

”

E-fólio A | Folha de resolução para E-fólio

UNIDADE CURRICULAR: Arquitetura de Computadores

CÓDIGO: 21010

DOCENTE: Gracinda Carvalho; José Coelho **TUTOR:** Carlos Sousa

A preencher pelo estudante

NOME: Júlio César Gomes de Barros

N.º DE ESTUDANTE: 1902295

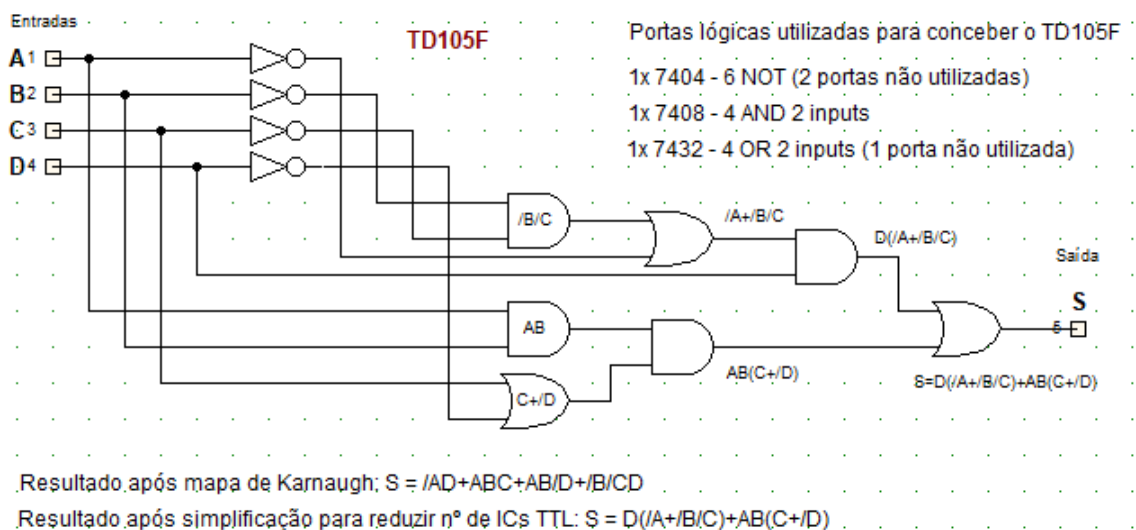
CURSO: Licenciatura em Engenharia Informática

A)

Foram procuradas 3 soluções para resolver o problema proposto, na 2ª forma canónica, na 1ª forma canónica (ver anexo A), tendo sido adotada a que resultou da simplificação da expressão encontrada nesta última.

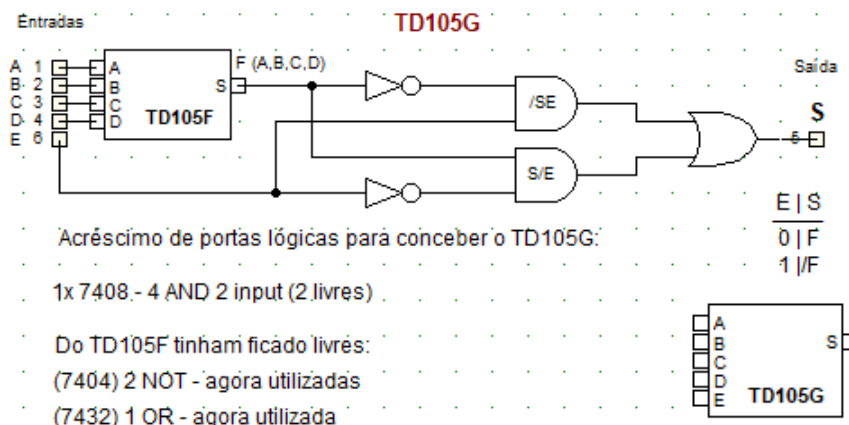
Simplificação adicional: $S = \bar{A}D + ABC + AB\bar{D} + \bar{B}\bar{C}D = D(\bar{A} + \bar{B}\bar{C}) + AB(C + \bar{D})$

Esta foi a opção adotada para o TD105F por economia e vulgaridade de ICs.



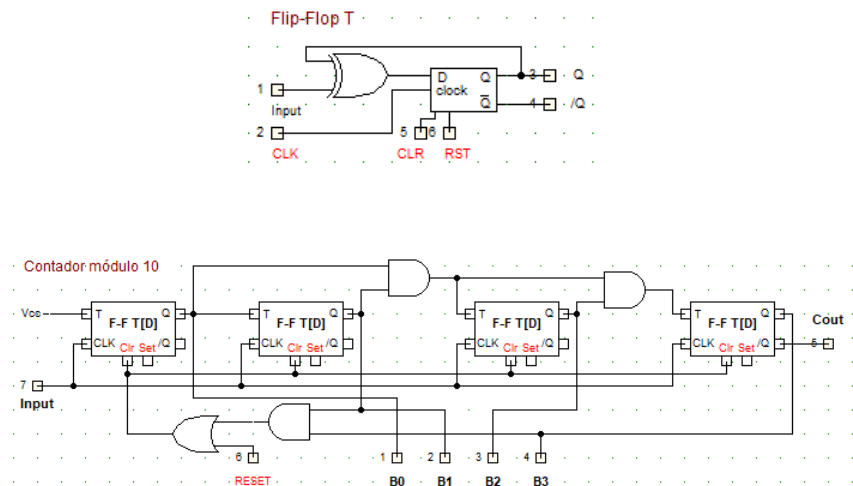
B)

Como a nova variável E se comporta como uma variável seletora, que quando a 0 fará com que à saída do novo componente TD105G se apresente a função $F(A,B,C,D)$ que consta na saída de TD105F, e quando $E=1$, termos a negação dessa função, isto é $\bar{F}$, um circuito simples permite resolver, com alguns recursos sobranes da implementação de TD105F, e apenas mais 2 portas AND.



