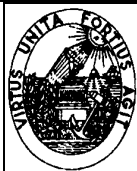


NOME: _____ 2ª CHAMADA - CORRECÇÃO _____ TURMA _____

**Universidade do Porto**

Faculdade de Engenharia

Licenciatura em Engenharia Electrotécnica e de Computadores

Sistemas Digitais (1999/2000)**2ª chamada - 5/Julho/2000****Duração: 2h 30m, sem consulta.****Antes de iniciar a prova, tenha em atenção as seguintes recomendações:**

- **Leia atentamente** toda a prova antes de a iniciar.
- **Justifique adequadamente** todos os passos das suas respostas.
- **A prova deverá ser resolvida no enunciado**, uma questão por folha. Se necessário, utilize o verso de cada folha para continuar a sua resolução.
- **Entregue todas as folhas do enunciado**, não se esquecendo de preencher o seu nome.

1 -

- a) Indique justificando o resultado da adição binária dos números $A=011001$ e $B=110011$, ambos representados em

Efectuando a adição binária dos dois números dados obtemos:

$$\begin{array}{r} 011001 \\ +110011 \\ \hline 1001100 \end{array}$$

Apesar do resultado obtido ser 1001100, o bit 1 da esquerda deve ser desprezado já que como ambos os operandos estão representados em complemento para dois com 6 bits, o resultado será também representado em complemento para dois, constituído apenas pelos 6 bits menos significativos: 001100. Para confirmar o resultado podemos representar os operandos em base 10:

$$\begin{aligned} A &= 011001 = 16+8+1 = +25 \\ B &= 110011 = -(001100+1) = -(001101) = -13 \\ A+B &= 25-13 = +12 = 001100 \end{aligned}$$

- b) Adicionando os números $X=0101$ e $Y=1111$ obtém-se um resultado com 4 bits igual a 0100. Diga, justificando, se ocorre *overflow* nessa operação se X e Y representarem números:

Antes de responder às questões seguintes é conveniente efectuar a operação de adição referida:

$$\begin{array}{r} 0101 \\ +1111 \\ \hline 10100 \end{array}$$

obtendo, como é dito no enunciado, um resultado em 4 bits igual a 0100

- i) inteiros em base 2 com 4 bits e sem sinal

Se os dois operandos representarem inteiros em base 2 com 4 bits ocorre overflow na operação referida porque é gerado um carry na soma dos bits mais significativos dos operandos. Por outras palavras, o resultado (positivo) da adição binária não pode ser representado por 4 bits.

- ii) inteiros em complemento para dois com 4 bits

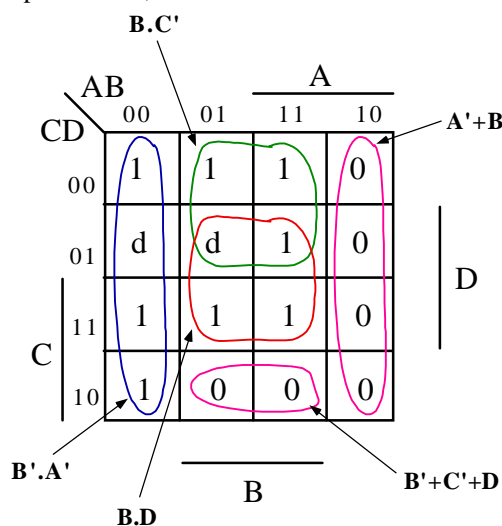
Se os dois operandos representarem inteiros em complemento para dois com 4 bits não ocorre overflow porque X e Y têm sinais contrários. Confirmando: $X=+5$, $Y=-1$ e $X+Y=+4$

NOME: _____ 2ª CHAMADA - CORRECÇÃO _____ TURMA _____

2 - O mapa de Karnaugh da figura representa uma função booleana $F(A,B,C,D)$

AB		A			
		00	01	11	10
CD	00	1	1	1	0
	01	d	d	1	0
	11	1	1	1	0
	10	1	0	0	0

a) Obtenha a expressão mais simples do tipo POS ou SOP, considerando que pretende uma realização com portas lógicas do tipo inversor, AND ou OR com duas entradas. Note que não se pretende que desenhe o circuito lógico.



