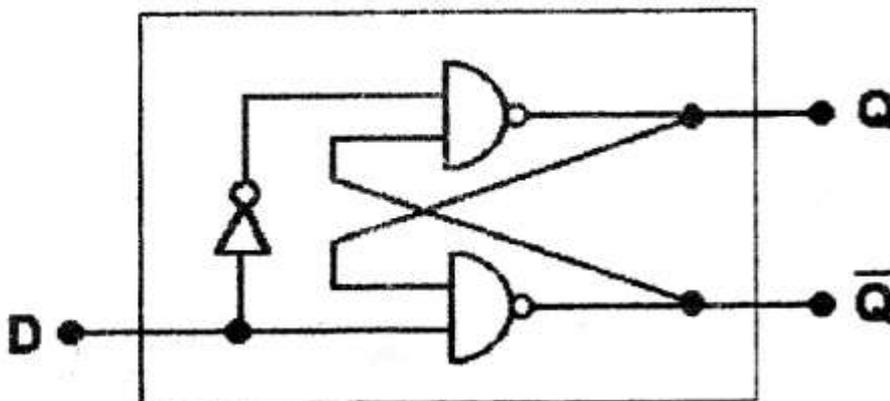
- O sinal que estiver em sua **entrada (D)** é o mesmo que vai para a **saída (Q)**.
- A sigla D vem de Data (dado), termo original em inglês.



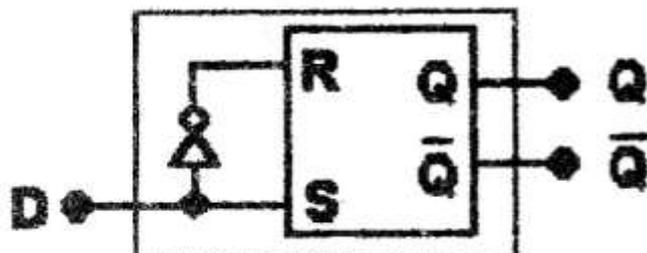
**Tabela Verdade**

D	Q
0	0
1	1

**Circuito baseado em Portas NAND**



- O biestável tipo D pode ser construído através do Latch RS utilizando uma **porta inversora**:



## LATCH D

---

Síncrono (Nível Alto):

Simbologia

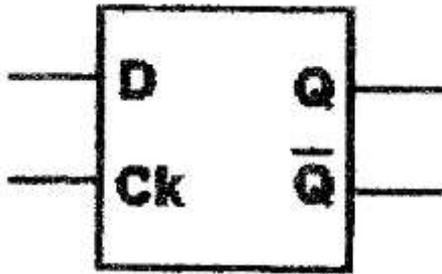
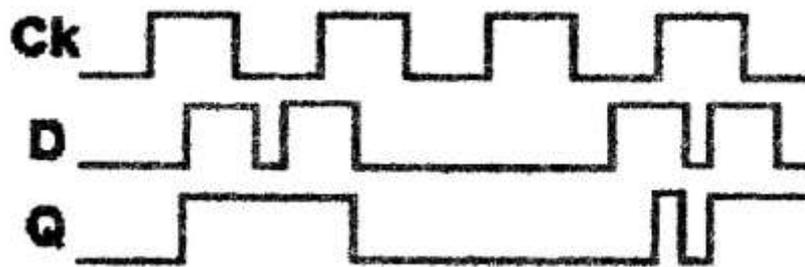


Tabela Verdade

Ck	D	Q
0	X	Mantém
1	0	0
1	1	1

Diagrama de Tempos



## FLIP-FLOP D

---

Síncrono (Borda de Subida)

Simbologia

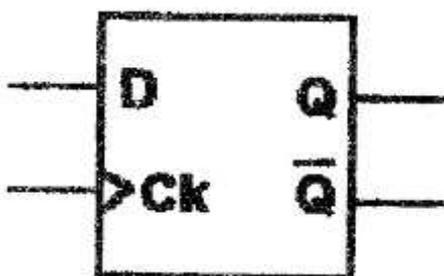
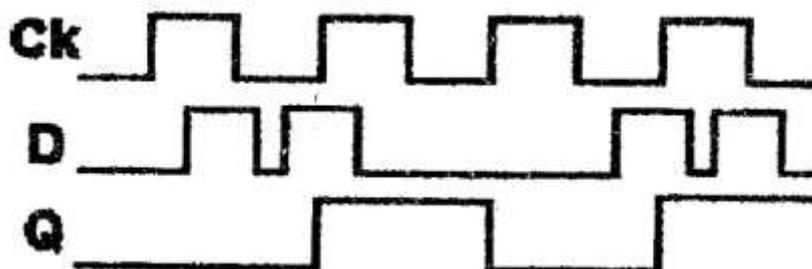


Tabela Verdade

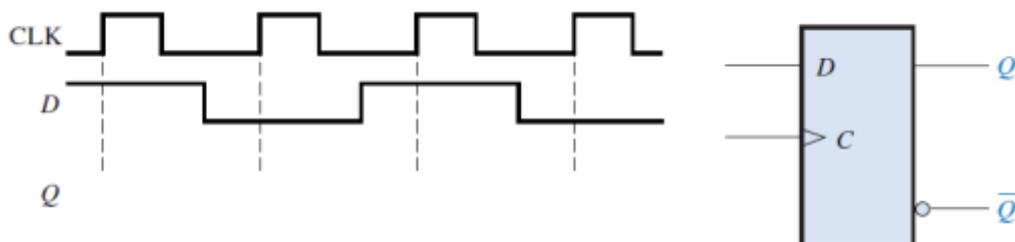
Ck	D	Q
0	X	Mantém
↑	0	0
↑	1	1

Diagrama de Tempos



## EXERCÍCIO COM DISCUSSÃO EM DUPLAS

5. Dadas as formas de onda na figura a seguir para a entrada D e o CLOCK (C), determine a forma de onda na saída Q se o flip-flop iniciar resetado.

