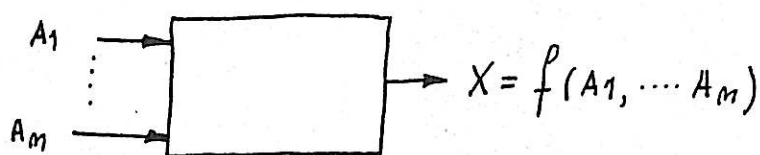


1.-ELEMENTOS DE MEMORIA - BIESTAVEIS

No circuitos combinacionais era possível definir o estado da função de saída de um bloco lógico a partir exclusivamente das diferentes combinações de estados das variáveis de entrada.



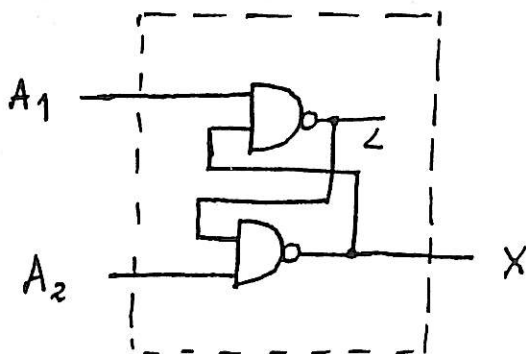
Se, pelo contrário, a função de saída depender não só da combinação das variáveis de entrada mas também do passado do sistema, ou seja, estados anteriores, estamos em presença de um circuito dito sequencial.

Podemos então dizer que:

$$X = f(A_1, \dots, A_n, L)$$

em que L representa o chamado estado interno do sistema, o qual, sendo função dos estados anteriores, vai fazer com que a saída seja função destes e das variáveis de entrada presentes. Logo, a esta noção de estado interno está associado o conceito de memória.

Exemplifiquemos com o seguinte circuito



$$X = f(A_1, A_2, L)$$

É fácil verificar que quando:

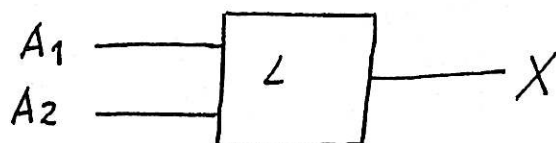
$A_1 \equiv A_2 = 1$; $L = 0$; $X = 1$ estamos perante um estado estável.

No entanto se :

$A_1 \equiv A_2 \equiv 1$; $L = 1$; $X = 0$ também é uma situação estável

Conclusão:

O sistema



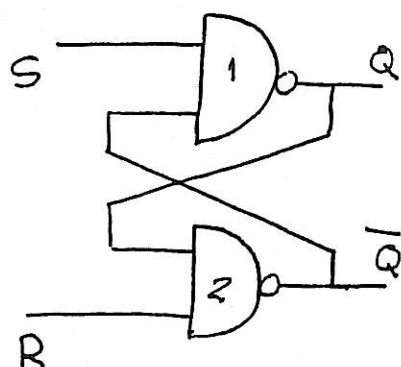
Quando tem presente à entrada os valores $A_1 \equiv A_2 = 1$, fornece uma resposta X à saída que depende do seu estado L pois $X = 0$ ou 1 consoante $L = 1$ ou 0

Os circuitos que estudaremos de seguida são deste tipo e constituem uma classe denominada de "Flip-flops" ou bistáveis

No estudo que se vai fazer será tido em consideração o atraso introduzido pelos circuitos lógicos reais utilizados.

1.1. - Bistável R-S implementado a partir de "Nand gates" e de "Nor gates"

NANDS



Preende-se um funcionamento tal que as funções representadas por Q e $\bar{Q}$ apareçam como complementares.

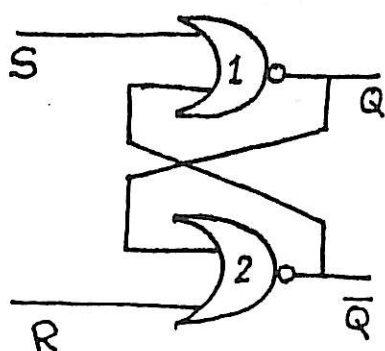
Do funcionamento das NANDS tira-se imediatamente que S e R não poderão simultaneamente estar com um nível de tensão equivalente a um zero lógico uma vez que tal implicaria em cada gate uma das entradas a "0" $\Rightarrow Q = \bar{Q} = 1$

Se $S=0$ e $R=1$, Q obrigatoriamente será "1" e como na entrada da "gate" 2 estão $R=1$ e $Q=1$ a saída $\bar{Q}$ será "0"

Simetricamente se $S=1$ e $R=0$ então $\bar{Q}=1$ e $Q=0$. Se $S=1$ e $R=1$ facilmente se verifica que o estado de saída de Q e $\bar{Q}$ corresponde ao que existia previamente antes da aplicação de $S=R=1$

Como já se referiu, neste flip-flop, a combinação de entradas $S=R=0$ que conduziria, enquanto presente, às saídas $Q=\bar{Q}=1$ é repetida como condição válida de entrada.

NORS



De modo semelhante ao caso anterior, a admissão de que simultaneamente S e R pudessem tomar o valor lógico "1" implicaria que $Q=\bar{Q}=0$ o que leva a rejeitar a

aceitação desta combinação de entrada uma vez que para o funcionamento desejado do circuito Q e $\bar{Q}$ deverão ser sempre complementares.

Se $S=1$ e $R=0$ Q obrigatoriamente terá o valor "0" uma vez que basta a existência de um "1" na entrada de uma Nor para que a saída esteja a "0". Então nas entradas da gate 2 estarão $R=0$ e $Q=0$ o que determinará $\bar{Q}=1$

Simetricamente se $S=0$ e $R=1 \Rightarrow \bar{Q}=0$ e devido a S e $\bar{Q}$ tomarem o valor "0" $\Rightarrow Q=1$.

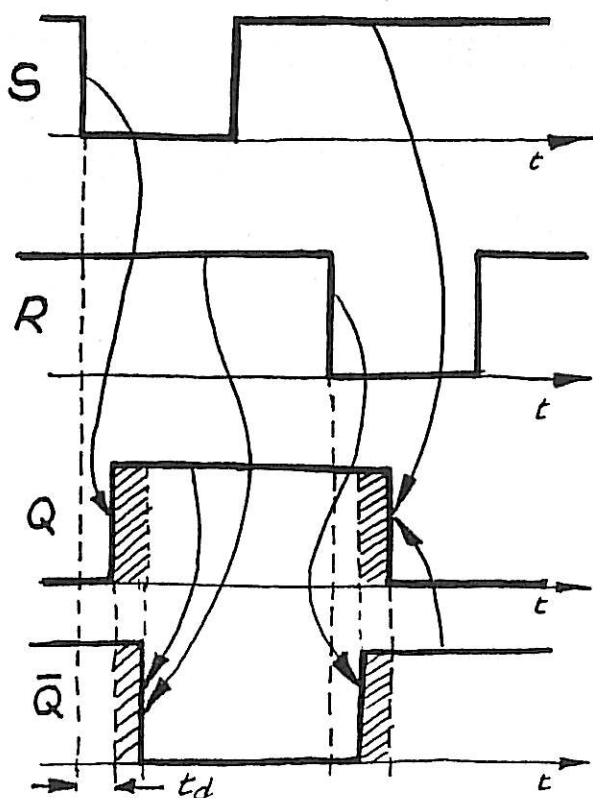
Resta analisar a situação $R=0$ e $S=0$

Se $Q=1$ e $\bar{Q}=0$ for a situação anterior nada se altera a saída

Se $Q=0$ e $\bar{Q}=1$ for a situação anterior também nada se altera a saída.

Portanto a situação $R=0=S$ mantém os valores das saídas que existiam anteriormente

Passar-se-á à análise temporal pulsada de um dos flip-flops, no caso o formado por NANDS. Admitir-se-á como tempo de atraso o valor t_d e simular-se-ão as várias combinações pulsadas de entrada.



Como se vê no diagrama temporal, devido aos tempos de atraso das gates existem zonas temporais em que Q e $\bar{Q}$ não observam valores complementares. Devido a serem intervalos da ordem dos tempos de atraso pode desprezar-se a sua existência na maioria das situações.

Bom conclusão pode-se sistematizar em duas tabelas o funcionamento lógico das biestáveis

NANDS

S	R	Q_t	$\bar{Q}_t$
0	1	1	0
1	0	0	1
1	1	Q_{t-1}	$\bar{Q}_{t-1}$
0	0	1	1

← ESTADO PROIBIDO

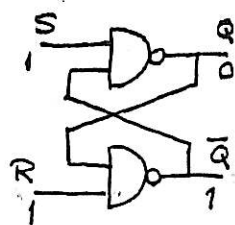
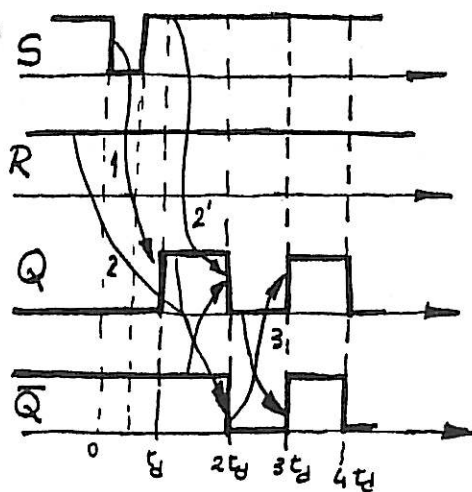
NORS

S	R	Q_t	$\bar{Q}_t$
0	1	1	0
1	0	0	1
0	0	Q_{t-1}	$\bar{Q}_{t-1}$
1	1	0	0

← ESTADO PROIBIDO

Normalmente associa-se a entrada S (set) a capacidade de quando levado ao estado diferente do de "não mudeça"; ($S=R=1$ no caso das NANDS e $S=R=0$ no caso das NORS), colocar a saída Q em "1" razão porque a seguir a esta convenção se deveria ter desenhado o esquema com NORS trocando o S com o R

Analisemos para finalizar uma possibilidade de oscilação dos flip-flops de NANDS desde que se admita tempos de subida e descida muito menores que t_d e largura do sinal S aproximadamente igual a t_d



Quando S vai para "0" durante t_d provoca que Q vá de "0" para "1" e enquanto S já retomou o valor "1"; $R=1$ e $Q=1$ provoca $\bar{Q}=0$ com um atraso de $2t_d$. Agora $S=1$ e $\bar{Q}=1$ obriga Q a retornar o valor "0" e assim sucessivamente.

