

# Sequencial

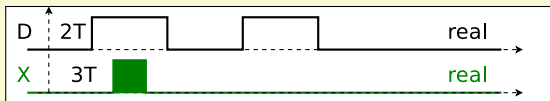
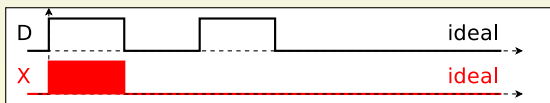
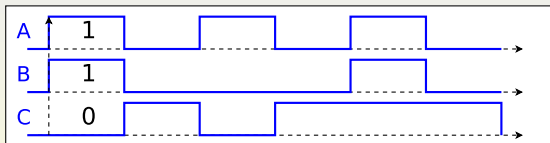
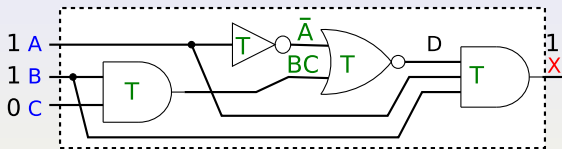
## Contadores

Fernando Pujaico Rivera<sup>1</sup>

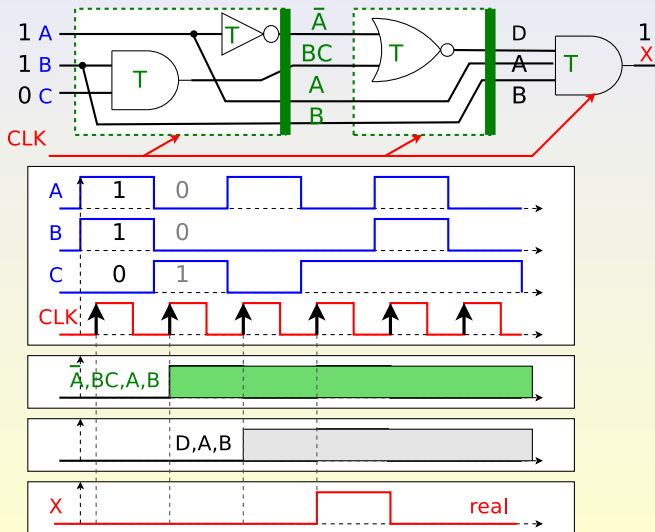
<sup>1</sup>Universidade Federal de Lavras

Aula-1 2016

# Lógica combinacional



# Lógica secuencial

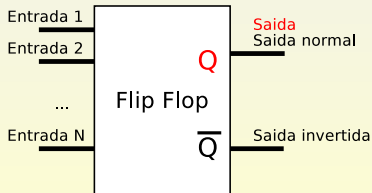


# Lógica sequencial

**Circuito combinacional** a saída depende exclusivamente da **combinação de suas entradas**.

**Circuito sequencial** a saída futura depende de sua **saída atual** e da **combinação de suas entradas** (Armazena valores antigos).

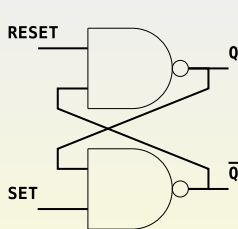
O elemento de memória mais importante é o Flip-Flop.

