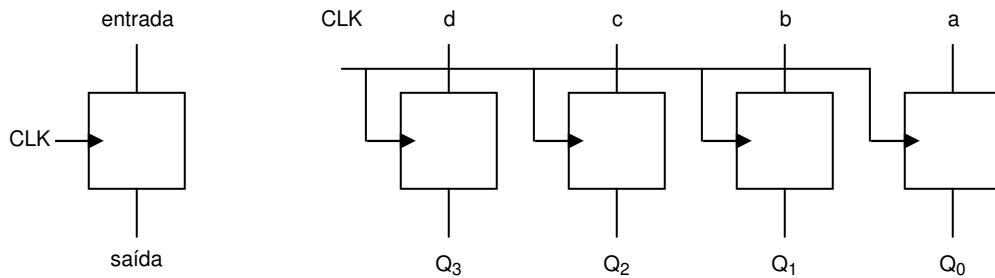


## Registradores

Um registrador é um arranjo de elementos lógicos capaz de armazenar uma palavra binária de ( N ) **bits**.

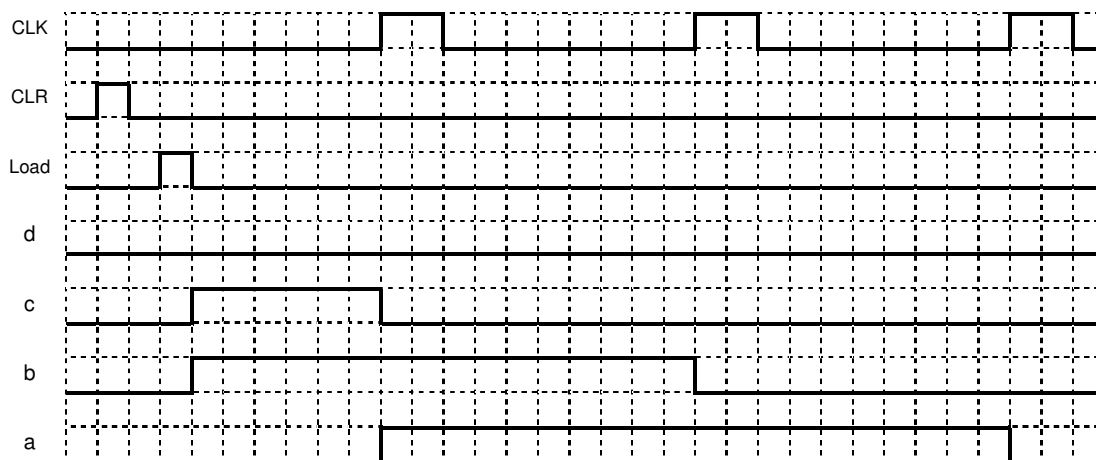
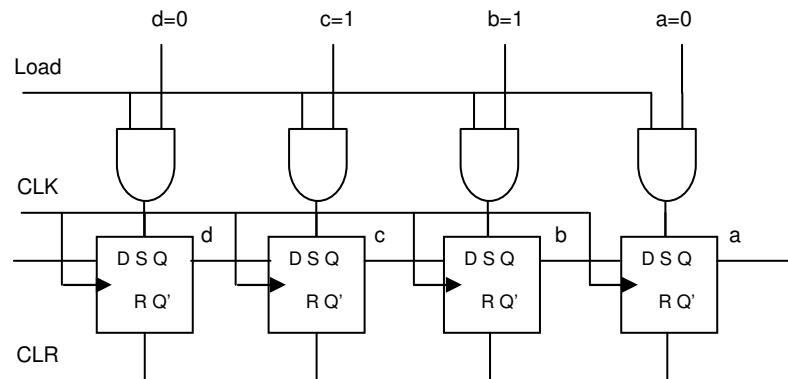
### Registradores para armazenamento

Um registrador para armazenamento é construído com células sensíveis à transição utilizadas em conjunto. O esquema abaixo sugere um arranjo com quatro células.