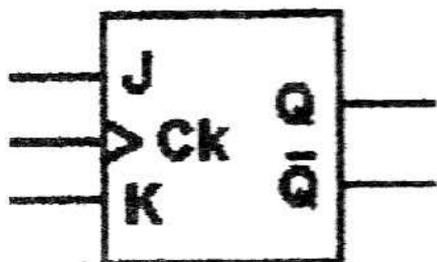


- Existem também os biestáveis Tipo D a nível baixo e borda de descida.

## FLIP-FLOP JK

- O biestável Tipo JK só existe na configuração borda de subida ou descida.
- Não é possível existir assincronamente ou a nível.

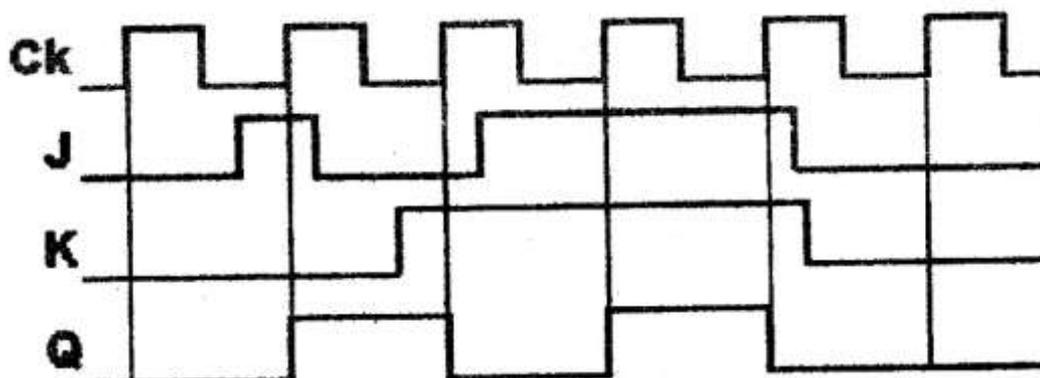
### Simbologia



### Tabela Verdade

Ck	J	K	Q
↑	0	0	Mantém
↑	0	1	0
↑	1	0	1
↑	1	1	Inverte

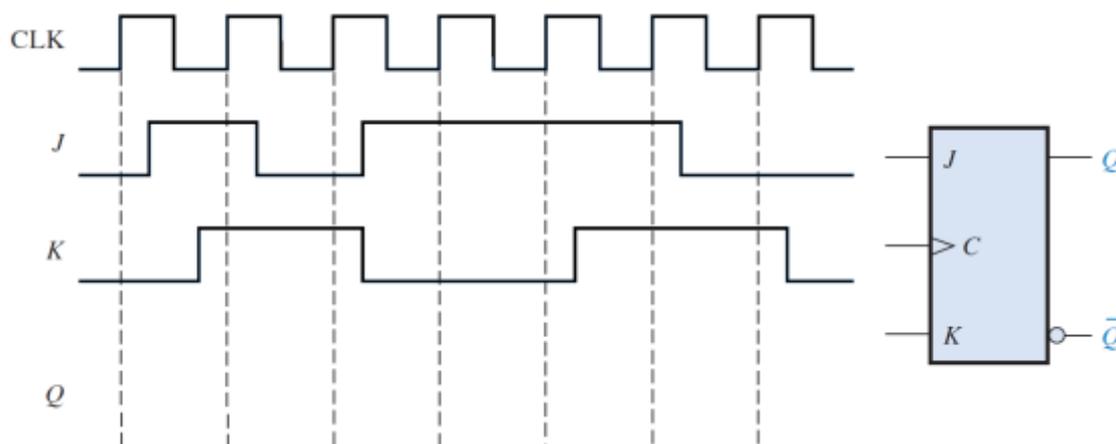
### Diagrama de Tempos



## EXERCÍCIO TUTORIADO

---

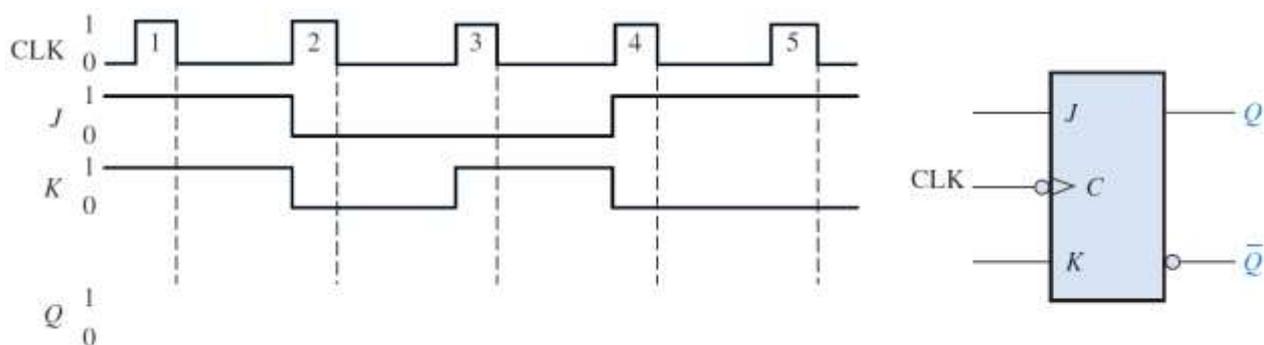
6. As formas de onda mostradas na figura abaixo são aplicadas nas entradas J, K e CLOCK (C) conforme indicado. Determine a saída Q, considerando que o flip-flop esteja inicialmente resetado.