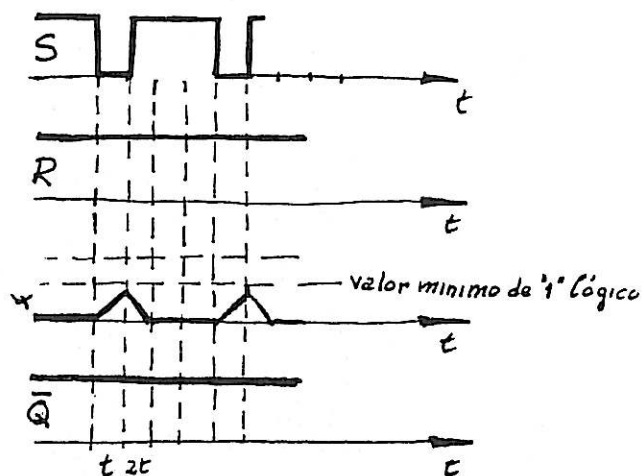
$$t > 0 : 1 - S=0 \text{ e } R=1 \Rightarrow Q \rightarrow 1 (t_d)$$

$$t > t_d : 2 - \begin{cases} R=1 \text{ e } Q=1 \Rightarrow \bar{Q} \rightarrow 0 (2t_d) \\ S=1 \text{ e } \bar{Q}=1 \Rightarrow Q \rightarrow 0 (2t_d) \end{cases}$$

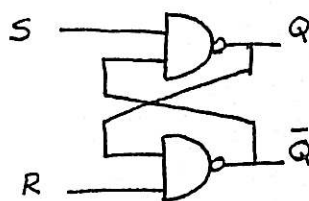
$$t > 2t_d : 3 - \begin{cases} S=R=1 \text{ e } \begin{cases} Q=0 \Rightarrow \bar{Q} \Rightarrow 1 \\ \bar{Q}=0 \Rightarrow Q \Rightarrow 1 \end{cases} (3t_d) \end{cases}$$

Portanto o flip-flop vai oscilar com uma frequência $f = \frac{1}{2t_d}$.
Do ponto de vista prático como os tempos de subida e descida não são muito mas da mesma ordem de grandeza que t_d total não pode.

Exemplaremos um exemplo com tempos de subida e descida nulos para os impulsos de R ou S e tempos finitos para as saídas Q e $\bar{Q}$.



$$t_{subida} = t_{desc.} \geq 2t$$

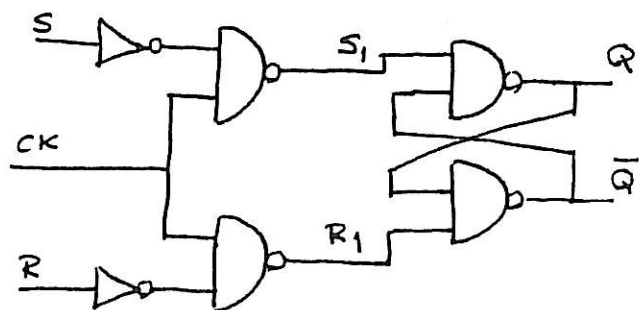


A saída $\bar{Q}$ mantém-se inalterável em '1' enquanto Q tenta atingir o valor '1' mas devido ao tempo diminuído em que S se mantém em '0' (t_d ou menor) não consegue atingir o valor mínimo de '1' que pudesse provocar um basculamento.

1.2 - BISTÁVEL R-S (SÍNCRONO)

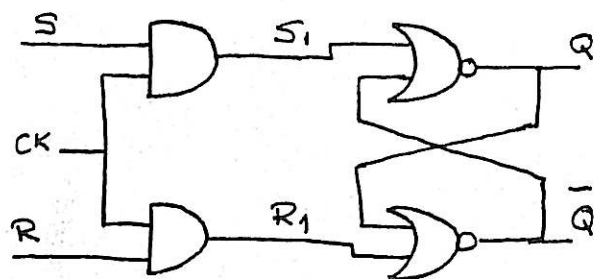
Como anteriormente se viu, quer nos flip-flops implementados com Nands quer com NORs, existem respectivamente os estados $R=S=1$ e $R=S=0$ que não determinam por si só o estado do flip-flop.

A introdução de um 'clock' permite manter as entradas nesses estados inócuos até ao momento em que o clock tomar o valor '1'.



Enquanto $CK=0$ $S_1=R_1=1$;

quando $CK=1$ $S_1=S$ e $R_1=R$



Enquanto $ck=0$ $S_1=R_1=0$

quando $ck=1$ $S_1=S$ e $R_1=R$

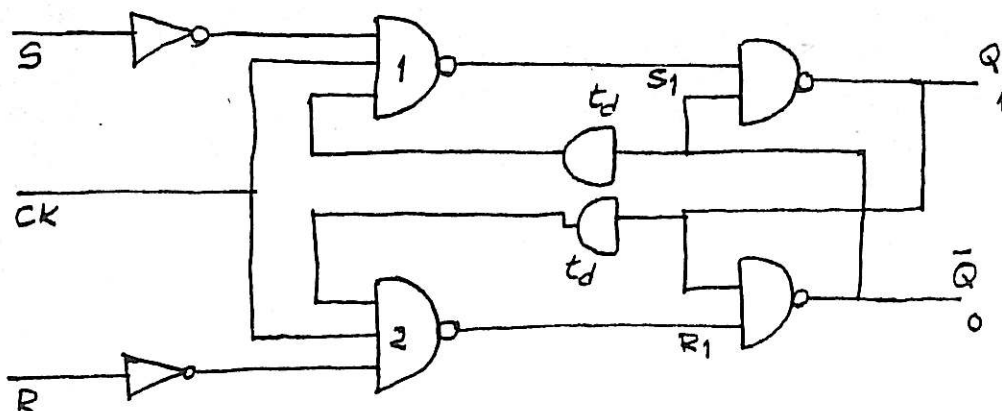
Se um flip-flop de Nands se colocar $S=R=0$ quando $ck=1 \Rightarrow S_1=R_1=0$ e $Q=\bar{Q}=1$. Quando ck retorna o valor "0" fica $S_1=R_1=1$ e o bistável avaliará para uma posição que poderá ser $Q=1$ e $\bar{Q}=0$ ou $Q=0$ e $\bar{Q}=1$ indeterminada.

Situação semelhante se passa com os flip flops feitos com NORs quando $S=R=1$

1.2.1 - Eliminação do estado proibido de entrada do flip-flop RS síncrono.

É possível, através de um processo de realimentação e impondo algumas restrições ao sinal de clock, conseguir que os flip flops baseiem para a posição complementar sempre que suas entradas R-S sejam colocadas $R=S=0$, para a implementação com Nands e $R=S=1$ para a implementação com NORs

NANDS



Coloque-se $R=S=0$ e admita-se a saída em $Q=1$; $\bar{Q}=0$. A gatez encontra-se com possibilidades de ser aberta desde que $ck=1$ uma vez que $R_1 = \bar{R}Qck$.

A gate 1 devido a $\bar{Q}=0$ não será aberta quando $ck=1$ uma vez que $S_1 = \bar{S}\bar{Q}ck$.

Portanto quando o clock ck evolui para "1" a combinação: $S=1$ e $R=0$ que se obtém provoca a mudança de estado de saída para $Q=0$ e $\bar{Q}=1$.

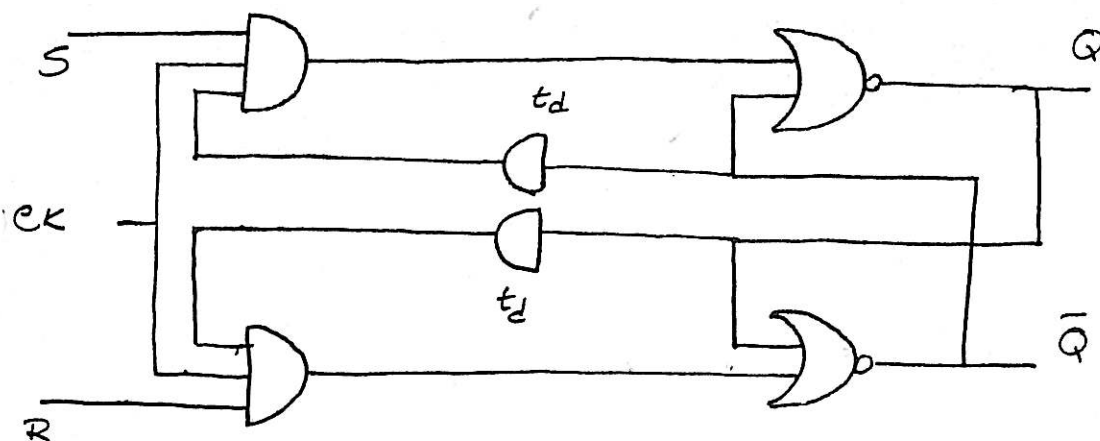
Há que impor uma limitação à largura de impulso positivo "1" de ck que não deve permitir que a mudança das saídas Q e $\bar{Q}$ venham actuar novamente nas gates 1 e 2 provocando sucessivas basculações.