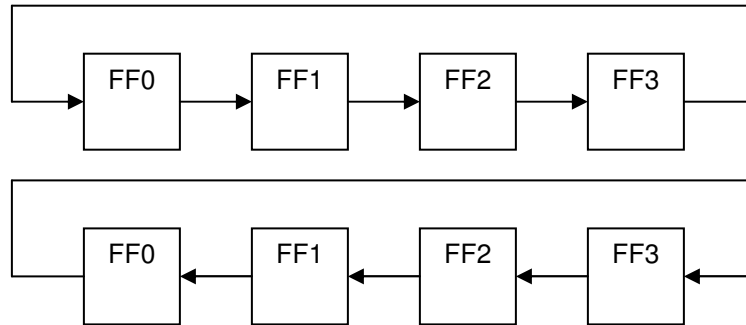
### Conversão paralelo-série

Um arranjo particular de entradas síncronas e assíncronas pode ser construído para realizar a conversão paralelo-série: dados são carregados (**load**) simultaneamente e, em seguida, passados à saída, um de cada vez.

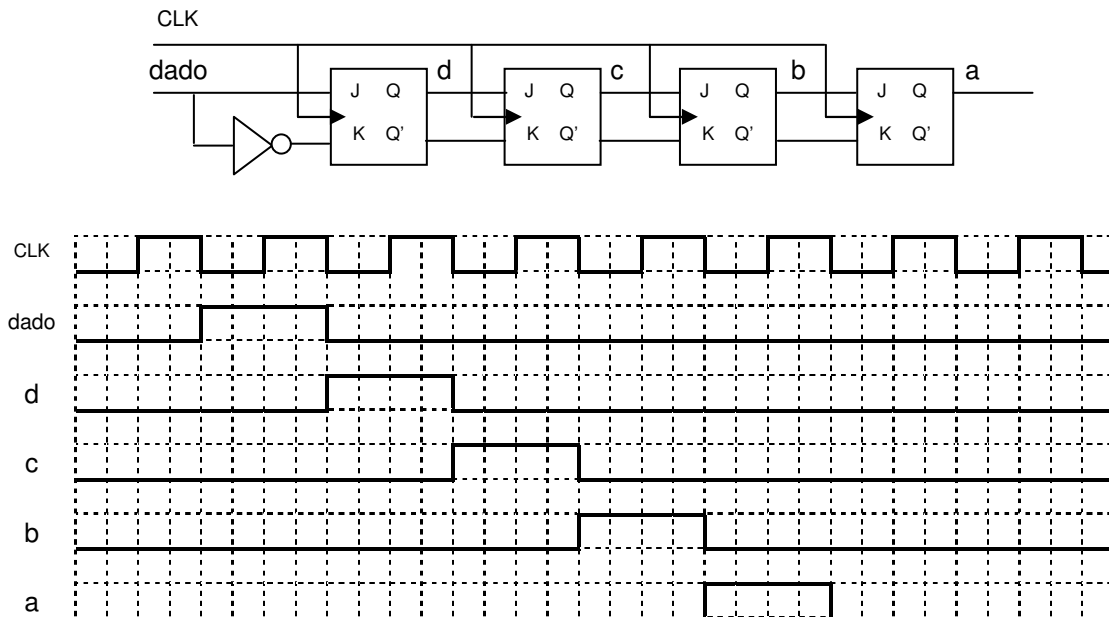


Registadores de deslocamento (**shift register**)

Registadores podem realizar rotações de **bits** para direita ou para esquerda.



Um registrador de deslocamento serve para mover dados para as células vizinhas quando houver variação (pulsos) do **clock**. O esquema abaixo sugere um arranjo com quatro células.