## EXERCÍCIO COM DISCUSSÃO EM DUPLAS

---

7. As formas de onda mostradas na figura abaixo são aplicadas nas entradas J, K e CLOCK (CLK) conforme indicado. Determine a saída Q, considerando que o flip-flop esteja inicialmente resetado.

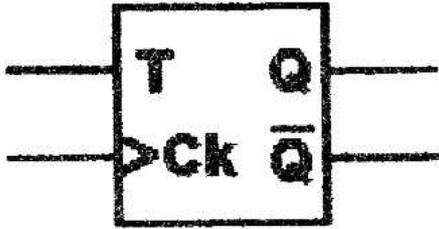


## FLIP-FLOP T

---

- É obtido a partir de um Flip-Flop JK com as entradas J e K curto-circuitadas (uma ligada à outra).
- A sigla T vem de Toggle (comutado).
- O biestável Tipo T **só existe na configuração borda de subida ou descida**.

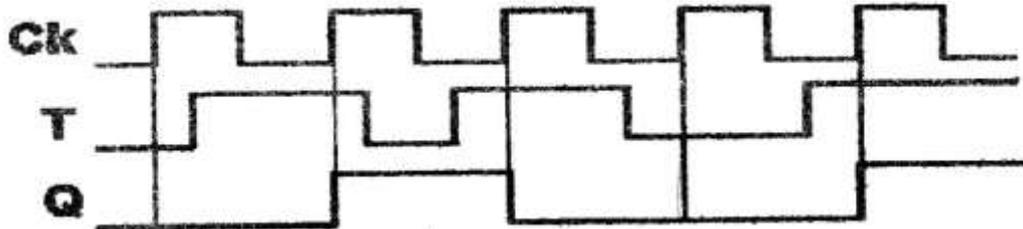
### Simbologia



### Tabela Verdade

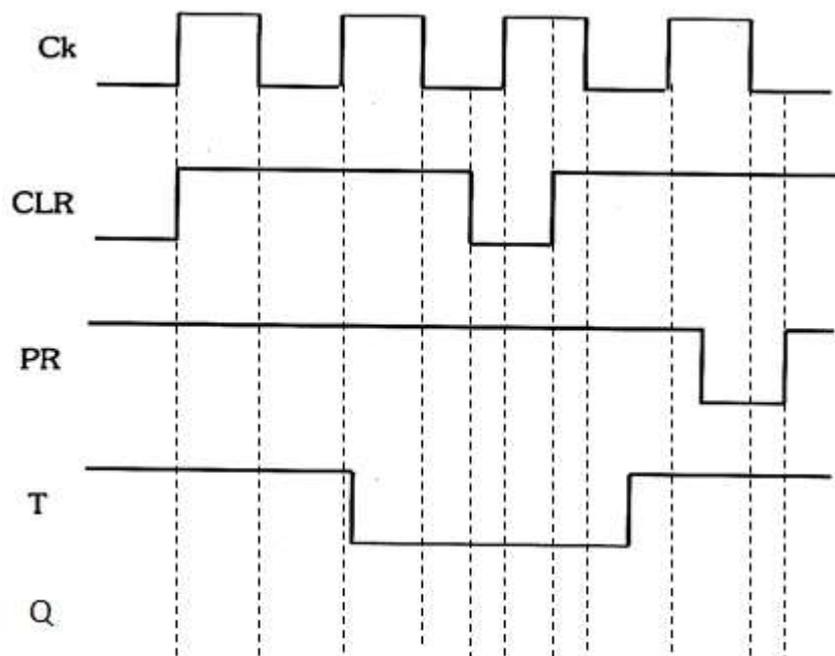
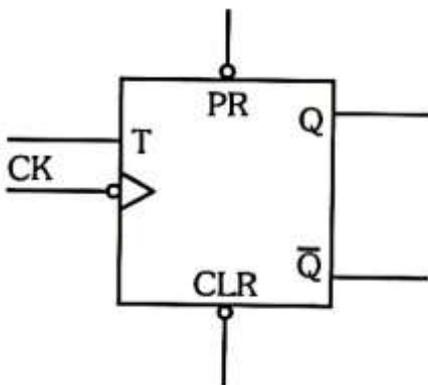
Ck	T	Q
↑	0	mantém
↑	1	inverte