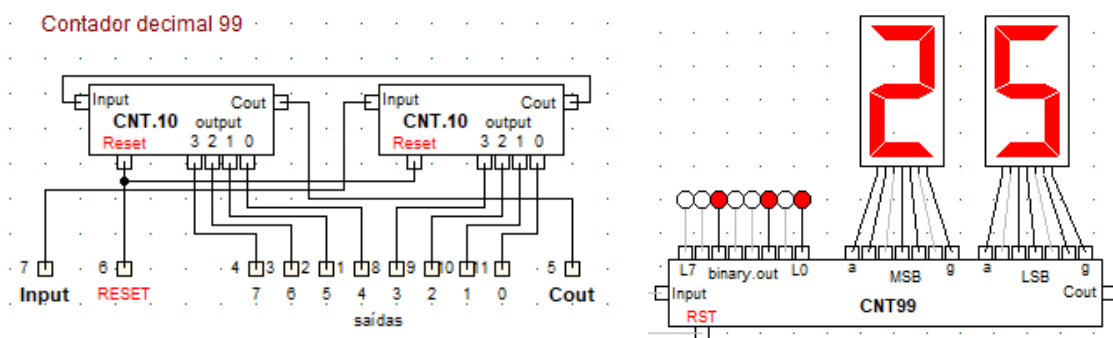
C)

dec	B3	B2	B1	B0
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

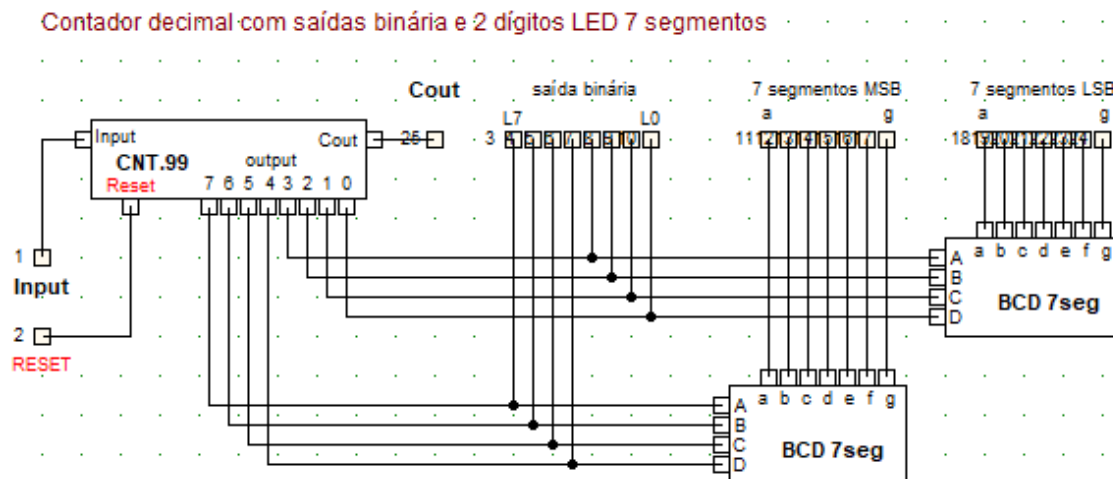


Para efetuar uma contagem até 10, de 0 a 9, (sem usar por exemplo um contador binário duplo 74393) é necessário recorrer a 4 flip-flops ($2^4=16$) e fazer-se o reset simultâneo destes, quando as saídas Q dos geradores de B1 e B3, estiverem a 1, sinónimo de que se atingiu o limite imposto para a contagem. As portas AND acima dos flip-flops vão deslocando o nível alto para a entrada T do flip-flop seguinte (da esquerda para a direita), tendo a do 1º sido fixada a esse nível, ao ritmo do clock aplicado a todos, com resposta no flanco ascendente, isto é, transições de 0 para 1. Esta resposta torna inadequado que se faça a saída Cout a partir da saída Q do último flip-flop, pois essa transição ocorre aí quando o contador atinge 1000, e não é esse o valor de transição, pelo que Cout é recolhido na saída /Q, que transitará de 0 para 1, quando for aplicado o reset a todos os flip-flops, reiniciando a contagem desse dígito, e então é feito o transporte para um contador seguinte permitindo adicionar mais contadores ao circuito, sendo possível assim representar-se, um dígito para as dezenas, centenas, milhares... se forem adicionados mais 1, 2, 3... contadores de 4 bits, ligando a saída Cout do contador das unidades, à entrada Input do contador das dezenas, e assim sucessivamente. Foi adicionada ao contador uma porta OR com 2 entradas, que recebe numa delas o sinal de reset de contagem, e na outra entrada um reset externo que pode ser ativado em qualquer momento pelo utilizador. Associando 2 contadores módulo 10 criou-se o contador Cont.99 modular, que conta até 99 inclusive, e com saída Cout que o torna expansível. Para realizar o Cont.99, integrado no CNT99, foram utilizadas a 2 portas AND

que tinham ficado disponíveis depois de concebido o TD105G, e foram necessários 1x 7432 (4 OR) ficando livres 2 portas. Quanto aos flip-flops, não encontrei paralelo com a realidade pois tipicamente as entradas para reset são ativas ao nível baixo e no DigitalWorks são-no ao nível alto, pelo que se poderia usar um 7402 que tem 4 portas NOR, em vez do 7432 (4 OR), para as entradas de RESET. Para os Flip-Flops D poderia ser usado 4x 7474 que contém 2 flip-flops D por IC, com clock ativo no flanco ascendente, $\overline{PRESET}$ e $\overline{CLEAR}$.



Foram acrescentados 2 conversores BCD para mostrar os 2 dígitos num mostrador LED de 7 segmentos, que tinha sido previamente realizado no decurso das atividades formativas (ver anexo B) para completar o CNT99.



D)

Encontra-se no Anexo C o circuito, a tabela de verdade, os mapas de Karnaugh e as expressões booleanas que definiram a construção da máquina de estados, com recurso a 3 flip-flops D, pois $k = \log_2(8) = 3$, e codificação binária dos estados.