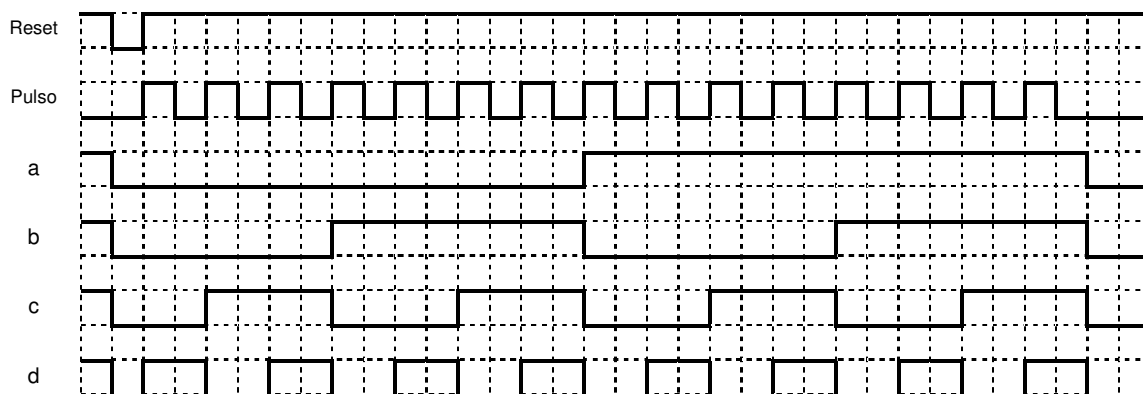
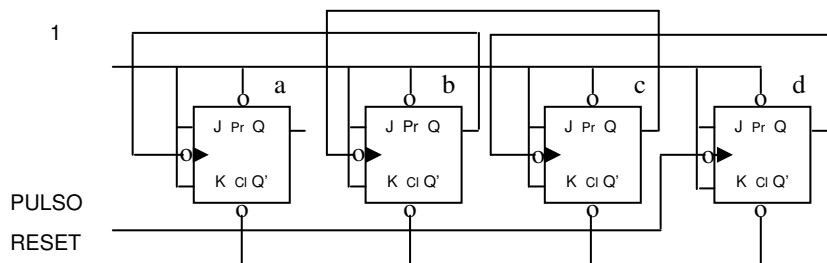
Se as entradas de dados também variarem ao longo do tempo, mais de uma informação estará passando pelo conjunto de registradores em um mesmo instante.

tempo	entradas		d	c	b	a	saídas
0	0 1 1 1						
t	0 1 1	→	1				
2t	0 1	→	1	1			
3t	0	→	1	1	1		
4t			0	1	1	1	
5t				0	1	1	→ 1
6t					0	1	→ 1 1
7t						0	→ 1 1 1
8t							→ 0 1 1 1

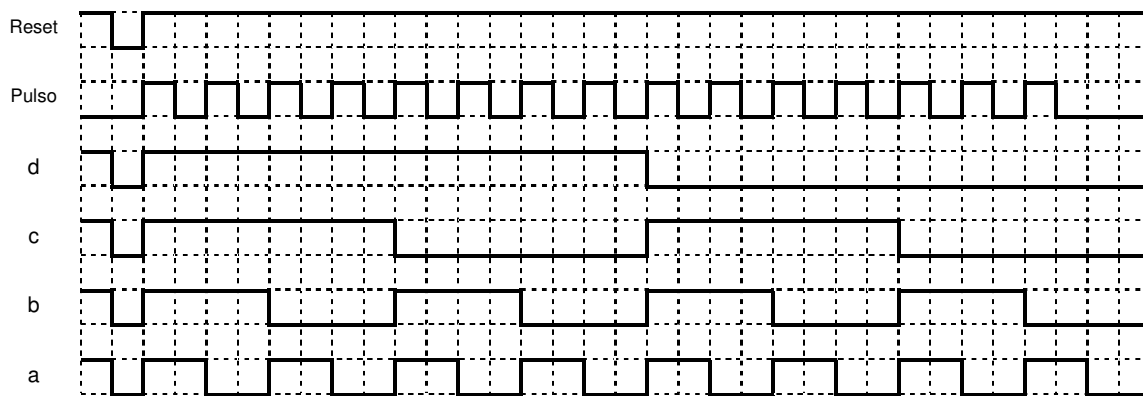
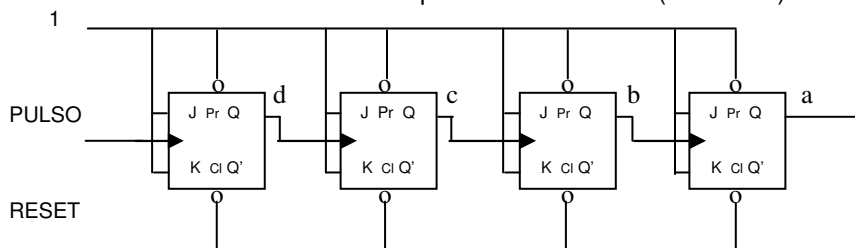
## Contadores de pulsos

Contadores são arranjos de registradores capazes de contar pulsos presentes nas entradas. A saída pode ser vista como um número em binário, crescente ou decrescente.

A seguir, um exemplo de contador de pulsos em ordem crescente (de 0 a 15).