**Estados:**

$Q=1$ logo $\bar{Q}=0$
$Q=0$ logo $\bar{Q}=1$

# Flip-Flop básico - SR NAND LATCH (SET-RESET)

Usando portas NAND



Q	RESET	SET
X	0	0
X	0	1
X	1	0
X	1	1

**INVÁLIDO:  $Q = \bar{Q} = 1$**



Q	$\bar{Q}$
1	1
1	0
0	1
X	$\bar{X}$

**INVÁLIDO**

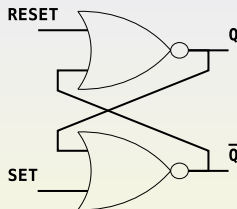
set

reset

não muda

# Flip-Flop básico - SR NOR LATCH (SET-RESET)

Usando portas NOR

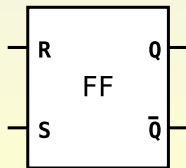


Q	RESET	SET
X	0	0
X	0	1
X	1	0
X	1	1

Q	$\bar{Q}$
X	$\bar{X}$
1	0
0	1
0	0

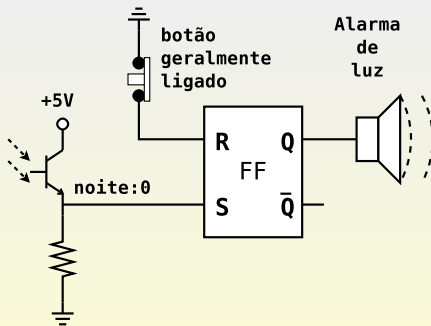
não muda  
set  
reset  
**INVÁLIDO**

**INVÁLIDO:  $Q=\bar{Q}=0$**



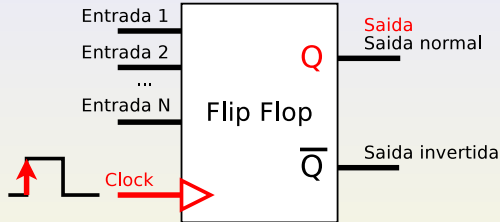
# Flip-Flop básico - SR NOR LATCH (SET-RESET)

Usando portas NOR



Atual estável	Acontece		Futuro estável	
	RESET	SET	Q	
Q			Q	
X	0	0	X	não muda
X	0	1	1	set
X	1	0	0	reset
X	1	1	0/0	INVÁLIDO

# Flip-Flop com CLOCK

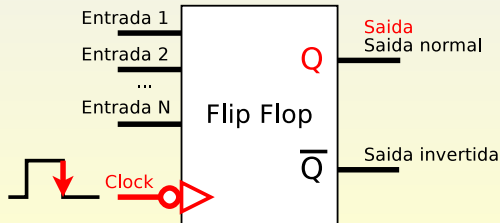


## **Estados:**

$Q=1$  logo  $\bar{Q}=0$

$Q=0$  logo  $\bar{Q}=1$

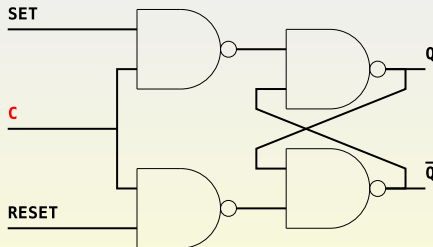
***Os estados  
só  
mudam quando  
há um  
evento de  
clock***





# Flip-Flop SR NAND com CLOCK

Usando portas NAND



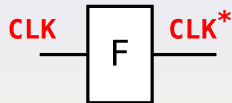
Acontece
C
RESET
SET

C	RESET	SET
0	X	X
1	0	0
1	0	1
1	1	0
1	1	1

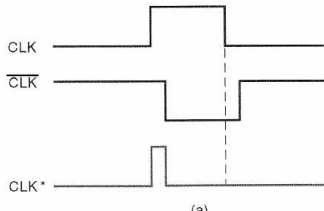
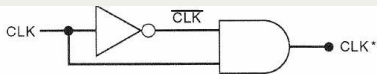
Futuro estável
Q
Q
Q
1
0
1/1

não muda  
não muda  
set  
reset  
**INVÁLIDO**

# Detetor de transição

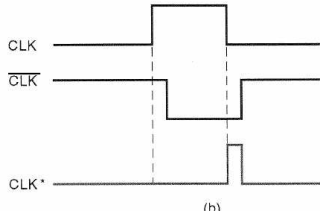
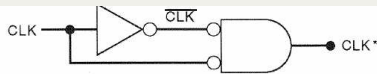


Transição positiva



(a)