Dada a complexidade de resolução do problema e o receio de insuficiência de tempo, não foram abordadas outras possibilidades de resolução que eventualmente a tornassem mais simples, embora a abordagem com recurso a básculas tipo D seja mais simples do que o recurso a JK, porque a tabela de transições para básculas D já contém a informação necessária, dado que do estado seguinte se extraem as condições necessárias às entradas, e com codificação binária temos uma báscula por estado. Tão pouco foram abordadas realizações de controladores com recurso a contadores ou microprogramados.

Na máquina de estados, as transições para estado seguinte, dependem do estado atual e da entrada F, que comutará o estado atual para um de dois possíveis estados seguintes, consoante $F=0$ ou $F=1$.

Pela tabela seguinte verifica-se que as únicas ocorrências de contagem vão dar-se quando se entra no estado S3, via S1, ou no estado S7, via S6, uma vez que a variável E a 0 permite $S=F(A,B,C,D)$, e $E=1$ inverte o valor da função. Por exemplo, quando no estado S3 ocorre $S=0$ à saída do TD105F, e $E=1$, S vai ser invertido, pelo que chega 1 ao contador. No estado S1, à saída do TD105F temos $S=1$, mas com $E=1$ a chegar ao TD105G, a sua saída vai ser uma inversão de S, pelo que fica a 0, não contando.

Estados		codificação do estado	Saídas					TD105F					TD105G	
								in				out	in	
			A	B	C	D	E	A	B	C	D	S	E	out
S0	000/01011	0 0 0	0	1	0	1	1	0	1	0	1	1	1	0
S1	001/10011	0 0 1	1	0	0	1	1	1	0	0	1	1	1	0
S2	010/01000	0 1 0	0	1	0	0	0	0	1	0	0	0	0	0
S3	011/10101	0 1 1	1	0	1	0	1	1	0	1	0	0	1	1
S4	100/10110	1 0 0	1	0	1	1	0	1	0	1	1	0	0	0
S5	101/10100	1 0 1	1	0	1	0	0	1	0	1	0	0	0	0
S6	110/11001	1 1 0	1	1	0	0	1	1	1	0	0	1	1	0
S7	111/00110	1 1 1	1	0	1	1	0	1	0	1	1	1	0	1

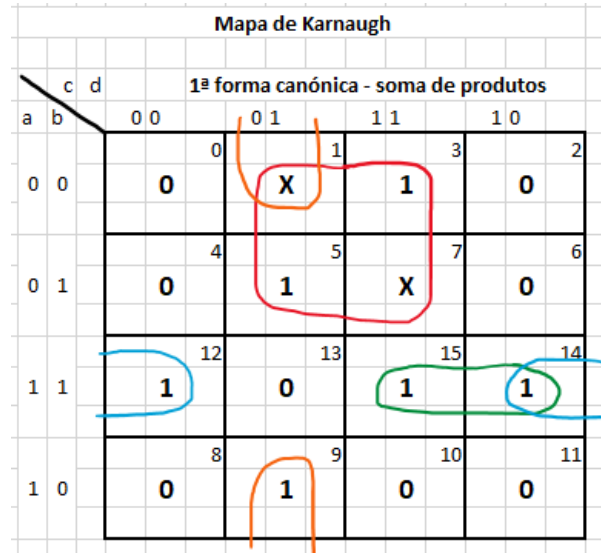
Foi alterada a referencia a dar ao componente que agrega a máquina de estados, para BK200X-B, por se lhe ter acrescentado 3 saídas que derivam das básculas, por forma a poder mostrar-se no circuito final, o estado em que a máquina se encontra, via mostrador LED de 7 segmentos.

Anexo A

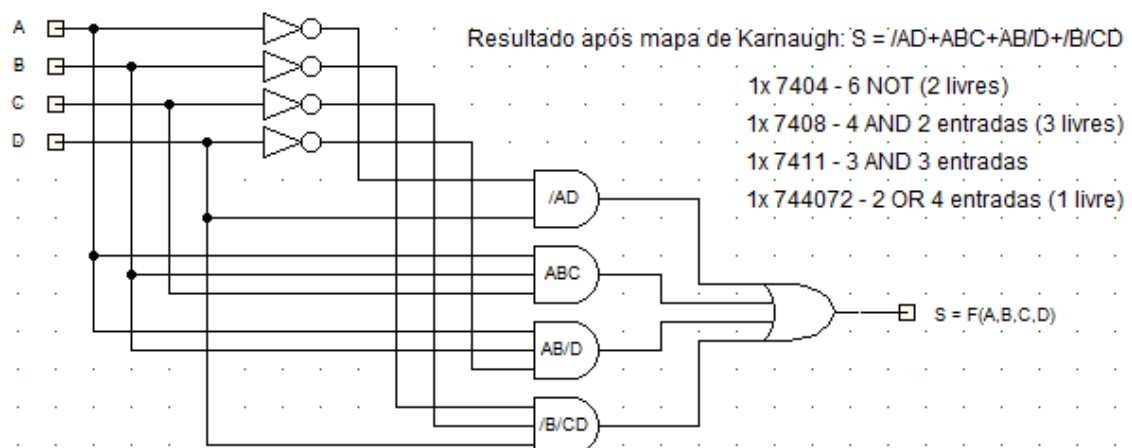
Resoluções alternativas para a alínea A)

Tabela de verdade

termos	A	B	C	D	S
0	0	0	0	0	0
1	0	0	0	1	X
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	X
8	1	0	0	0	0
9	1	0	0	1	1
10	1	0	1	0	0
11	1	0	1	1	0
12	1	1	0	0	1
13	1	1	0	1	0
14	1	1	1	0	1
15	1	1	1	1	1



1ª forma canónica: $S = F(A, B, C, D) = \sum m(1, 3, 5, 7, 9, 12, 14, 15) =$
 $= \bar{A}D + ABC + AB\bar{D} + \bar{B}CD$