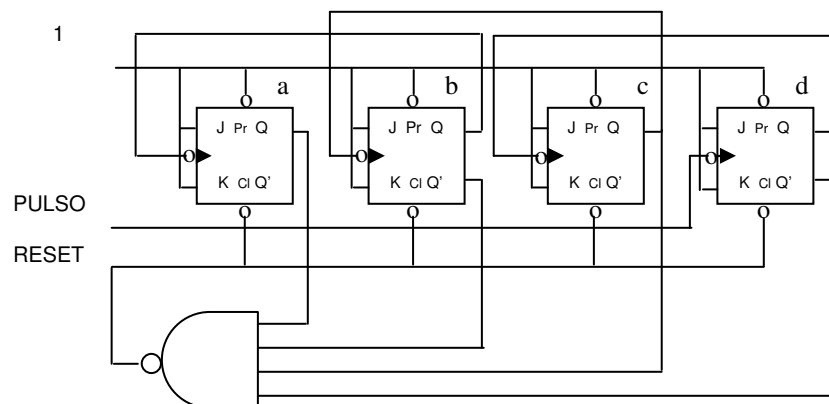


O esquema abaixo ilustra um contador de pulsos decrescente (de 15 a 0).

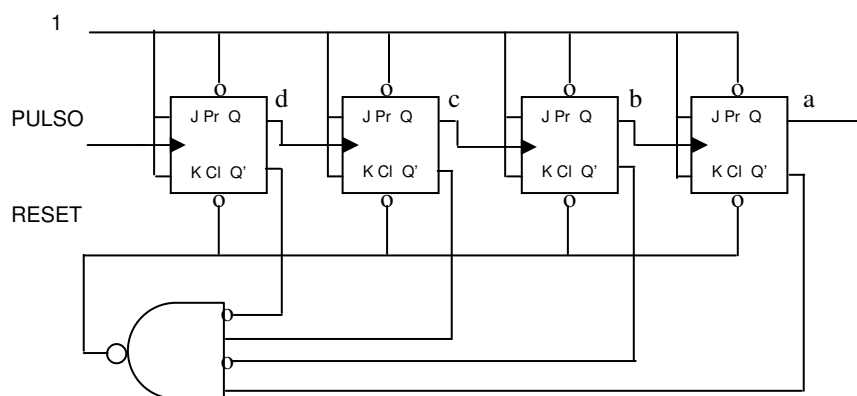


## Contadores decádicos

Contadores *decádicos* (ou em *décadas*) são contadores que voltarão a zero (**reset**) quando o valor das entradas for igual a dez pulsos (ou 1010, em binário).



Esses contadores também podem contar dez pulsos em ordem decrescente. Nesse caso, a contagem deverá reiniciar o contador quando o valor equivalente a cinco (ou 0101, em binário).



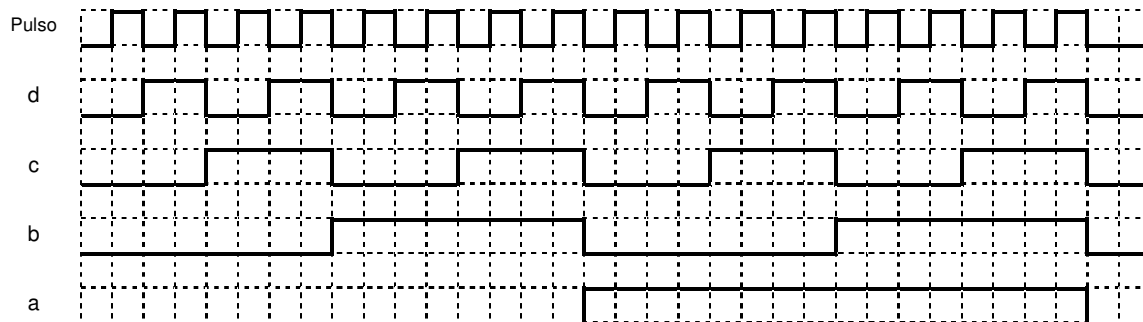
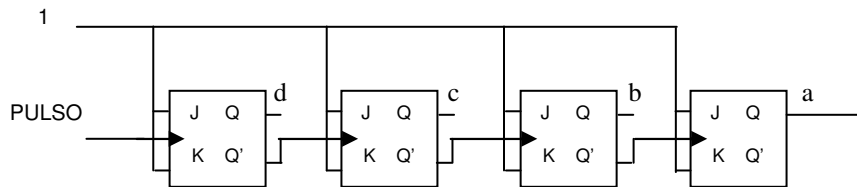
### Contadores assíncronos

Contadores assíncronos são circuitos lógicos cujas **flip-flops** não atuam nas mesmas frequências de **clock**.