Assim ck deverá manter-se em "1" durante pelo menos $2t_d$ mas deverá ser inferior a $6t_d$.

Esta limitação torna o circuito não utilizável e prepara-mos para o estudo de bistáveis comandados por flanco de impulso (edge triggered).

É possível desenhar também um flip flop de NORS com as características do flip flop de NANDS descrito atrás.

NORS.



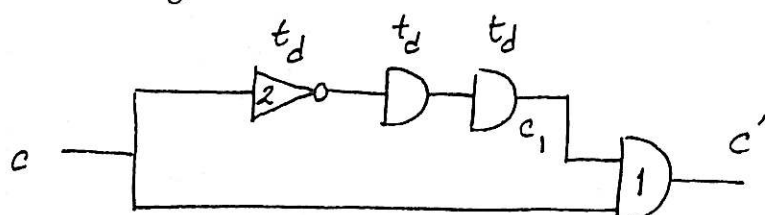
Do mesmo modo a combinação de entradas $R=S=1$ provocará o basculamento das saídas (quaisquer que sejam) e identicamente ter-se-á que impor limitações ao clock ck de modo que este seja maior que $2t_d$ e menor que $6t_d$.

1.3 - EDGE TRIGGERED FLIP FLOP

Preende-se que o funcionamento dos bistáveis anteriormente estudados se tornem independentes da largura (excessiva) dos impulsos do clock.

Então, se em vez de c_k se utilizar um circuito que converta qualquer impulso de largura maior que $6t_d$ num impulso adequado ($2t_d < t < 6t_d$) ter-se-á um bistável sem as limitações anteriormente requeridas.

Análise-se o circuito:



Quando $c = "0"$ $c_1 = "1"$ e $c' = "0"$

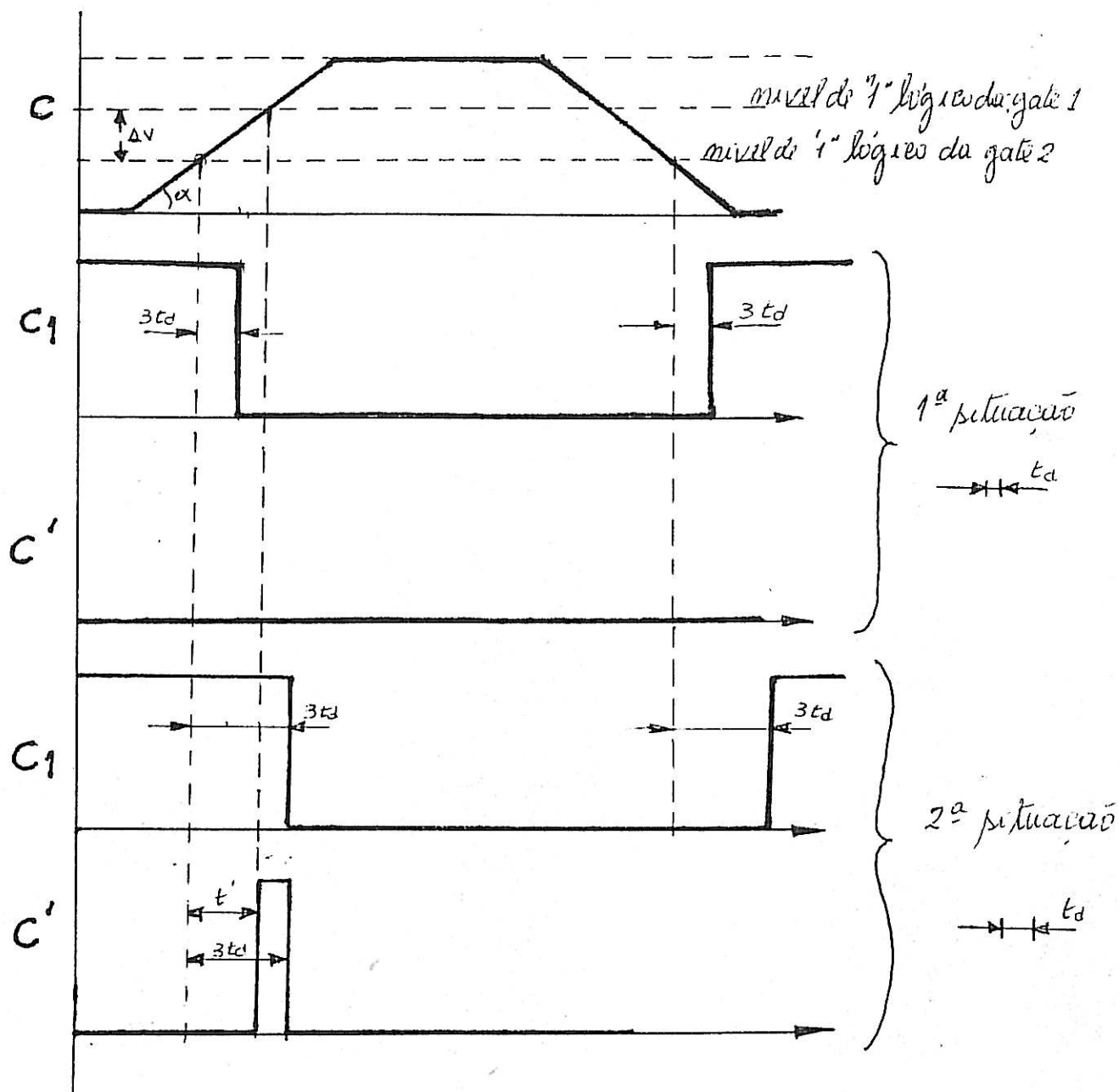
Considere-se agora que o tempo de subida do impulso c' muito menor que t_d .

Então quando $c \rightarrow "1"$ $c_1 \rightarrow "0"$ mas só atinge o valor "0" passado um tempo $3t_d$. Portanto durante este tempo $3t_d$ coexistem à entrada da gate 1 $c_1 = c = "1"$ e portanto c' mantém-se como um "1" lógico durante um tempo $3t_d$, qualquer que seja a duração do impulso em c .

Substituindo nos flip flops anteriores a entrada do clock por c' teremos um flip flop que dispara portanto no flanco de subida do clock e que passado o tempo $3t_d$ bloqueia as entradas de S e R.

Aparece contudo uma limitação relativa ao tempo de subida do clock que poderá ser descrita facilmente pelo diagrama seguinte.

Adverte-se que o nível de tensão para o qual a gate 2 reconhece um "1" lógico é inferior ao da gate 1

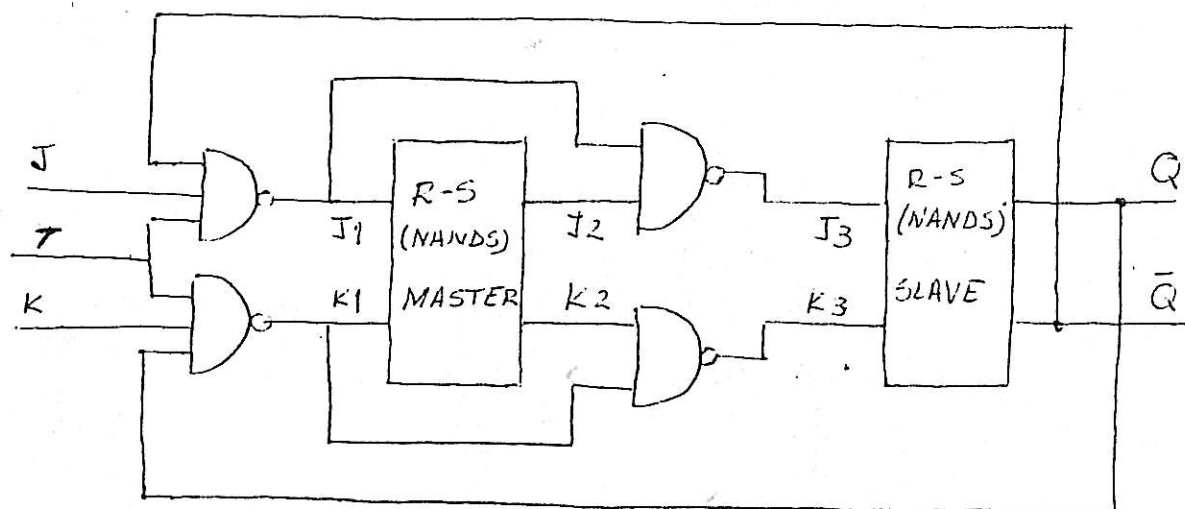


Na segunda situação o impulso vem deslocado do tempo de atraso da gate 1 mas com uma duração equivalente tempo durante o qual coincideu $c = c_1 = 1'$ que é $3t_d - t'$. Daqui, e dado que o impulso c' tem que ter pelo menos $2t_d$ pode definir-se a inclinação mínima do flanco de c , conforme as tolerâncias de valores dadas para o "1" das gates 1 e 2

1.4 - BIESTAVEL J-K MASTER-SLAVE (domo - escravo)

No exemplo anterior apareceu a necessidade de limitar a largura do impulso de clock de modo a que a realimentação das saídas para a entrada não venha a provocar um processo oscilatório.

Para fugir a este tipo de limitação introduz-se uma configuração a dois tempos (normalmente associadas aos tempos de subida e descida do clock), mediante a utilização de dois biestáveis colocados em série:



Assim, durante o primeiro tempo a informação J-K determina o posicionamento do biestável R-S Master e durante o segundo esta informação é transferida para o biestável Slave e aparece à saída.