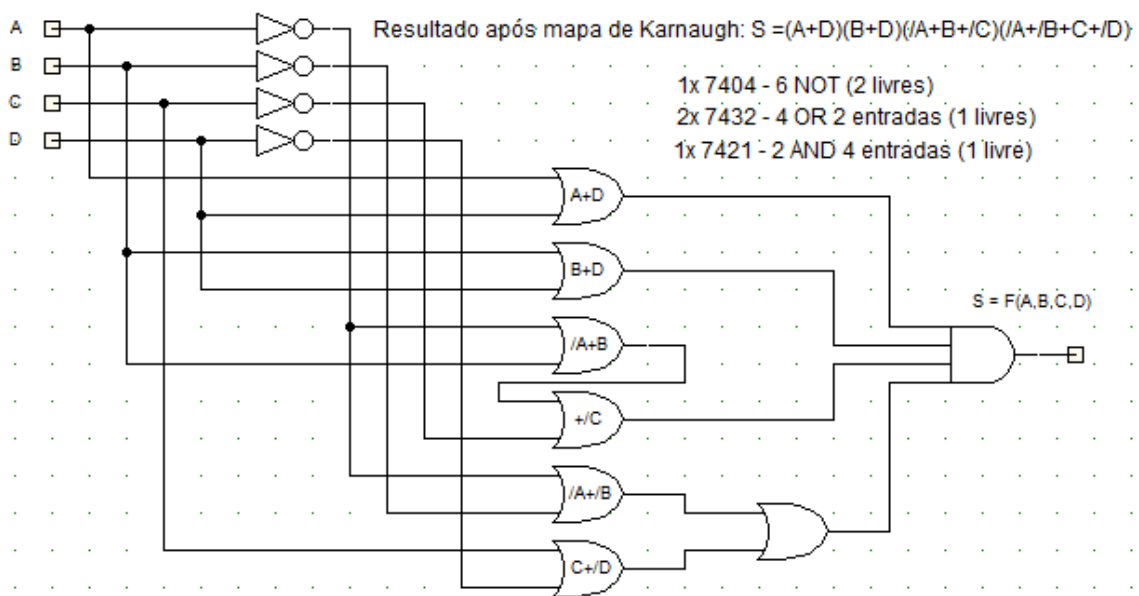


2ª forma canónica:

		2ª forma canónica - produto de somas			
		c d			
a b		0 0	0 1	1 1	1 0
0 0	0	0	X	1	0
0 1	0	0	1	X	0
1 1	1	0	1	1	1
1 0	0	0	1	0	0

$$S = F(A, B, C, D) = \Pi M(0, 2, 4, 6, 8, 10, 11, 13) =$$

$$= (A + D)(B + D)(\bar{A} + B + \bar{C})(\bar{A} + \bar{B} + C + \bar{D})$$

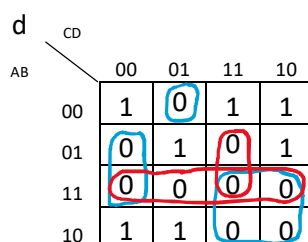
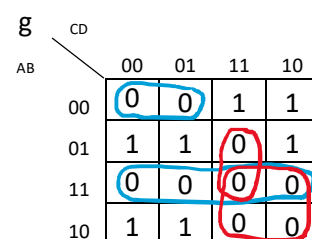
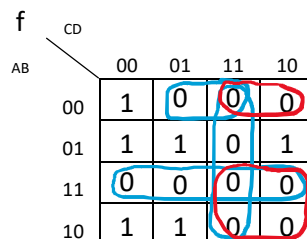
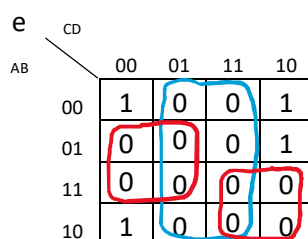
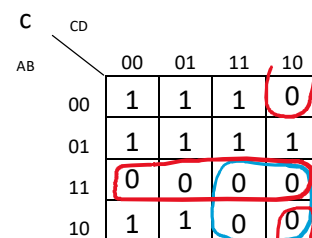
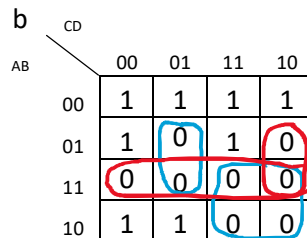
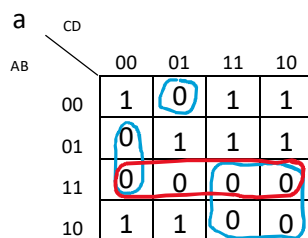
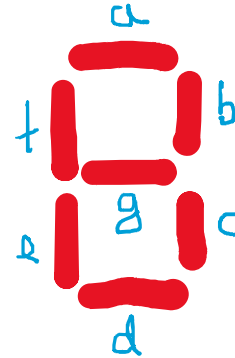


ICs: 1x7404 (2 livres), 1x 7432 (2 das portas para $(\bar{A} + B + \bar{C})$), 1x744072 ou outro 7432 (3 portas para $(\bar{A} + \bar{B} + C + \bar{D})$; 1 livre), 1x 7421 (1 livre)