Exemplo:

Supor o arranjo de quatro **flip-flops** tipo JK mestre-escravo em cascata.

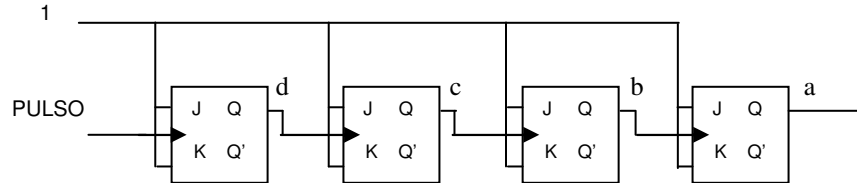
Como esse tipo de **flip-flop** só mudará de estado na descida do **clock**, o resultado será uma seqüência de pulsos com o dobro da largura inicial, ou metade da frequência do circuito anterior.



Pulso	a	b	c	d
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
16	0	0	0	0

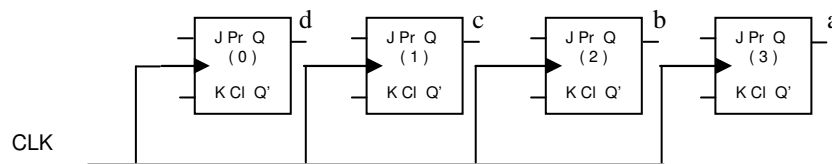
### Contadores assíncronos decrescentes

Contadores assíncronos decrescentes são circuitos cujas saídas são os complementos das saídas de contadores assíncronos, como mostrado anteriormente. Isso pode ser feito tomando os valores das saídas complementadas ( $Q'$ ) ou invertendo-se os sinais de **clock**.



### Contadores síncronos

Em frequências mais altas, os contadores assíncronos podem apresentar pequenas variações nos tempos de resposta e, com isso, gerar erros. Nos contadores síncronos isso pode ser minimizado, pois todos os blocos funcionarão na mesma frequência.



### Contadores síncronos decádicos

Para projetar um contador síncrono decádico, será preciso voltar à tabela característica de um **flip-flop** JK e reorganizá-la em uma *tabela de transições*, como se mostrará abaixo.

Caso	J	K	$Q_{t+1}$	Casos	$Q_t$	$Q_{t+1}$	J	K
A	0	0	$Q_t$	A e B	0	0	0	X
B	0	1	0	C e D	0	1	1	X
C	1	0	1	B e D	1	0	X	1
D	1	1	$Q_t'$	A e C	1	1	X	0

Os valores representados por ( X ) indicam que o valor (0 ou 1) não importa (**don't care**).

Pulso	a	b	c	d	J3	K3	J2	K2	J1	K1	J0	K0
1	0	0	0	0	0	X	0	X	0	X	1	X
2	0	0	0	1	0	X	0	X	1	X	X	1
3	0	0	1	0	0	X	0	X	X	0	1	X
4	0	0	1	1	0	X	1	X	X	1	X	1
5	0	1	0	0	0	X	X	0	0	X	1	X
6	0	1	0	1	0	X	X	0	1	X	X	1
7	0	1	1	0	0	X	X	0	X	0	1	X
8	0	1	1	1	1	X	X	1	X	1	X	1
9	1	0	0	0	X	0	0	X	0	X	1	X
10	1	0	0	1	X	1	0	X	0	X	X	1
11	0	0	0	0								

J3

	c'	c'	c	c	
a'	0	0	0	0	b'
a'	0	0	1	0	b
a	X	X	X	X	b
a	X	X	X	X	b'
	d'	d	d	d'	

K3

	c'	c'	c	c	
a'	X	X	X	X	b'
a'	X	X	X	X	b
a	X	X	X	X	b
a	0	1	X	X	b'
	d'	d	d	d'	

J2

	c'	c'	c	c	
a'	0	0	1	0	b'
a'	X	X	X	X	b
a	X	X	X	X	b
a	0	0	X	X	b'
	d'	d	d	d'	

K2

	c'	c'	c	c	
a'	X	X	X	X	b'
a'	0	0	1	0	b
a	X	X	X	X	b
a	X	X	X	X	b'
	d'	d	d	d'	

J1

	c'	c'	c	c	
a'	0	1	X	X	b'
a'	0	1	X	X	b
a	X	X	X	X	b
a	0	0	X	X	b'
	d'	d	d	d'	