Admita-se um determinado estado de saída assim como o posicionamento interno que lhe pode ter dado origem.

$$Q = "1" \quad J_3 = 0 \quad J_2 = "1"$$

$$\bar{Q} = "0" \quad K_3 = 1 \quad K_2 = "0"$$

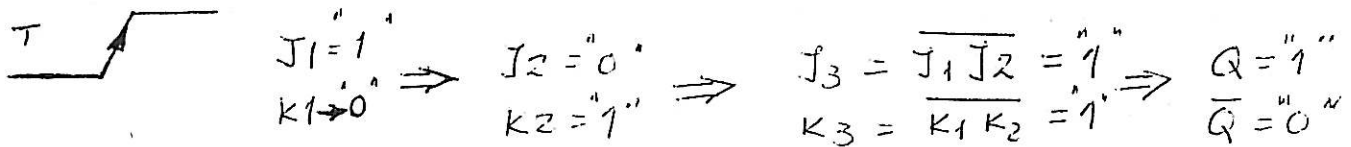
$$T = "0"$$

Admita-se $J = "0"$ e $K = "1"$

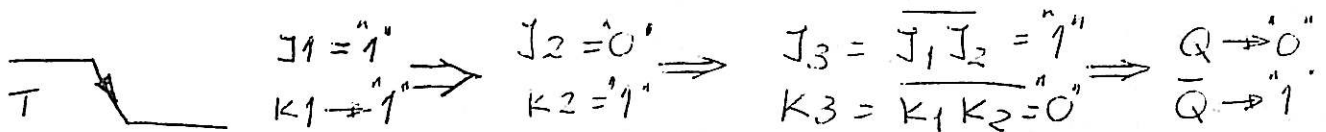
Quando $T=0$ $J_1 \equiv K_1 = "1"$; $J_3 = "0"$, $K_3 = "1"$.

Com o $\bar{Q} = 0$ à entrada da gate 1, J será sempre "1" independente de T . Como $Q = 1$ à entrada da gate 2 K_1 tomará o valor "0" quando $T \rightarrow "1"$ uma vez que $K = "1"$.

Com a transição de T de "0" para "1" vamos ver o que acontece.

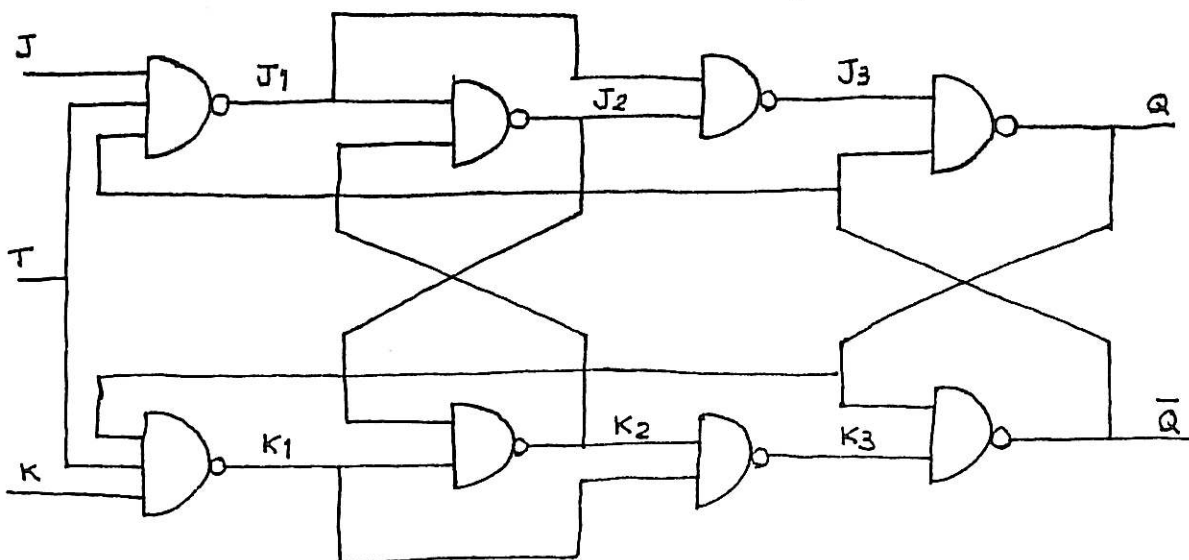


Quando se dá a transição de T de "1" para "0" temos:



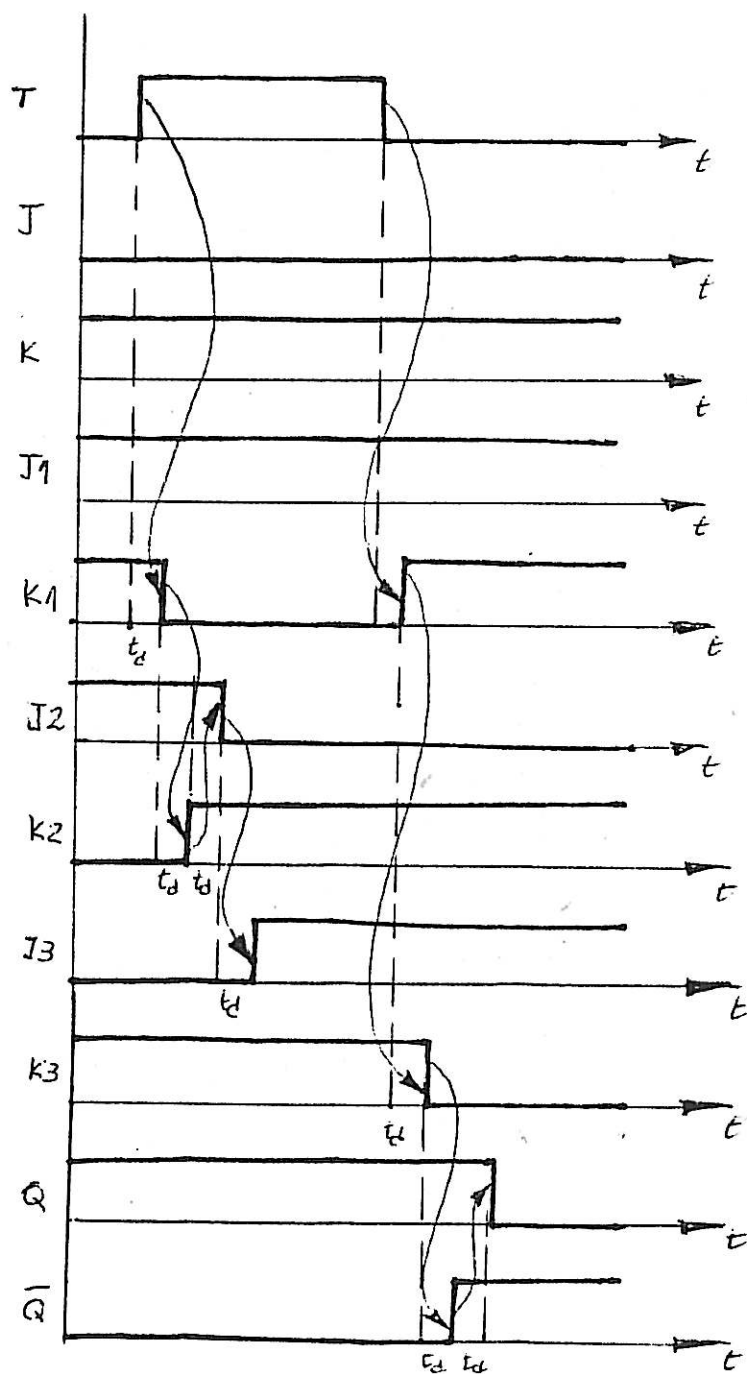
Concluindo, durante a evolução do clock de "0" para "1" a informação contida em J e K vai determinar o posicionamento do flip flop Master ou seja os valores de J_2 e K_2 . No entanto devido à existência das gates 3 e 4 só quando $J_1 = K_1 = "1"$ ocorrer, o que acontece quando o clock evolui de "1" para "0", é que J_3 e K_3 podem ser posicionadas determinando o estado de saída do flip flop slave.

Uma implementação real integrada poderia ser a seguinte:



Os flancos de subida poderão agora ser feitos assim como é possível a existência de assimetrias de nível de disparo. Por outro lado o bloqueamento conseguido entre a entrada e saída através da existência de dois bistáveis impede processos oscilatórios.

Para o funcionamento atrás descrito é possível estabelecer o seguinte diagrama temporal



Limitação:
 largura do impulso T
 $> 2t_d$
 conforme se pode
 verificar

Da análise do biestável descrito pode-se inferir que:

J	K	Q_{t+1}
0	0	Q_t
1	0	1
0	1	0
1	1	$\bar{Q}_t$

esta tabela pode
ser descrita tam-
bem dos seguintes
modos:

Q_m	J	K	Q_{m+1}
0	0	0	0
0	1	0	1
0	0	1	0
0	1	1	1
1	0	0	1
1	1	0	1
1	0	1	0
1	1	1	0

Q_m	J K				Q_{m+1}
	00	01	11	01	
0	0	1	1	0	
1	1	1	0	0	

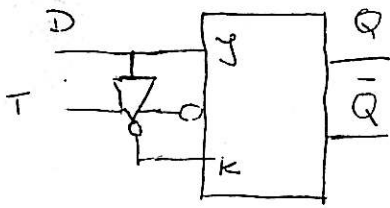
Q_m	Q_{m+1}	J	K
0	0	0	ϕ
0	1	1	ϕ
1	0	ϕ	1
1	1	ϕ	0

Os símbolos ϕ significam que é irrelevante essas variáveis tomarem o valor "1" ou "0" para se provocarem as transições re-
queridas, desde que as outras condições se verifiquem.

1.5- Biestável D. tipo edge triggered, tipo latch e tipo Master Slave.

Preende-se um biestável em que a informação seja forne-
cida numa entrada denominada D e a sua transferência
para a saída seja feita sincronamente com o clock.