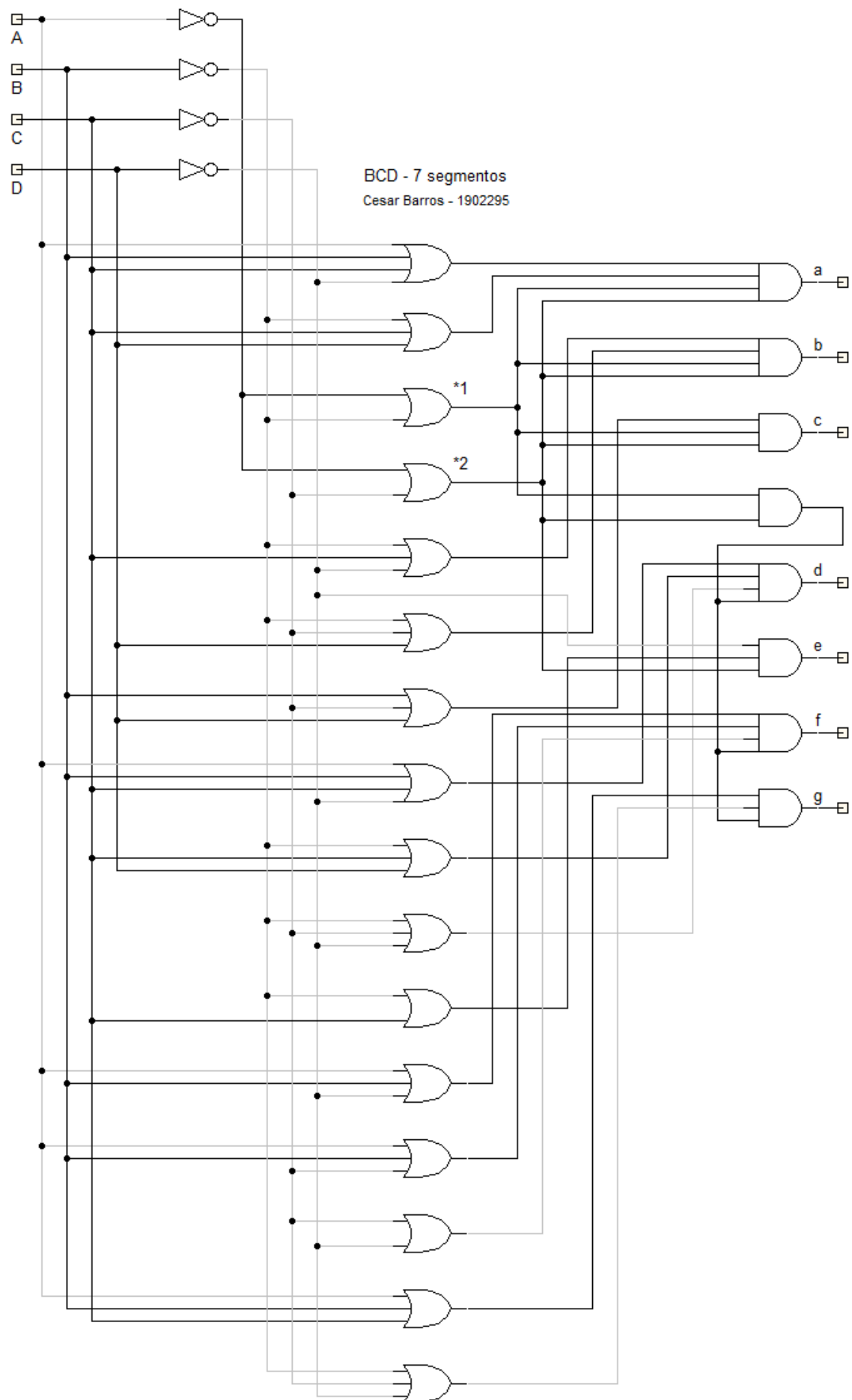
Anexo B

Conversor BCD para LED 7 segmentos (elaborado durante as AFs)

Dig	BCD				LED 7 segmentos						
	A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1
na	1	0	1	0	0	0	0	0	0	0	0
na	1	0	1	1	0	0	0	0	0	0	0
na	1	1	0	0	0	0	0	0	0	0	0
na	1	1	0	1	0	0	0	0	0	0	0
na	1	1	1	0	0	0	0	0	0	0	0
na	1	1	1	1	0	0	0	0	0	0	0



$$\begin{aligned}
 a &= (A + B + C + \bar{D})(\bar{B} + C + D)(\bar{A} + \bar{B})(\bar{A} + \bar{C}) \\
 b &= (\bar{B} + C + \bar{D})(\bar{B} + \bar{C} + D)(\bar{A} + \bar{B})(\bar{A} + \bar{C}) \\
 c &= (B + \bar{C} + D)(\bar{A} + \bar{B})(\bar{A} + \bar{C}) \\
 d &= (A + B + C + \bar{D})(\bar{B} + C + D)(\bar{B} + \bar{C} + \bar{D})(\bar{A} + \bar{B})(\bar{A} + \bar{C}) \\
 e &= \bar{D}(\bar{B} + C)(\bar{A} + \bar{C}) \\
 f &= (A + B + \bar{D})(A + B + \bar{C})(\bar{C} + \bar{D})(\bar{A} + \bar{B})(\bar{A} + \bar{C}) \\
 g &= (A + B + C)(\bar{B} + \bar{C} + \bar{D})(\bar{A} + \bar{B})(\bar{A} + \bar{C})
 \end{aligned}$$



Anexo C

Máquina de estados projetada com flip-flops D e codificação binária. 8 estados implicam o recurso a 3 flip-flops [$k=\log_2(8)=3$]

Est.	Est. Atual			In	Est. Seg.			Saidas					Est Seg
	Q2	Q1	Q0		Q2	Q1	Q0	A	B	C	D	E	
S0	0	0	0	0	0	0	1	0	1	0	1	1	S1
	0	0	0	1	0	1	0	0	1	0	1	1	S2
S1	0	0	1	0	1	0	0	1	0	0	1	1	S4
	0	0	1	1	0	1	1	1	0	0	1	1	S3
S2	0	1	0	0	1	0	0	0	1	0	0	0	S4
	0	1	0	1	1	0	1	0	1	0	0	0	S5
S3	0	1	1	0	1	1	1	1	0	1	0	1	S7
	0	1	1	1	0	0	0	1	0	1	0	1	S0
S4	1	0	0	0	1	0	1	1	0	1	1	0	S5
	1	0	0	1	0	0	0	1	0	1	1	0	S0
S5	1	0	1	0	1	1	0	1	0	1	0	0	S6
	1	0	1	1	0	1	0	1	0	1	0	0	S2
S6	1	1	0	0	1	0	0	1	1	0	0	1	S4
	1	1	0	1	1	1	1	1	1	0	0	1	S7
S7	1	1	1	0	0	1	1	0	0	1	1	0	S3
	1	1	1	1	1	0	1	0	0	1	1	0	S5