### Diagrama de Tempos



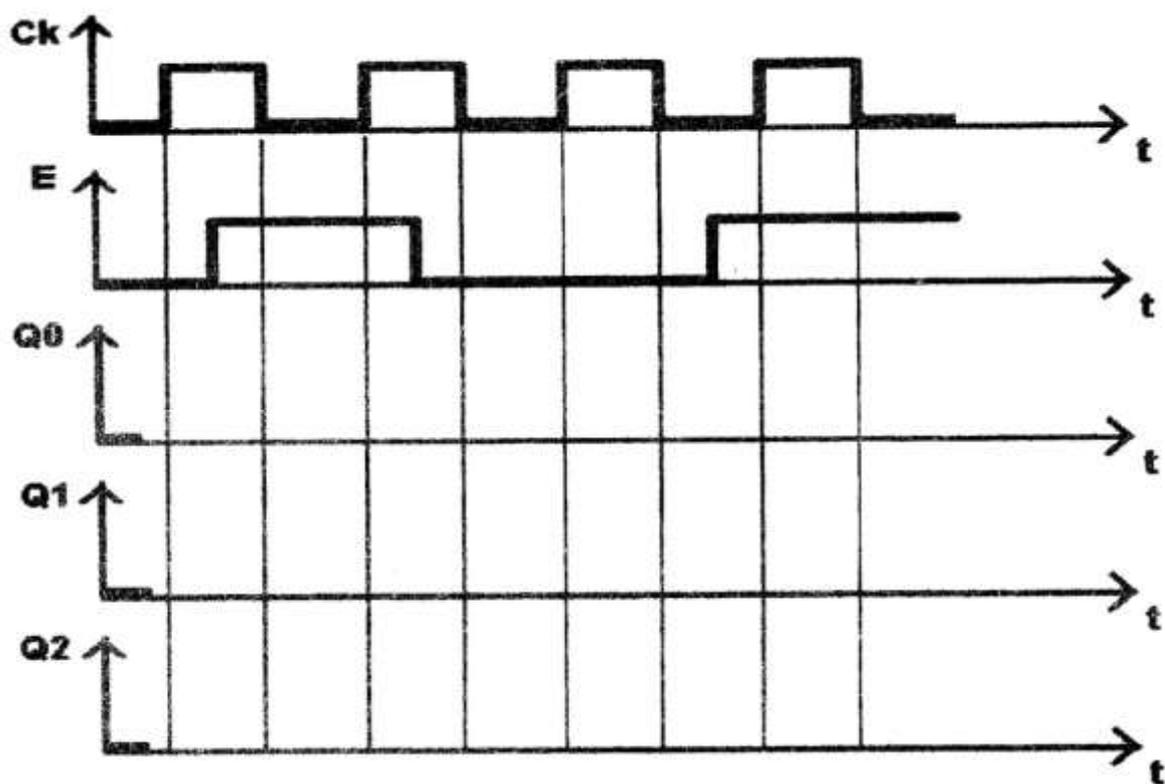
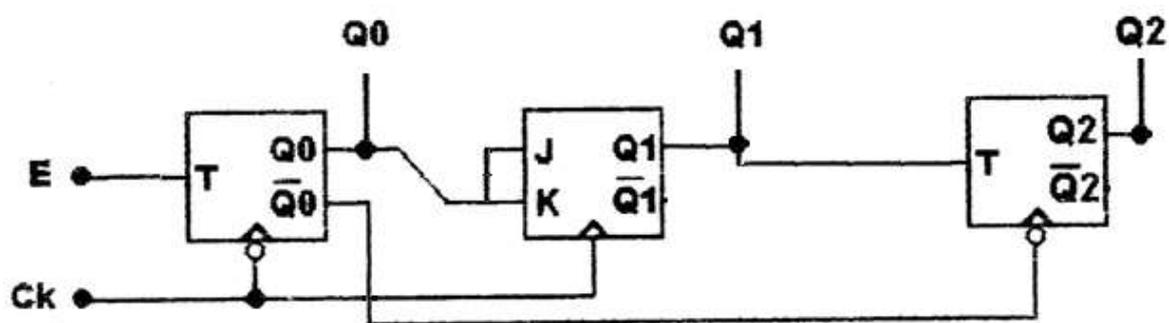
### EXERCÍCIO TUTORIADO

8. Determine a forma de onda da saída Q do flip-flop tipo T visto na figura abaixo, em função dos sinais aplicados nas entradas, considere Q inicialmente em nível baixo:



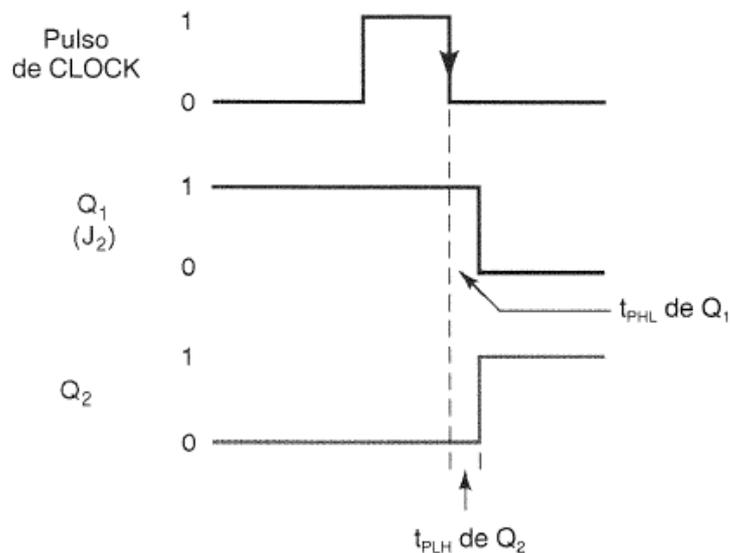
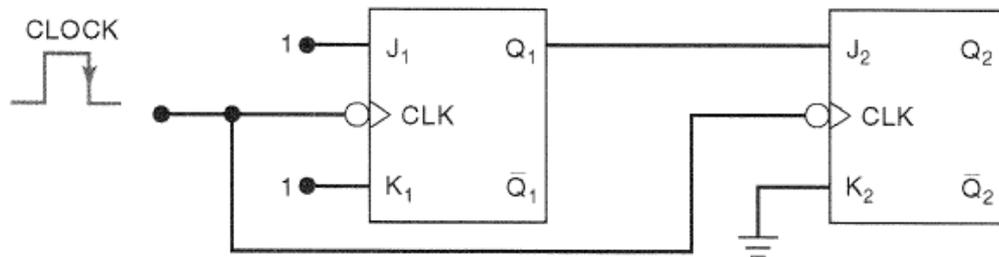
## EXERCÍCIO TUTORIADO

9. Dado o circuito abaixo, desenhe as formas de onda das saídas Q0, Q1 e Q2, considerando que inicialmente estas saídas estão em nível lógico baixo (resetado):



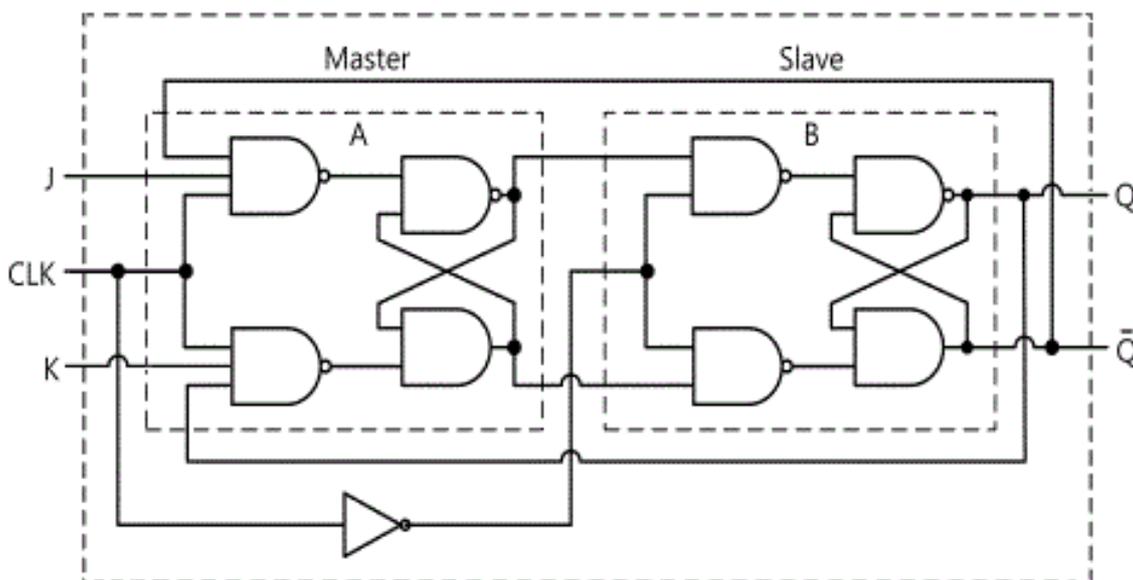
## FLIP-FLOP JK Mestre-Escravo

- Em muitos circuitos digitais a saída de um Flip-Flop é conectada diretamente ou através de portas lógicas a entrada de outro Flip-Flop temporizados pelo mesmo clock, conforme mostrado a seguir.