1.5.1 Flip-flop D, master slave.



De imediato se vê que a introdução do inversor entre J e K torna obrigatório que $J = \bar{K}$ ou seja, as únicas combinações possíveis são:

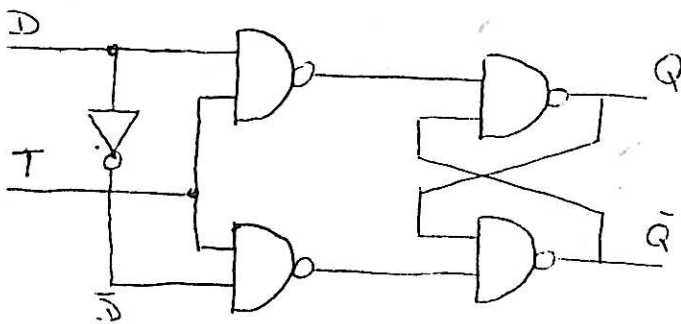
$$D = 0 \rightarrow J = 0 \quad K = 1 \rightarrow Q = 0$$

$$D = 1 \rightarrow J = 1 \quad K = 0 \rightarrow Q = 1$$

ou seja a saída Q tomara o valor da entrada D sincronamente com o flanco descendente do clock (flip-flop master-slave).

D	Q_{n+1}
0	0
1	1

1.5.2 - Flip-flop D latch



Se $D = 0$ $\bar{D} = 1$ e quando $T \rightarrow 1$ $\bar{Q}_1 \rightarrow 0$

$$Q \rightarrow 0$$

Se $D = 1$ $\bar{D} = 0$ e quando $T \rightarrow 1$ $Q \rightarrow 1$

$$\bar{Q} \rightarrow 0$$

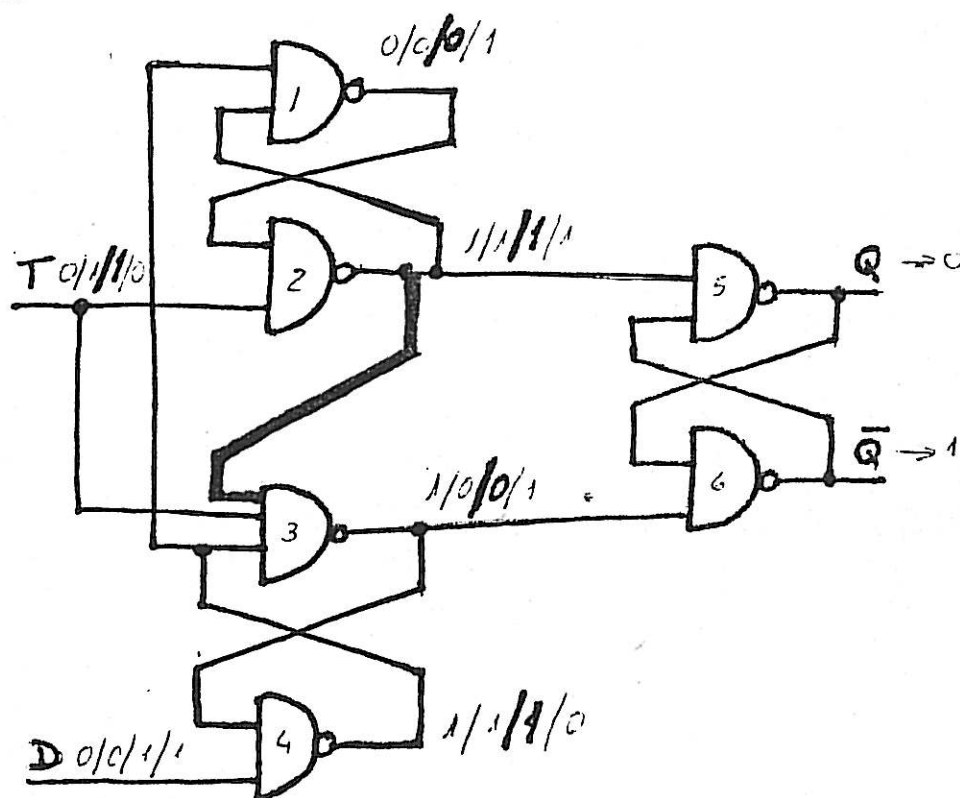
Se durante o tempo em que T está em 1 a entrada D mudar o seu valor a saída seguirá-a

1.5.3 - Biestável D edge triggered.

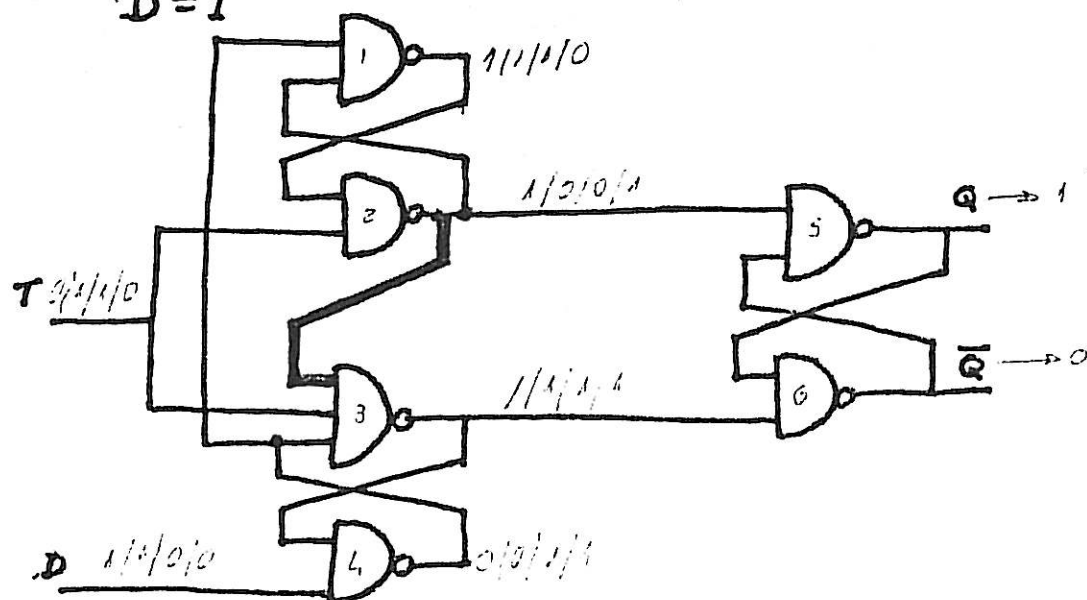
A substituição da entrada T por um circuito equivalente ao anteriormente estudado no caso do flip-flop JK edge triggered, permitiria a conversão imediata do biestável D latch.

Contudo a configuração existente na prática sendo relativamente diferente, torna aconselhável o seu estudo.

$D=0$



$D=1$

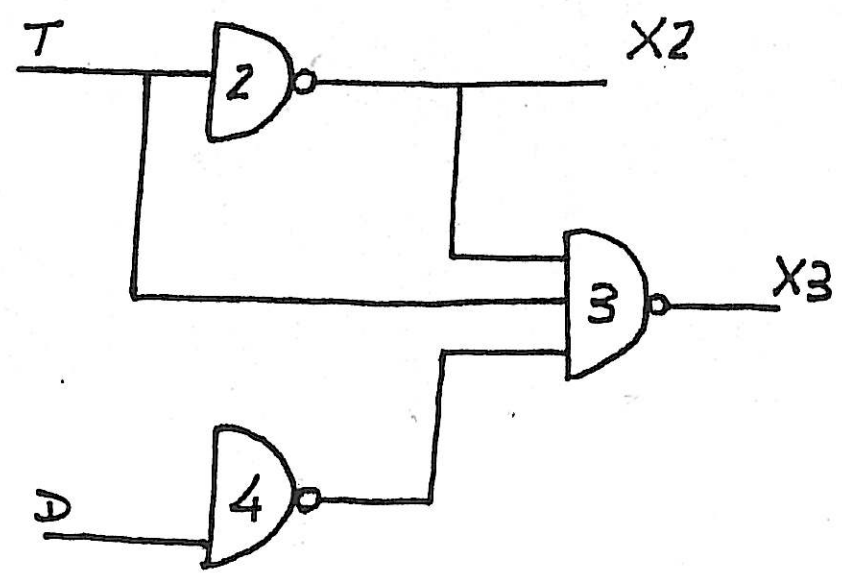


Conforme se vê, a entrada D deixa de influenciar o bistável de saída uma vez realizada a transição de "0" para "1" da saída do clock.