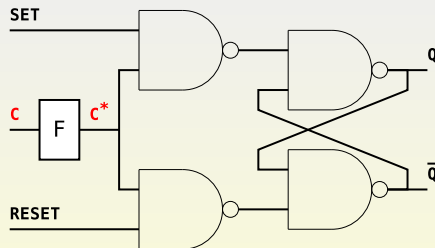
Transição negativa



(b)

# Flip-Flop SR NAND com CLOCK

Usando portas NAND



Acontece

C	RESET	SET
outro	X	X
↑	0	0
↑	0	1
↑	1	0
↑	1	1

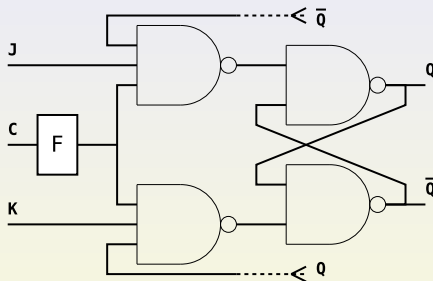
Futuro estável

Q
Q
Q
1
0
1/1

não muda  
não muda  
set  
reset  
**INVÁLIDO**

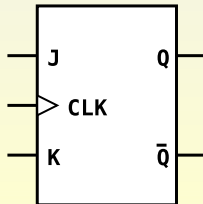
# Flip-Flop JK

Usando portas NAND

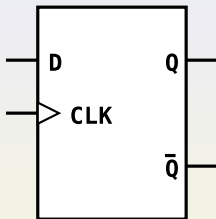


Acontece		
C	K	J
outro	X	X
0	0	0
0	0	1
0	1	0
0	1	1

Futuro estável	
Q	
Q	não muda
Q	não muda
1	set
0	reset
$\bar{Q}$	comuta



# Flip-Flop D



Acontece	
C	D
outro	X
	1
	0

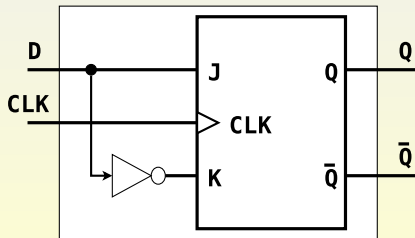
Futuro estável	
Q	
Q	não muda
1	set
0	reset

não muda  
set  
reset

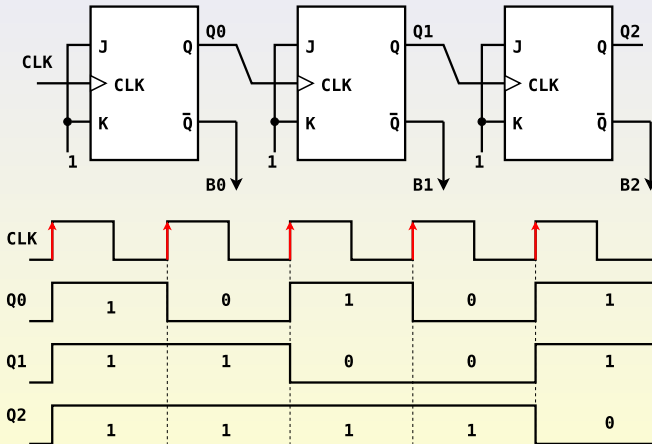
Acontece		
C	K	J
outro	X	X
	0	1
	1	0

Futuro estável	
Q	
Q	não muda
1	set
0	reset

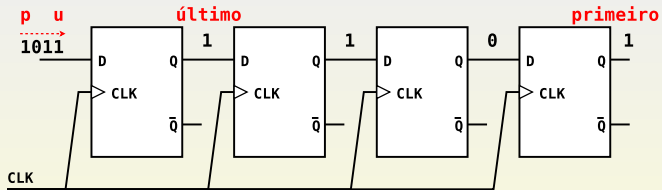
não muda  
set  
reset



# Exemplo Flip-Flop JK - contador assíncrono



# Exemplo Flip-Flop JK - contador assíncrono



# References I