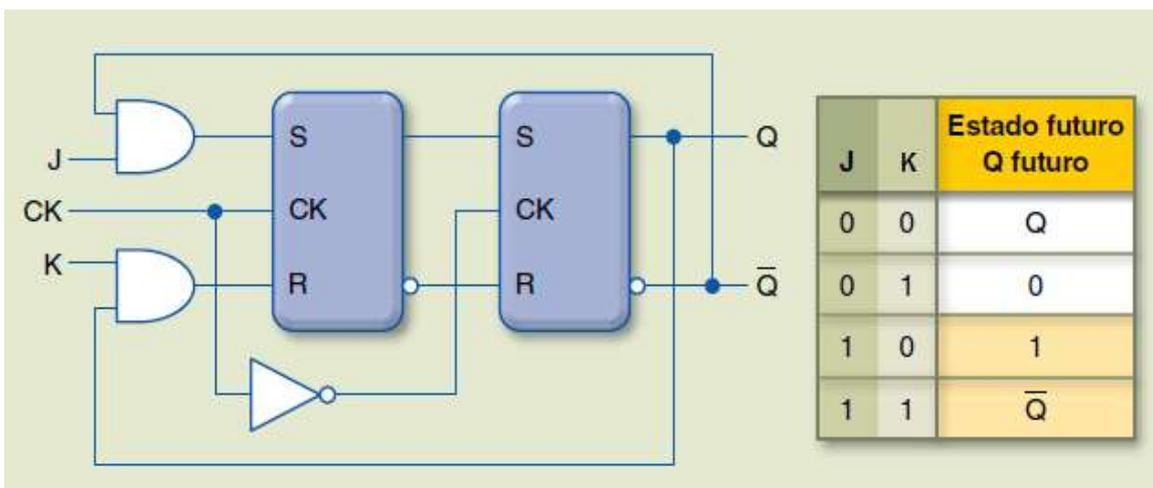


- Observa-se que  $Q_2$  responderá adequadamente ao nível presente em  $Q_1$  antes da descida do clock, desde que o tempo de atraso de  $Q_2$  ( $t_{PLH}$ ) seja menor do que o atraso de propagação de  $Q_1$ .
- Anterior a solução de utilização de Flip-Flops nível a borda com tempo de propagação muito pequenos, era adotado a solução de Flip-flop JK master-slave (mestre-escravo) formado por dois flip-flops RS síncronos que são ligados em cascata com um inversor entre a entrada de clock do primeiro (master ou mestre) e a entrada de clock do segundo (slave ou escravo).
- Possui uma realimentação oriunda das saídas para as entradas das portas lógicas.

### CIRCUITO LÓGICO COM PORTAS



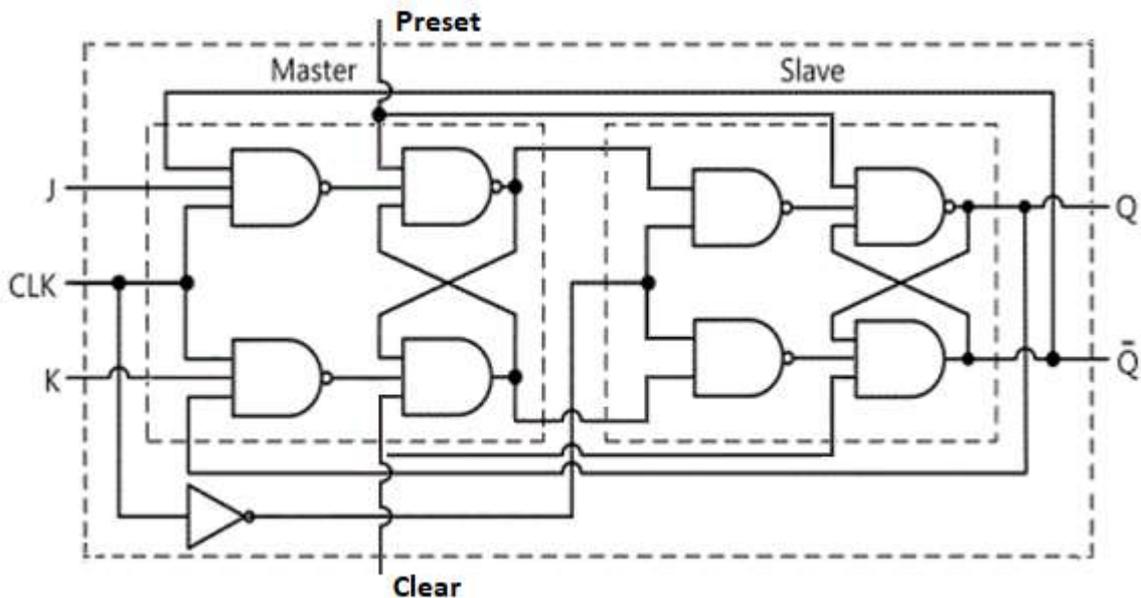
### CIRCUITO LÓGICO COM LATCH SR E PORTAS



- Foi desenvolvido um circuito que conforme é dado o pulso no clock as suas entradas finais (Q e Q') são bloqueadas e a saída só é fornecida quando o pulso deste clock é 0.

## FLIP-FLOP JK Mestre-Escravo com Preset e Clear

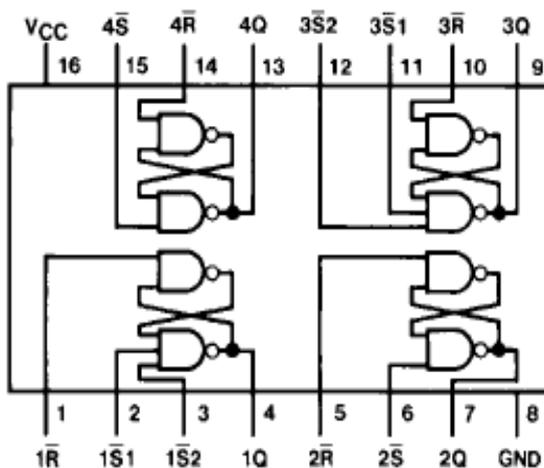
- Flip-flop JK pode assumir valores  $Q = 1$  e  $Q = 0$  mediante a utilização das entradas Preset e Clear apresentas a seguir.



## LATCHES E FLIP FLOP COMERCIAIS

### DM74LS279 QUAD S'R' Latch

#### Connection Diagram



#### Function Table

Inputs		Output
$\bar{S}$ (Note 1)	$\bar{R}$	Q
L	L	H (Note 2)
L	H	H
H	L	L
H	H	$Q_0$

H = HIGH Level

L = LOW Level

$Q_0$  = The Level of Q before the indicated input conditions were established.