D2

Q0 F

Q2Q1

	00	01	11	10
00	0	0	0	1
01	1	1	0	1
11	1	1	1	0
10	1	0	0	1

D1

Q0 F	00	01	11	10	
Q2Q1	00	0	1	1	0
	01	0	0	0	1
	11	0	1	0	1
	10	0	0	1	1

D0

Q0 F

Q2Q1

	00	01	11	10
00	1	0	1	0
01	0	1	0	1
11	0	1	1	1
10	1	0	0	0

A

Q0 F	00	01	11	10	
Q2Q1	00	0	0	1	1
	01	0	0	1	1
	11	1	1	0	0
	10	1	1	1	1

B

Q0 F

Q2Q1

	00	01	11	10
00	1	1	0	0
01	1	1	0	0
11	1	1	0	0
10	0	0	0	0

C

Q0 F	00	01	11	10	
Q2Q1	00	0	0	0	0
	01	0	0	1	1
	11	0	0	1	1
	10	1	1	1	1

D

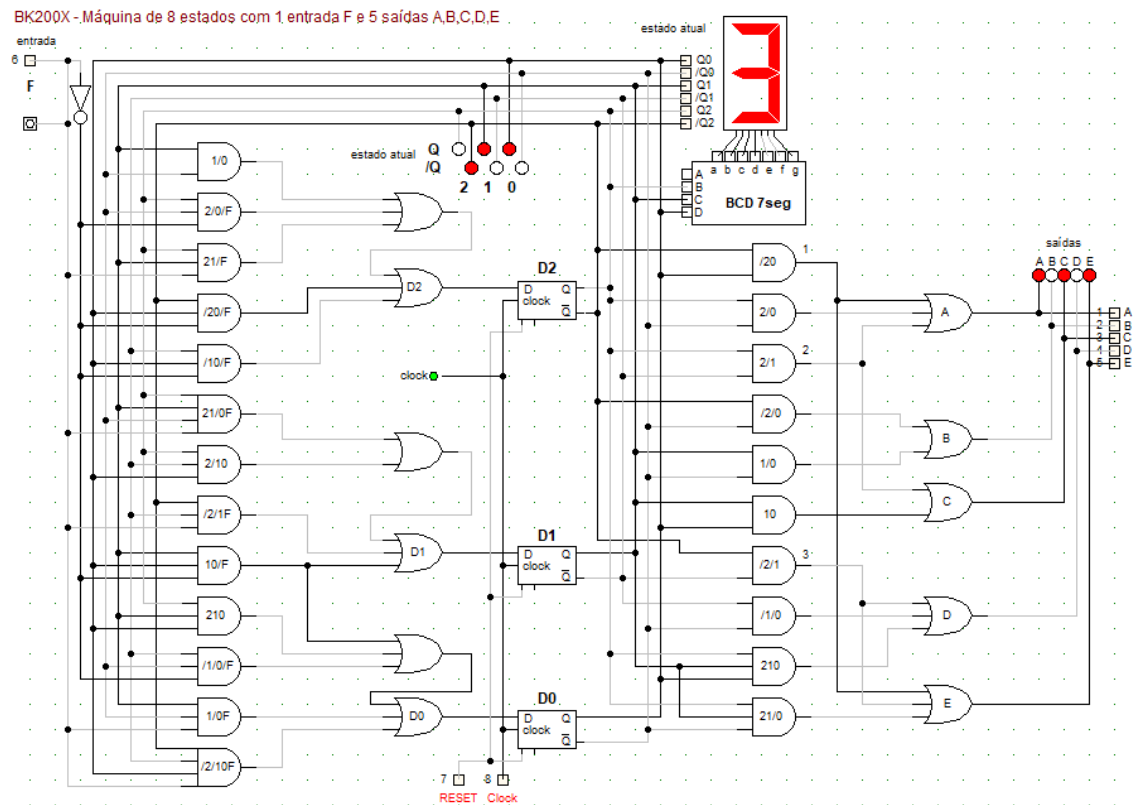
Q0 F	00	01	11	10	
Q2Q1	00	1	1	1	1
	01	0	0	0	0
	11	0	0	1	1
	10	1	1	0	0