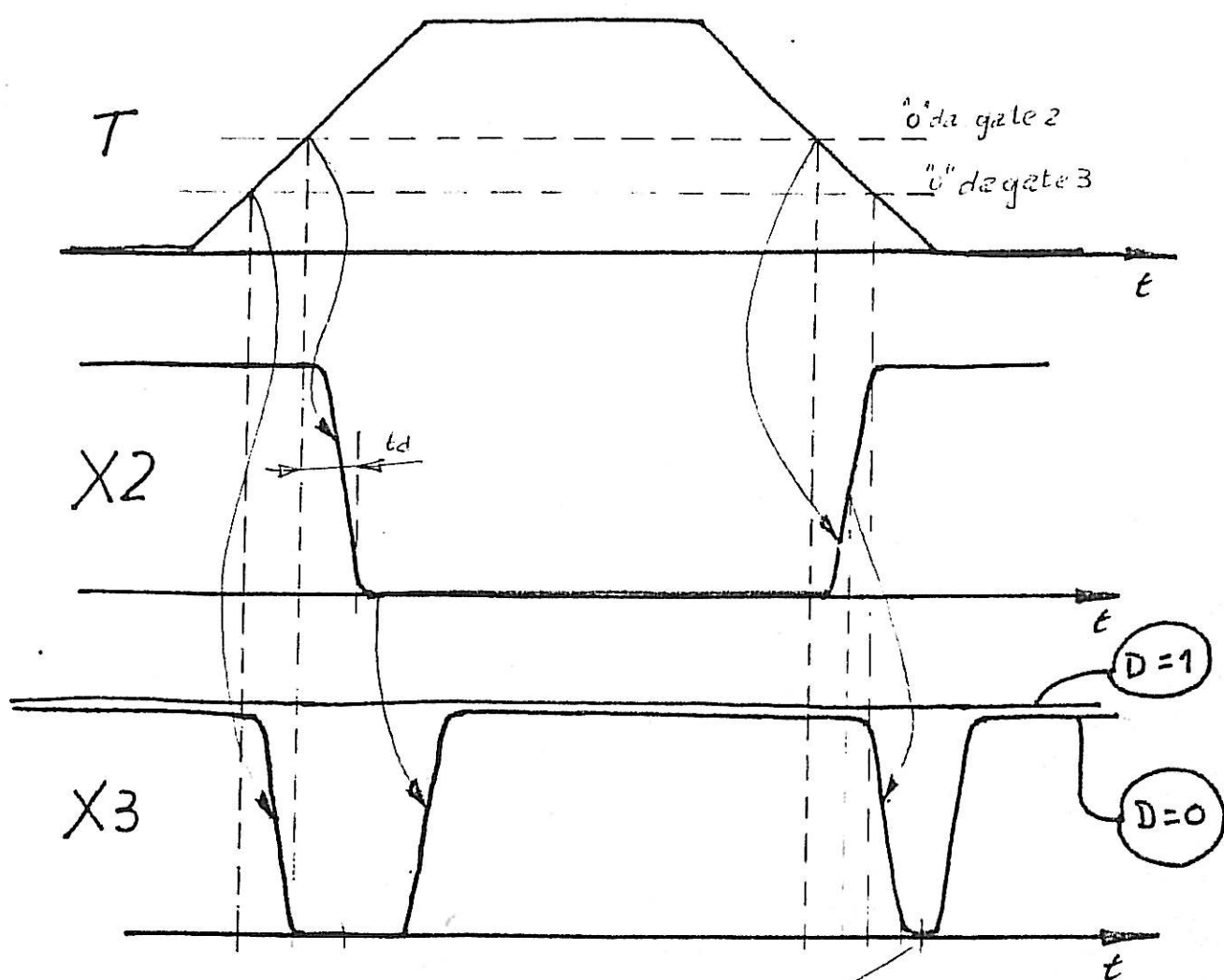
Analisemos separadamente cada um dos casos: $D = 0$ e $D = 1$ (ver figura)

Após se dar a transição $0 \rightarrow 1$ de T a entrada D não interfere; embora a saída da gate 4 evolua para "1" quando $D \rightarrow 0$, não há modificação a saída da gate 3 devido à ligação directa a clock (traco mais forte) fechar a gate 3 mantendo a sua saída o "1"

Quando T retorna o valor "0" há que ter cuidado com os níveis de disparo das gates 2 e 3 uma vez que se 2 reconhecer mais cedo que 3 o nível equivalente a um "0" entrão à sua saída aparece com um "1" mais cedo do que 3 é fechada pelo clock e devido à ausência de três "1" a entrada pode haver alteração da saída do bistável



Analisemos em detalhe o diagrama temporal deste fenómeno de glitch.



— pino que não deveria
 acionar e que provoca o
 aumento do fup - p/p

Portanto através das gates 1, 2, 3 e 4 consegue-se que D se
 active durante a transição de ϕ para ϕ'

2. Desenho expedito de circuitos sequenciais

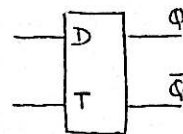
A partir dos circuitos de memória anteriormente descritos (FF D e JK) é possível implementar sistemas pulsados, susceptíveis de seguir determinados códigos anteriormente elaborados.

A cada bit do código se faz a correspondente equivalência a uma função de memória (FF).

Trataremos sistemas síncronos e assíncronos e faremos a sua análise e síntese.

a) Sistemas de memória

1- Flip Flop tipo D

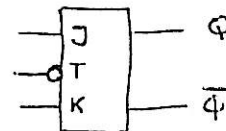


A transição "0" → "1" do clock (T) coloca a informação existente em D na saída Q. Assim o seu mapa de funcionamento, em que se faz intervir a variável tempo tem:

D_t	Q_{t+1}
1	1
0	0

Interpretação: Se colocarmos a informação em D no instante t, no instante t+1 (novo impulso de entrada) ela transitará para a saída.

2- Flip Flop tipo JK



De modo semelhante se pode estabelecer a tabela

J_t	K_t	Q_{t+1}
0	0	Q_t
0	1	0
1	0	1
1	1	$\bar{Q}_t$

→ Mostra a mudança

Do quadro atrás pode-se inferir que as quatro possibilidades de evolução da saída, conforme as situações de J e K, são:

Saída	J K	ϕ representa estados indiferentemente "0" ou "1".
$0 \rightarrow 0 \equiv \textcircled{0}$	0ϕ	
$0 \rightarrow 1 \equiv 1$	1ϕ	
$1 \rightarrow 1 \equiv \textcircled{1}$	$\phi 0$	
$1 \rightarrow 0 \equiv 0$	$\phi 1$	

2.1. Síntese de circuitos sequenciais síncronos

Escolha-se um conjunto de palavras que se pretende venham a se realidades de um modo sequencial.

Estado	A	B	C
0	0	0	0
1	1	0	0
2	0	1	1
3	1	1	1
4	0	0	1

Se a cada bit se fizer corresponder uma função de memória (FF) ter-se-á necessidade de três variáveis que serão representadas por A, B e C.

O problema consiste em modelar estas funções de memória de uma rede lógica combinacional que determine que as variáveis A, B e C sigam os estados indicados sempre que o sistema seja estimulado por impulsos que o actuem simultaneamente.

Estabelecemos um quadro intermédio que nos dê conta dos estados seguintes das variáveis (X_A , X_B , X_C) e do tipo de modificação que se determinou.

A	B	C	X_A	X_B	X_C
0	0	0	1	$\textcircled{0}$	$\textcircled{0}$
1	0	0	0	1	1
0	1	1	1	$\textcircled{1}$	$\textcircled{1}$
1	1	1	0	0	$\textcircled{1}$
0	0	1	$\textcircled{0}$	$\textcircled{0}$	0

O problema encontra-se agora simplificado uma vez que se conhecem as condições a que devem obedecer as variáveis de estado das funções memória D e JK, a fim de provocarem as transições pretendidas.

	FF "D"
1	$0 \rightarrow 1$
①	$1 \rightarrow 1$
①	$0 \rightarrow 0$
0	$1 \rightarrow 0$

$\Rightarrow D=1$
 $\Rightarrow D=0$

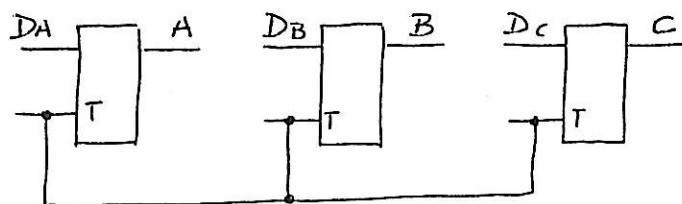
	FF "JK"	J	K
1	$0 \rightarrow 1$	$\Rightarrow$	$1 \ \phi$
①	$1 \rightarrow 1$	$\Rightarrow$	$\phi \ 0$
①	$0 \rightarrow 0$	$\Rightarrow$	$0 \ \phi$
0	$1 \rightarrow 0$	$\Rightarrow$	$\phi \ 1$

Com estes dados podemos completar o quadro de fim da pag. anterior ou seja:

2.1.1- utilizando FF tipo D

	A	B	C	X _A	X _B	X _C	D _A	D _B	D _C
0	0	0	0	1	①	①	1	0	0
1	1	0	0	0	1	1	0	1	1
2	0	1	1	1	①	①	1	1	1
3	1	1	1	0	0	①	0	0	1
4	0	0	1	①	①	0	0	0	0

NOTA: o significado, por exemplo, de $D_A = 1$ quando $A=B=C=0$ é que, com o próximo sinal de clock Φ_A tomará o valor "1".



Passemos agora à obtenção da função representativa das variáveis D_A, D_B, D_C para os estados possíveis de A, B e C.

Recomendo aos diagramas de Káry a fim de simplificar essas funções vem:

	B	$\bar{B}$	
A	X	0	$\bar{C}$
	0	X	C
$\bar{A}$	1	0	C
	X	1	$\bar{C}$

$$D_A = \bar{A}B + \bar{A}\bar{C}$$

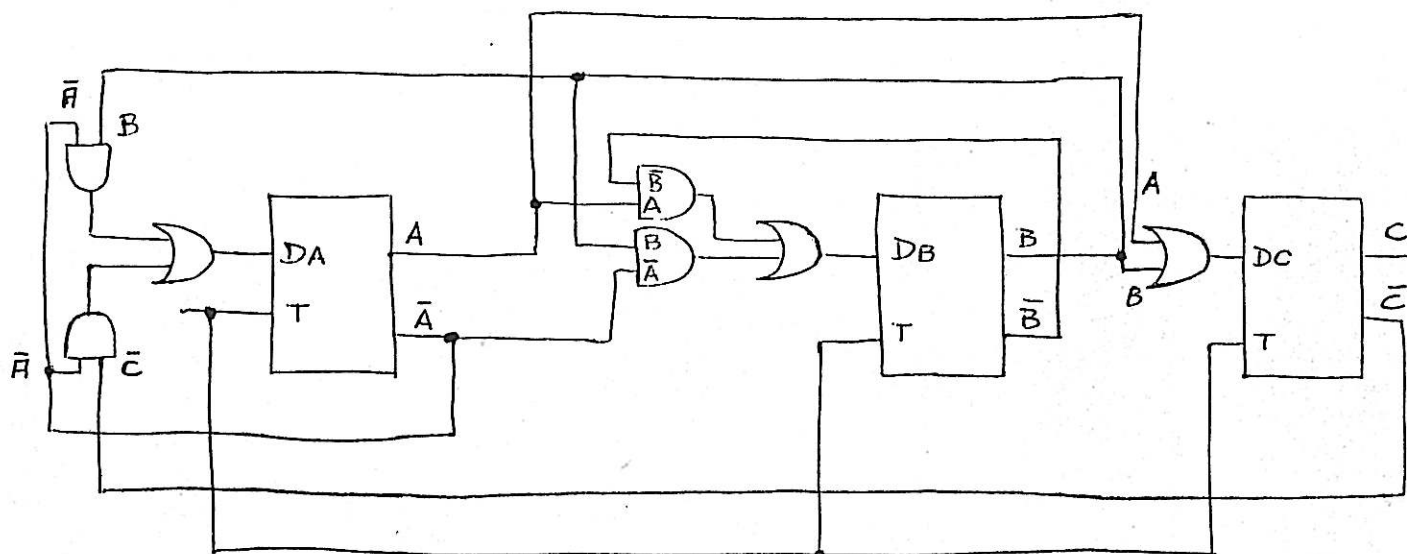
	B	$\bar{B}$	
A	X	1	$\bar{C}$
	0	X	C
$\bar{A}$	1	0	C
	X	0	$\bar{C}$