Note 1: For latches with double  $\bar{S}$  inputs:

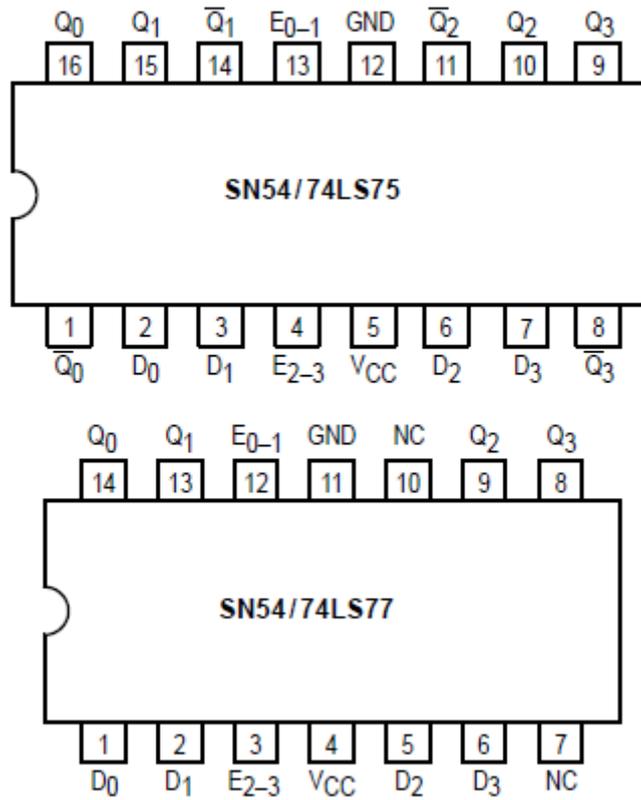
H = both  $\bar{S}$  inputs HIGH

L = one or both  $\bar{S}$  inputs LOW

Note 2: This output level is pseudo stable; that is, it may not persist when the  $\bar{S}$  and  $\bar{R}$  inputs return to their inactive (HIGH) level.

## 74LS75/74LS77 4 BIT D Latch

### CONNECTION DIAGRAMS DIP (TOP VIEW)



### PIN NAMES

D <sub>1</sub> -D <sub>4</sub>	Data Inputs
E <sub>0-1</sub>	Enable Input Latches 0, 1
E <sub>2-3</sub>	Enable Input Latches 2, 3
Q <sub>1</sub> -Q <sub>4</sub>	Latch Outputs (Note b)
$\overline{Q_1}$ - $\overline{Q_4}$	Complimentary Latch Outputs (Note b)

### NOTES:

- a) 1 Unit Load (U.L.) = 40  $\mu$ A HIGH.  
 b) The Output LOW drive factor is 2.5 U.L. for Military (54) and 5 U.L. for Commercial (74) Temperature Ranges.

### TRUTH TABLE (Each latch)

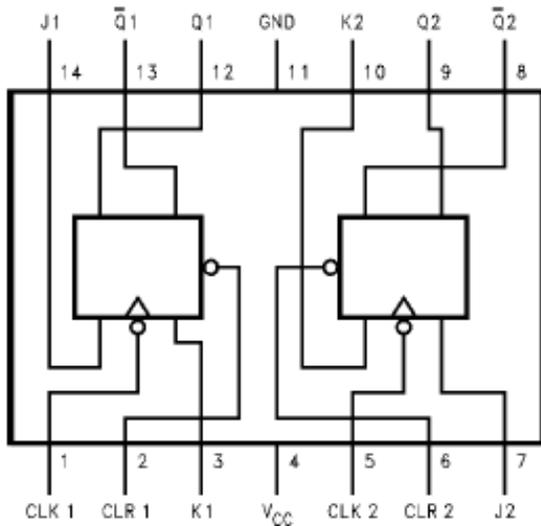
$t_n$	$t_{n+1}$
D	Q
H	H
L	L

### NOTES:

- $t_n$  = bit time before enable negative-going transition  
 $t_{n+1}$  = bit time after enable negative-going transition

# DM7473 DUAL Master-Slave J-K Flip-Flops

## Connection Diagram



## Function Table

Inputs				Outputs	
CLR	CLK	J	K	Q	$\bar{Q}$
L	X	X	X	L	H
H		L	L	$Q_0$	$\bar{Q}_0$
H		H	L	H	L
H		L	H	L	H
H		H	H	Toggle	

H = HIGH Logic Level

L = LOW Logic Level

X = Either LOW or HIGH Logic Level

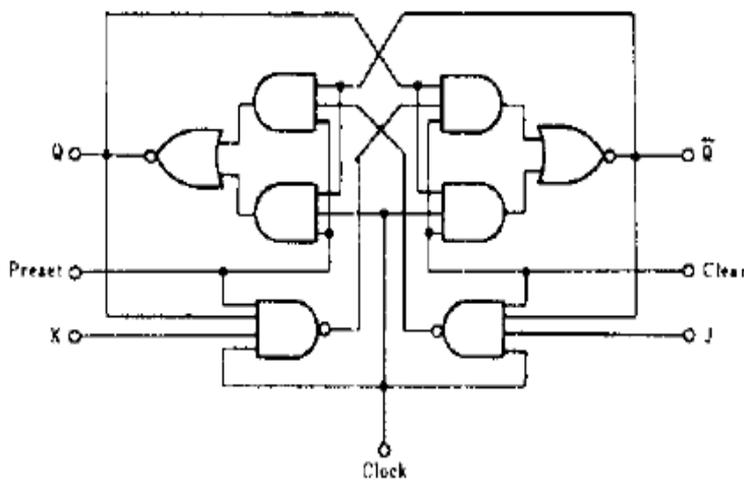
= Positive pulse data. the J and K inputs must be held constant while the clock is HIGH. Data is transferred to the outputs on the falling edge of the clock pulse.