E

Q0 F	00	01	11	10	
Q2Q1	00	1	1	1	1
	01	0	0	1	1
	11	1	1	0	0
	10	0	0	0	0

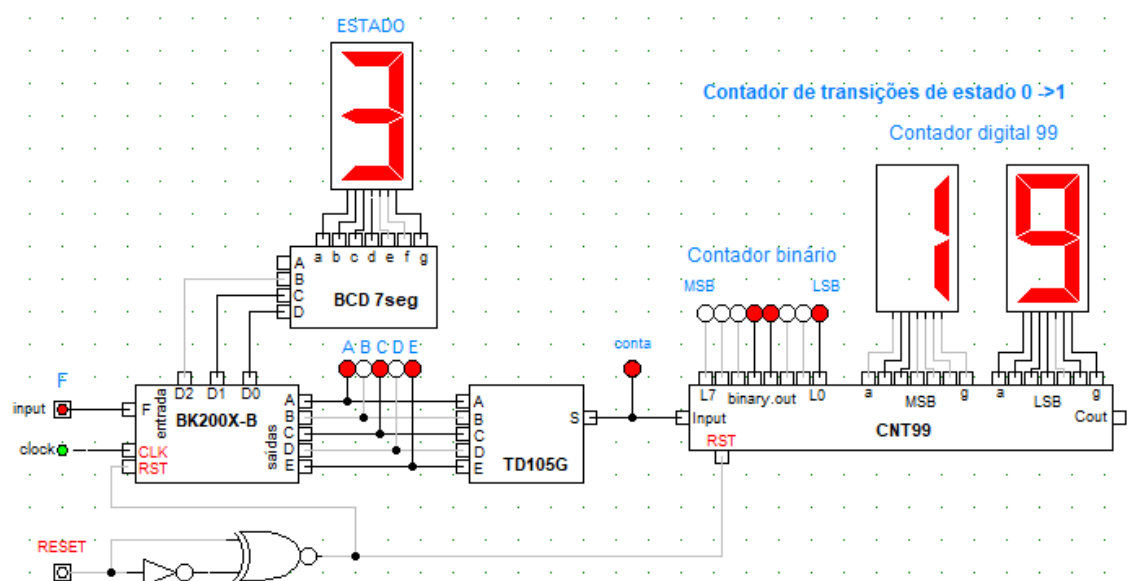
$$\begin{aligned}
 D2 &= Q_1 \overline{Q_0} + Q_2 \overline{Q_0} \overline{F} + Q_2 Q_1 F + \overline{Q_2} Q_0 \overline{F} + \overline{Q_1} Q_0 \overline{F} \\
 D1 &= \overline{Q_2} \overline{Q_1} F + Q_1 Q_0 \overline{F} + Q_2 Q_1 \overline{Q_0} F + Q_2 \overline{Q_1} Q_0 \\
 D0 &= \overline{Q_1} \overline{Q_0} \overline{F} + Q_1 \overline{Q_0} F + \overline{Q_2} \overline{Q_1} Q_0 F + Q_2 Q_1 Q_0 + Q_1 Q_0 \overline{F} \\
 A &= \overline{Q_2} Q_0 + Q_2 \overline{Q_0} + Q_2 \overline{Q_1} \\
 B &= \overline{Q_2} \overline{Q_0} + Q_1 \overline{Q_0} \\
 C &= Q_1 Q_0 + Q_2 \overline{Q_1} \\
 D &= \overline{Q_2} \overline{Q_1} + \overline{Q_1} \overline{Q_0} + Q_2 Q_1 Q_0 \\
 E &= \overline{Q_2} \overline{Q_1} + \overline{Q_2} Q_0 + Q_2 Q_1 \overline{Q_0}
 \end{aligned}$$

Havendo repetições nas expressões das variáveis de saída, houve portas que foram reaproveitadas, de acordo com a diferenciação de cores nas expressões.



Anexo D

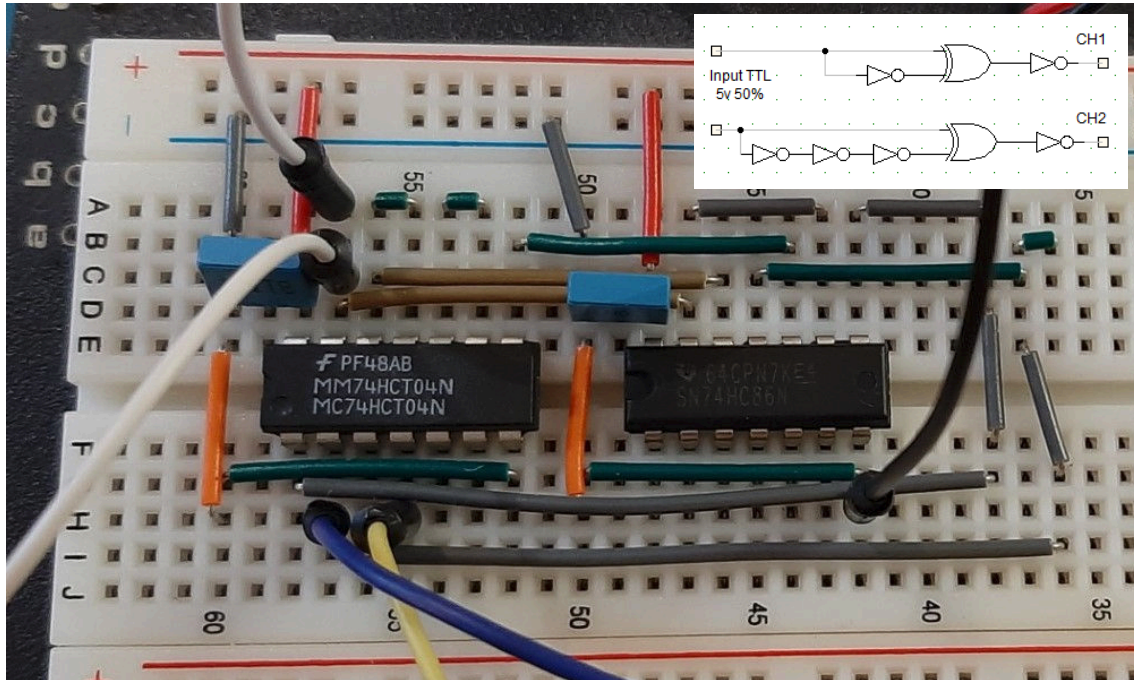
Circuito global composto pela máquina de estados a que se deu o nome de BK200X-B, por conter informação das saídas das básculas D_0 , D_1 e D_2 , por forma a informar via mostrador LED de sete segmentos, o estado atual da máquina. Os LEDs nas linhas que ligam BK200X-B ao circuito lógico TD105G mostram o estado das variáveis A, B, C, D e E, e à saída deste, outro LED indica se ocorre impulso de contagem para o contador de transições de estado de 0 para 1, com 2 mostradores LED de sete segmentos onde se mostra a contagem digital, possível até 99, e também em formato binário por via de barra de 8 LEDs



Um pequeno circuito de RESET foi adicionado, permitindo que o reset da máquina de estados e contadores se faça, pulsando o botão, sem necessidade de preocupação em desligar para não manter o reset ativo, pois apenas um pulso é enviado, quer quando se liga, ou desliga o botão. Isto é conseguido devido ao atraso provocado pelo NOT, que faz com que momentaneamente à entrada do XNOR, os valores sejam iguais, produzindo apenas um pulso positivo.