$$D_B = \bar{A}B + A\bar{B}$$

	B	$\bar{B}$	
A	X	1	$\bar{C}$
	1	X	C
$\bar{A}$	1	0	C
	X	0	$\bar{C}$

$$D_C = A + B$$

Então a implementação do contador será:



A análise deste mesmo sistema poderá agora ser feito colocando o sistema num dos estados pertencentes ao código e verificando depois se ele segue a sequência correcta ao se actuar a entrada T. Exemplifiquemos:

$A_t B_t C_t$	$D A_t$	$D B_t$	$D C_t$	$A_{t+1} B_{t+1} C_{t+1}$
3 $\equiv$ 1 1 1	0	0	1	0 0 1
4 $\equiv$ 0 0 1	0	0	0	0 0 0
0 $\equiv$ 0 0 0	1	0	0	1 0 0
1 $\equiv$ 1 0 0	0	1	1	0 1 1
2 $\equiv$ 0 1 1	1	1	1	1 1 1

Como vimos a obtenção das funções DA, DB, DC foi feita na presunção de que os estados "110", "010" e "101" nunca sucederiam.

Por acidente, pode, contudo, o contador cair num destes estados. Interessa desenvolver a análise destas situações, e o modo de sair delas.

Admitamos que por efeito fortuito $A=B=1, C=0$

A_t	B_t	C_t	DA_t	DB_t	DC_t	A_{t+1}	B_{t+1}	C_{t+1}
1	1	0	0	0	1	0	0	1
0	0	1	0	0	0			

Como se vê o sistema retorna o código após um impulso do clock.

Vejamos as outras duas cases:

→ 0 1 0 0 1 1 0 1 1
 0 1 1 → Retornou o código

→ 1 0 1 0 1 1 0 1 1
 0 1 1 → Retornou o código

Note-se que este retorno do código não é obrigatório, e existem situações em que nunca mais se dá o retorno ao código principal.

Uma das maneiras de obviar a isto seria não utilizar os "don't care states" para simplificar as funções "DD", e simultaneamente garantir que nesses estados $DA=DB=DC=0$, o que obrigaria o contador a voltar ao estado inicial (000).

Vejamos como faríamos utilizando ainda o exemplo anterior.

	A	B	C	X _A	X _B	X _C	D _A	D _B	D _C
0 ≡	0	0	0	1	0	0	1	0	0
1 ≡	1	0	0	0	1	1	0	1	1
2 ≡	0	1	1	1	1	1	1	1	1
3 ≡	1	1	1	0	0	1	0	0	1
4 ≡	0	0	1	0	0	0	0	0	0
	1	1	0	0	0	0	0	0	0
	0	1	0	0	0	0	0	0	0
	1	0	1	0	0	0	0	0	0

	B	$\bar{B}$	$\bar{C}$
A	0	0	$\bar{C}$
	0	0	C
A	1	0	$\bar{C}$
	0	1	$\bar{C}$

	B	$\bar{B}$	$\bar{C}$
A	0	1	$\bar{C}$
	0	0	C
$\bar{A}$	1	0	$\bar{C}$
	0	0	$\bar{C}$

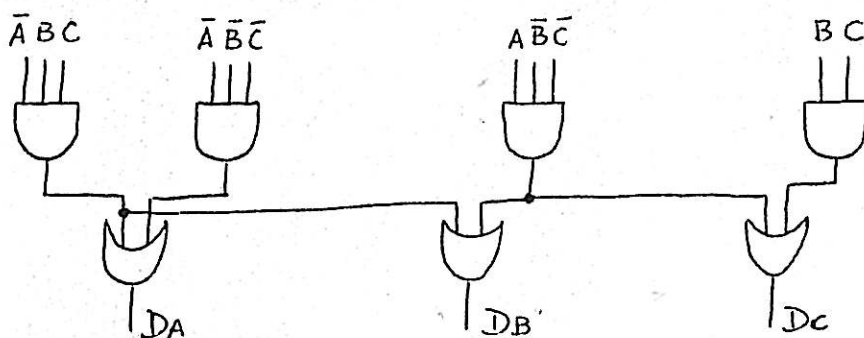
	B	$\bar{B}$	$\bar{C}$
A	0	1	$\bar{C}$
	1	0	C
$\bar{A}$	1	0	$\bar{C}$
	0	0	$\bar{C}$

$$D_A = \bar{A}BC + \bar{A}\bar{B}\bar{C}$$

$$D_B = \bar{A}BC + A\bar{B}\bar{C}$$

$$D_C = A\bar{B}\bar{C} + BC$$

Donde vêm as implementações:



Neste sistema, qualquer saída diferente do código principal, obriga-o a regressar a $A=B=C=0$ no próximo impulso.

Poder-se-ia tê-lo obrigado a regressar a qualquer outro estado, o que iria apenas modificar a parte inferior da tabela.

Seguidamente poderíamos fazer a verificação do contador de um modo semelhante ao que foi feito atrás.

2.1.2- Sistemas usando FF tipo JK

De um modo semelhante ao anterior há que definir um código e um mapa de transições para os bistáveis.

Como a tabela de funcionamento dos FF "JK" é mais elaborada do que a dos FF "D", aparece uma ligeira complexidade adicional, traduzida pela conveniência de preencher um mapa intermédio.

Tomemos a tabela do FF "JK"

J	K	Q_{n+1}		J	K
0	0	Q_n	< 0	0	ϕ
1	0	1		1	ϕ
0	1	0		ϕ	1
1	1	$\overline{Q_n}$	< 1	ϕ	0

$0 \rightarrow 0 \equiv \textcircled{0} \rightarrow 0 \phi$
 $1 \rightarrow 0 \equiv 0 \rightarrow \phi 1$
 $1 \rightarrow 1 \equiv \textcircled{1} \rightarrow \phi 0$
 $0 \rightarrow 1 \equiv 1 \rightarrow 1 \phi$

A tabela de lado direito permite individualizar os quatro casos possíveis.

A representação ϕ significa ser indiferente a função (J ou K) tomar o valor "0" ou "1". Por exemplo, para provocar uma transição "0" $\rightarrow$ "1", J deverá ser sempre "1", mas K pode ser indiferentemente "0" ou "1".

Retomemos o código anteriormente escolhido:

A	B	C	X_A	X_B	X_C	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	1	$\textcircled{0}$	$\textcircled{0}$	1	ϕ	0	ϕ	0	ϕ
1	0	0	0	$\textcircled{0}$	1	ϕ	1	0	ϕ	1	ϕ
0	0	1	1	1	$\textcircled{1}$	1	ϕ	1	ϕ	ϕ	0
1	1	1	0	0	0	ϕ	1	ϕ	1	ϕ	1

MAPA INTERMÉDIO

O modo como a partir desta tabela se simplificam as fun-

com, através de quadros de Karnaugh, e é em tudo similar ao que anteriormente foi referido aquando da implementação do sistema utilizando bistáveis "D", inclusivamente a utilização dos "casos que não sucedem".

Simplifiquemos:

	B	$\bar{B}$	
A	X	$\emptyset$	$\bar{C}$
	$\emptyset$	X	C
$\bar{A}$	X	1	
	X	1	$\bar{C}$

$$J_A = 1$$

	B	$\bar{B}$	
A	X	1	$\bar{C}$
	1	X	C
$\bar{A}$	X	$\emptyset$	
	X	$\emptyset$	$\bar{C}$

$$K_A = 1$$

	B	$\bar{B}$	
A	X	0	$\bar{C}$
	$\emptyset$	X	C
$\bar{A}$	X	1	
	X	0	$\bar{C}$

$$J_B = C$$

	B	$\bar{B}$	
A	X	$\emptyset$	$\bar{C}$
	1	X	C
$\bar{A}$	X	$\emptyset$	
	X	$\emptyset$	$\bar{C}$

$$K_B = 1$$

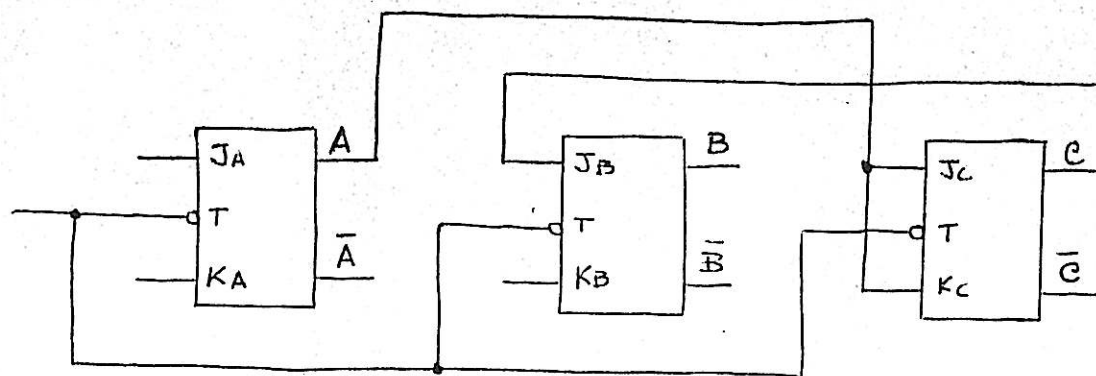
	B	$\bar{B}$	
A	X	1	$\bar{C}$
	$\emptyset$	X	C
$\bar{A}$	X	$\emptyset$	
	X	0	$\bar{C}$

$$J_C = A$$

	B	$\bar{B}$	
A	X	$\emptyset$	$\bar{C}$
	1	X	C
$\bar{A}$	X	0	
	X	$\emptyset$	$\bar{C}$

$$K_C = A$$

Donde vem a implementação:



A verificação do contador é imediata.