K1

	c'	c'	c	c	
a'	X	X	1	0	b'
a'	X	X	1	0	b
a	X	X	X	X	b
a	X	X	X	X	b'
	d'	d	d	d'	

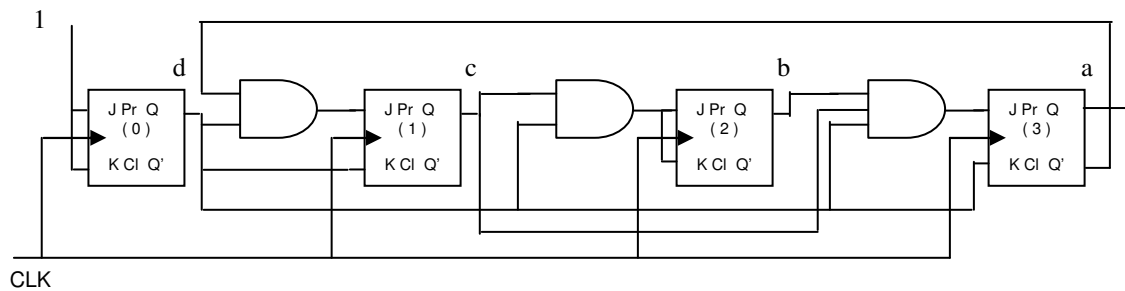
J0

	c'	c'	c	c	
a'	1	X	X	1	b'
a'	1	X	X	1	b
a	X	X	X	X	b
a	1	X	X	X	b'
	d'	d	d	d'	

K0

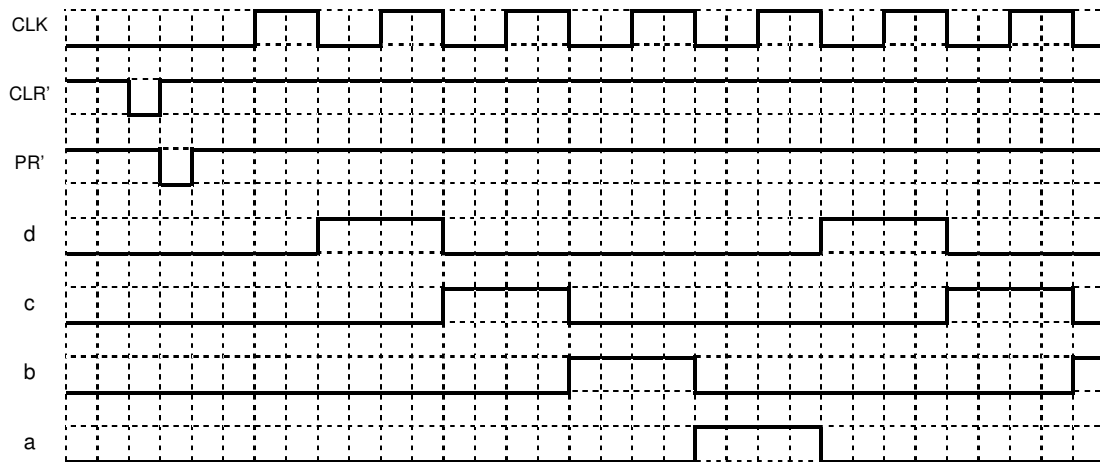
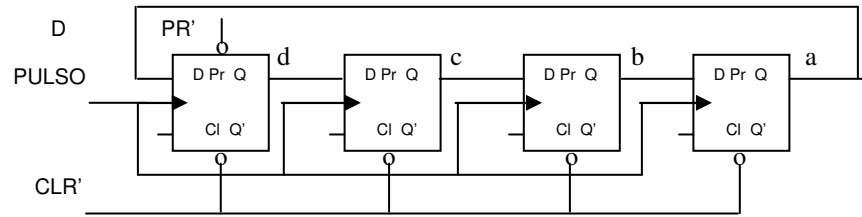
	c'	c'	c	c	
a'	X	1	1	X	b'
a'	X	1	1	X	b
a	X	X	X	X	b
a	X	1	X	X	b'
	d'	d	d	d'	

O circuito mostrado abaixo implementa essas simplificações.



## Contadores em anel

O contador em anel uma vez ativado deslocará um **bit** por vez e será reinicializado ao final da sequência de **bits**.



O contador em anel torcido fará o mesmo, mas não necessitará de um sinal de **preset**.