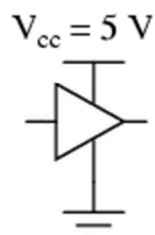
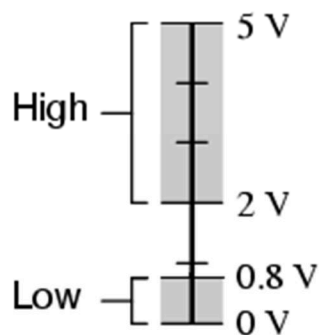
Num circuito real, o funcionamento desta implementação está dependente de diversas temporizações, nomeadamente do tempo de propagação da porta NOT, pois se for muito rápida a propagar para a saída a inversão do sinal alterado à entrada, pode nunca ocorrer o diferencial pretendido à entrada do XNOR (valores iguais em ambas as entradas) de forma a produzir o pulso pretendido à saída.

A título de curiosidade foram realizados alguns testes práticos, para o que se implementaram dois circuitos de simulação adaptados, compostos por um 74HCT04 (6 NOT) e um 74HC86 (4 XOR), por não haver disponível um XNOR.

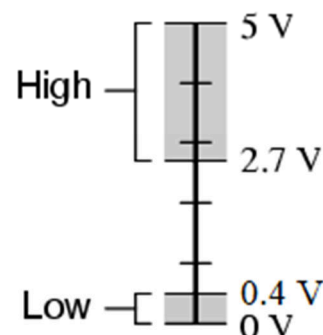


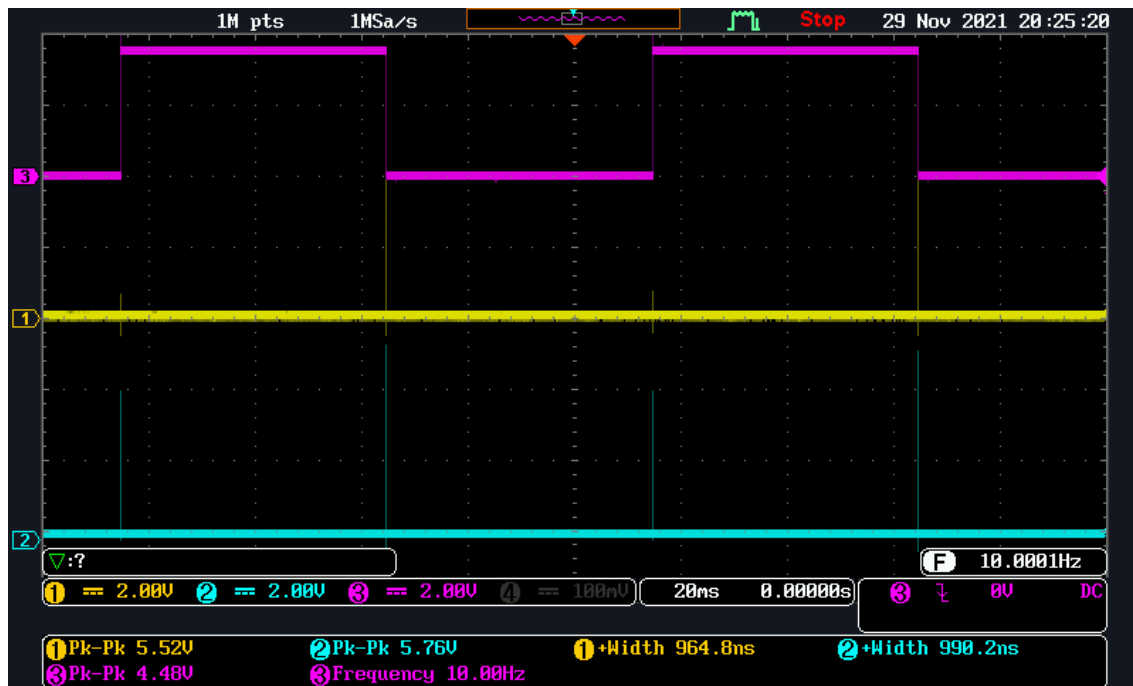
Sem grande rigor pretendia-se verificar na prática se seria exequível implementar um impulso de RESET por esta via, com valores aceitáveis para a norma TTL, e se apenas com um NOT se consegue o objetivo, ou seriam necessários 3 NOT seguidos. Mais haveria a dizer, e a considerar, como variações de resposta a diferentes frequências, ICs de diferentes fabricantes...

Acceptable TTL gate input signal levels



Acceptable TTL gate output signal levels





Na imagem de osciloscópio temos no canal 3 (CH3-rosa) o sinal retangular de 4.5Vpp com fator de trabalho 50%, aplicado à entrada dos circuitos; no canal 1 (CH1-amarelo) temos a saída do circuito que passa por 1 NOT aplicado numa das entradas do XOR, com outro NOT à saída (configuração XNOR); no canal 2 (CH2-azul) temos a saída quando numa das entradas do XOR estão 3 NOT seguidos.

Com um só NOT à entrada, temos à saída um pulso com boa amplitude no flanco descendente do sinal de entrada, mas uma amplitude reduzida quando à entrada ocorre uma transição de 0 para 1, sinónimo de um diferencial dos tempos de propagação do NOT, a cada um dos dois estados de transição, sendo que o NOT é mais rápido a apresentar uma inversão de sinal à saída quando ocorre a transição de 0 para 1 à sua entrada.

Com 3 NOT seguidos, canal 2 do osciloscópio, teoricamente triplicamos o tempo de propagação e por conseguinte damos mais tempo ao XOR para ter um sinal estável diferente entre as entradas, produzindo à saída do circuito pulsos com amplitudes mais regulares, quer com transições do sinal de entrada, de 0 para 1, quer de 1 para 0. Em suma, com 3 portas NOT e com impulsos de baixa frequência, típicos do pulsar de um comutador por um humano, será garantido um sinal de reset estável.