Parta-se por exemplo do estado $A=B=C=0$

$$\left. \begin{array}{l} J_A = 1 \\ K_A = 1 \end{array} \right\} \Rightarrow A \rightarrow 1$$

$$\left. \begin{array}{l} J_B = 0 \\ K_B = 1 \end{array} \right\} \Rightarrow B \rightarrow 0$$

$$\left. \begin{array}{l} J_C = 0 \\ K_C = 0 \end{array} \right\} \Rightarrow C \rightarrow 0$$

Logo 2º estado:

$$A = 1$$

$$B = 0$$

$$C = 0$$

2º Estado:

$$\left. \begin{array}{l} J_A = 1 \\ K_A = 1 \end{array} \right\} \Rightarrow A \rightarrow 0$$

$$\left. \begin{array}{l} J_B = 0 \\ K_B = 1 \end{array} \right\} \Rightarrow B \rightarrow 0$$

$$\left. \begin{array}{l} J_C = 1 \\ K_C = 1 \end{array} \right\} \Rightarrow C \rightarrow 1$$

Logo 3º Estado:

$$A = 0$$

$$B = 0$$

$$C = 1$$

3º Estado:

$$\left. \begin{array}{l} J_A = 1 \\ K_A = 1 \end{array} \right\} \Rightarrow A \rightarrow 1$$

$$\left. \begin{array}{l} J_B = 1 \\ K_B = 1 \end{array} \right\} \Rightarrow B \rightarrow 1$$

$$\left. \begin{array}{l} J_C = 0 \\ K_C = 0 \end{array} \right\} \Rightarrow C \rightarrow 1$$

Logo 4º Estado:

$$A = B = C = 1$$

4º Estado

$$\left. \begin{array}{l} J_A = 1 \\ K_A = 1 \end{array} \right\} \Rightarrow A \rightarrow 0$$

$$\left. \begin{array}{l} J_B = 1 \\ K_B = 1 \end{array} \right\} \Rightarrow B \rightarrow 0$$

$$\left. \begin{array}{l} J_C = 1 \\ K_C = 1 \end{array} \right\} \Rightarrow C \rightarrow 0$$

Logo 5º Estado é de novo o estado inicial de que partimos

2.2. Síntese de circuitos sequenciais assíncronos

Nestes sistemas não existe um sinal de relógio (clock) que atue simultaneamente todos os biestáveis, mas sim um conjunto de sinais que se vão propagando e vão servindo de impulsos para os diferentes FF.

A síntese deste tipo de circuitos terá que ser um pouco sistemática dadas as possíveis opções, como se verá adiante.

A partir do código que se quer percorrer, construa-se uma tabela auxiliar, em que se indicará se, com o próximo impulso, algum dos biestáveis comuta. A existência de comutação será indicada por "1" e a ausência desta por "x".

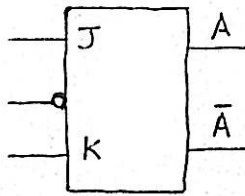
Tomemos como exemplo um contador que siga o código BCD.

	A	B	C	D	X _A	X _B	X _C	X _D
0	0	0	0	0	1	x	x	x
1	1	0	0	0	①	1	x	x
2	0	1	0	0	1	x	x	x
3	1	1	0	0	①	①	1	x
4	0	0	1	0	1	x	x	x
5	1	0	1	0	①	1	x	x
6	0	1	1	0	1	x	x	x
7	1	1	1	0	①	①	①	1
8	0	0	0	1	1	x	x	x
9	1	0	0	1	①	x	x	①
					?			

Escolhamos agora os FF que vamos utilizar. Poderão ser, p. ex., tipo JK, comutando na transição de clock de "1" para "0".

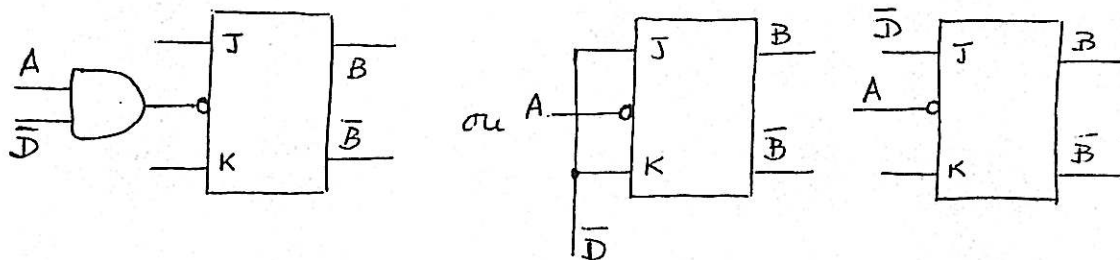
Os ① representados na tabela, representam transições "1" → "0" que poderemos ser utilizadas directamente.

O 1º FF comuta sempre que recebe um impulso, o que sugere imediatamente um FF "JK" com $J = K = 1$.

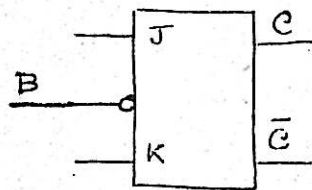


O 2º FF poderia utilizar directamente a saída A para o relógio, desde que a transição do estado 9 para 0 fosse inibida.

A função $A\bar{D}$ realta este efeito. Poder-se-ia utilizar também a inibição através do J e de K. Assim, se $\bar{D}$ for ligado a J ou a J e K, consegue-se o mesmo efeito.



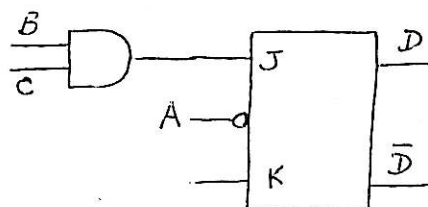
O 3º FF muda simultaneamente com as transições activas de "1" → "0" indicadas em XB. Então poderia ser B o relógio de FF C, deixando J e K em "1".



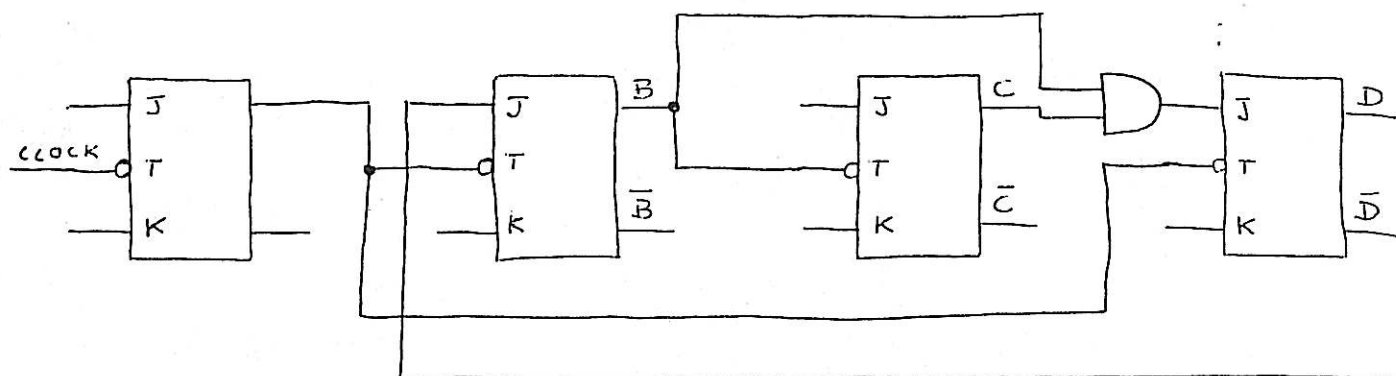
Finalmente o 4º FF deverá mudar no estado 7 e 9.

Xc indica que poderia ser utilizada a transição "1" → "0" de C para provocar a mudança do estado 7 para 8. No entanto, a observação do código mostra que B e C tomam o valor "1" simultaneamente durante o estado 7.

Assim, se ligarmos a função B.C ao terminal J do FF nº 4 (D) e usarmos A como relógio, este seguirá o código desejado.



A implementação final seria



Conclusão: A implementação de contadores assíncronos poderia ser realizada de um modo semi-sistematizado, dependendo a solução encontrada em parte da intuição do projectista, desde que o método seguido seja o indicado.

Devido ao facto do sistema ser assíncrono, as transições dos FF ocorrem com atrasos diferentes, podendo por isso gerar-se estados prionisólios, ou seja de passagem, que não correspondem ao código desejado.

Por exemplo, se, na transição de 7 para 8, o FF C demorar mais tempo a comutar que o FF D, devido ao atraso do impulso de relógio que o actua, então é possível o aparecimento do estado $A=0$, $B=0$, $C=1$ e $D=1$ durante um certo tempo, até o FF C por sua vez comutar.