$Q_0$  = The output logic level before the indicated input conditions were established.

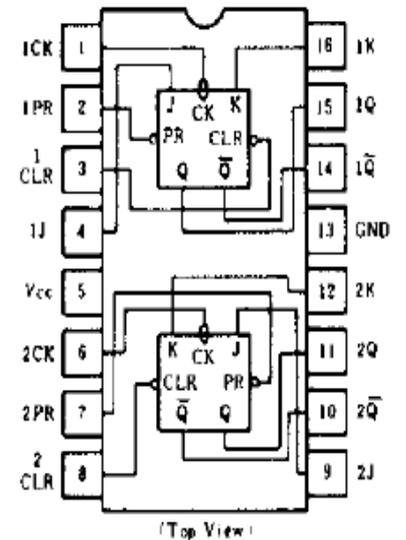
Toggle = Each output changes to the complement of its previous level on each HIGH level clock pulse.

# HD74LS76A Dual J-K Flip-Flops (with Preset and Clear)

## BLOCK DIAGRAM(1/2)



## PIN ARRANGEMENT

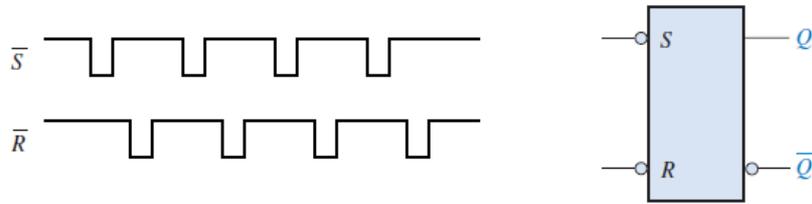


## FUNCTION TABLE

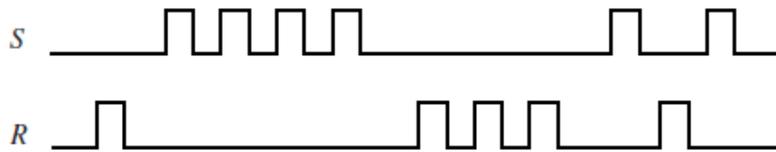
Inputs					Outputs	
Preset	Clear	Clock	J	K	Q	$\bar{Q}$
L	H	×	×	×	H	L
H	L	×	×	×	L	H
L	L	×	×	×	H*	H*
H	H	↓	L	L	Q <sub>0</sub>	$\bar{Q}_0$
H	H	↓	H	L	H	L
H	H	↓	L	H	L	H
H	H	↓	H	H	Toggle	
H	H	H	×	×	Q <sub>0</sub>	$\bar{Q}_0$

## EXERCÍCIOS EXTRA-CLASSE

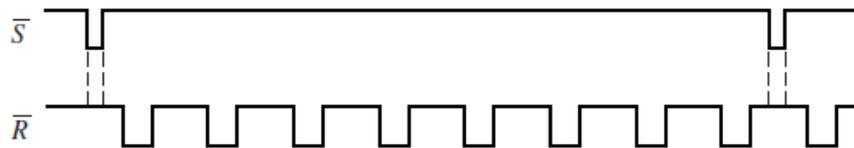
- Se as formas de onda a seguir são aplicadas no *latch* com entradas ativas em nível BAIXO, desene a forma de onda da saída Q resultante em relação às entradas. Considere a saída Q iniciando em nível BAIXO.



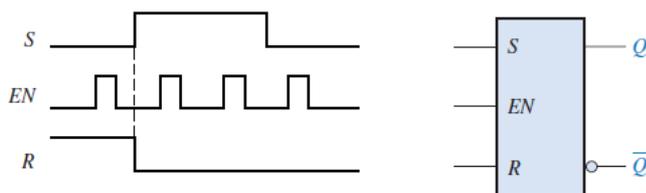
- Se as formas de onda a seguir são aplicadas no *latch* com entradas ativas em nível BAIXO, desene a forma de onda da saída Q resultante em relação às entradas. Considere a saída Q iniciando em nível ALTO.



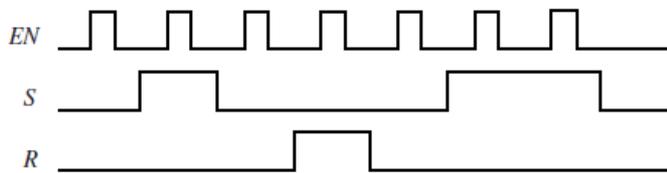
- Se as formas de onda a seguir são aplicadas no *latch* com entradas ativas em nível BAIXO, desene a forma de onda da saída Q resultante em relação às entradas. Considere a saída Q iniciando em nível BAIXO.



- Para um *latch* SR controlado, determine as saídas  $Q$  e  $\bar{Q}$  para as entradas dadas nas figuras abaixo. Mostre-as em relação à entrada de habilitação. Considere a saída Q iniciando em nível BAIXO.



a.



b.