Se o objectivo é obter a realização mais simples com inversores e portas lógicas com duas entradas, vamos escolher a expressão SOP ou POS que minimize o número de operadores lógicos dos tipos referidos. Agrupando os zeros obtemos a expressão do tipo POS:

$$F(A,B,C,D) = (B' + C' + D) \cdot (A' + B)$$

que necessita de 3 inversores, 3 OR e um AND

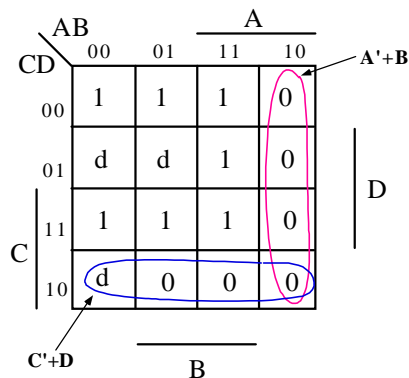
Agrupando os uns obtemos a expressão do tipo SOP:

$$F(A,B,C,D) = B' \cdot A' + B \cdot D + B \cdot C'$$

que utiliza 3 inversores, 3 AND e dois OR.

Assim, a representação mais simples para $F(A,B,C,D)$ é a expressão POS apresentada acima.

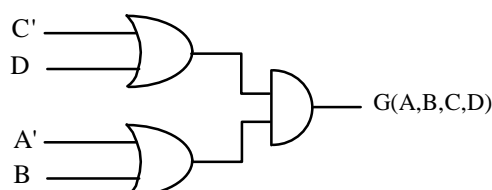
b) Considere agora uma função $G(A,B,C,D)$ que é idêntica a F , excepto no termo $ABCD=0010$ em que é indiferente. Construa um circuito minimizado utilizando apenas portas lógicas NAND de 2 entradas que realize essa função.



O objectivo é minimizar o número de portas lógicas NAND de duas entradas. Para construir um circuito lógico só com portas NAND é geralmente mais conveniente construir primeiro um circuito do tipo AND-OR, obtido por tradução directa da expressão SOP minimizada. No entanto, se a função $G(A,B,C,D)$ for considerada zero nesse termo, a expressão POS resulta mais simples do que a expressão SOP já que apenas requer dois termos de soma com dois literais cada um:

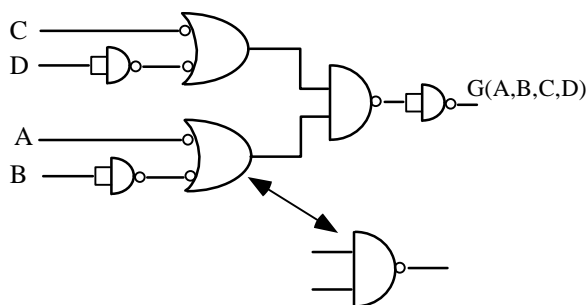
$$G(A,B,C,D) = (C' + D) \cdot (A' + B)$$

O circuito OR-AND que realiza aquela expressão é:



NOME: _____ 2ª CHAMADA - CORRECÇÃO _____ TURMA _____

podendo ser transformado no circuito equivalente usando apenas 6 portas NAND de duas entradas:

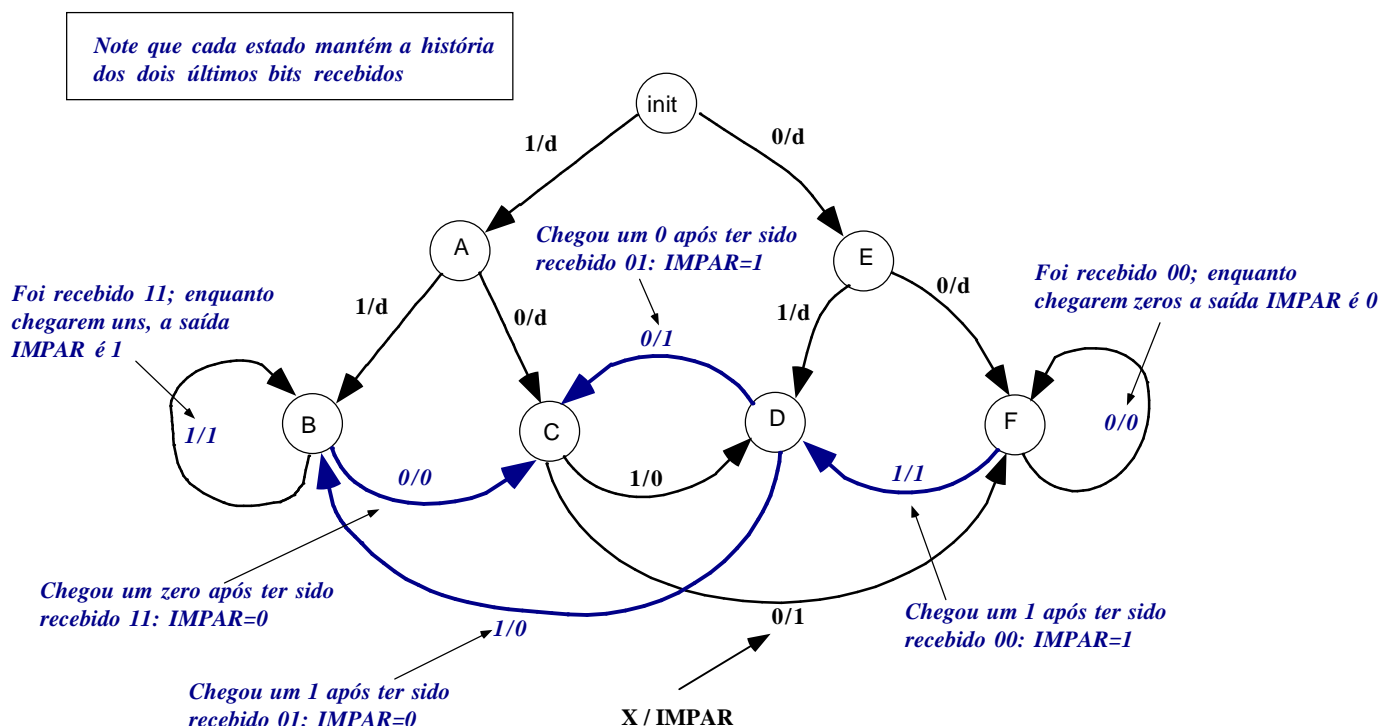


A expressão mais simples do tipo SOP seria igual à apresentada em a). Podemos concluir (sem desenhar o circuito!) que só para realizar os 3 termos de produto dessa expressão SOP seriam necessárias 6 portas NAND de duas entradas: 3 NANDs para as negações e mais 3 NANDs para realizar os 3 termos de produto. Como para além disso é ainda necessário realizar a soma lógica dos termos de produto, o número total de NANDs de duas entradas será superior ao número utilizado no circuito apresentado.

3 - Pretende-se construir uma máquina de estados de Mealy com uma saída IMPAR que é 1 quando os 3 últimos bits consecutivos colocados na sua única entrada X incluírem um número ímpar de uns, e zero no caso contrário. Depois do início do funcionamento da máquina de estados, a saída IMPAR só é considerada válida após o 3º ciclo de relógio. A figura mostra um exemplo de uma sequência de bits na entrada X e o valor correspondente para a saída IMPAR

X: 0101101011110100
 IMPAR: dd10000100110011
 ↑
 saída IMPAR é indiferente
 nos 2 primeiros ciclos

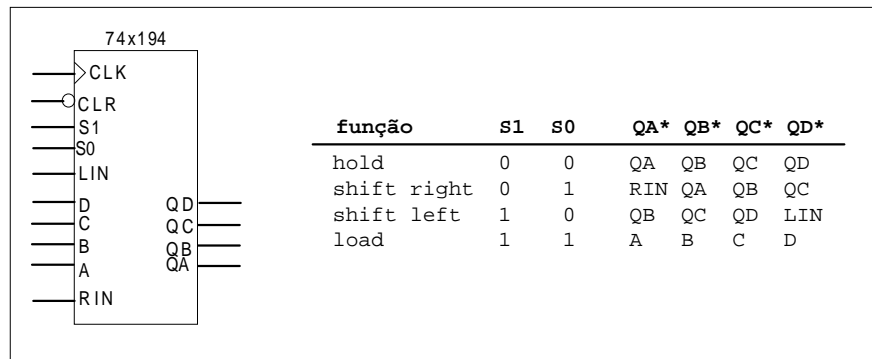
a) A figura representa um diagrama de transição de estados incompleto para a máquina de Mealy referida. Complete-o indicando claramente as transições de estado, condições de transição de estado e valores para a saída IMPAR que faltam.



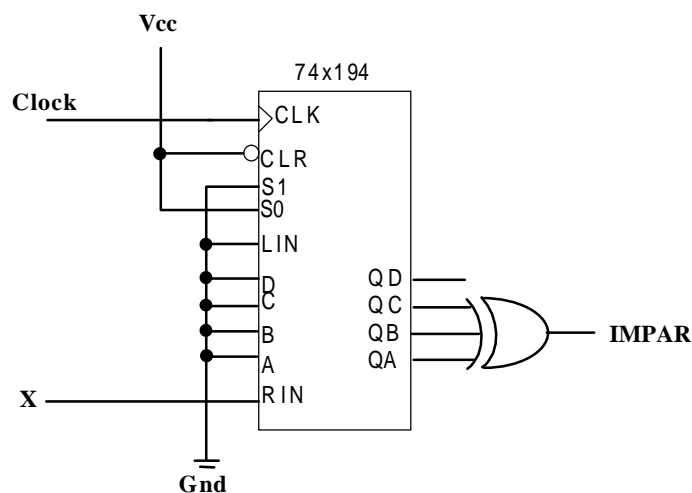
NOME: _____ 2ª CHAMADA - CORRECÇÃO _____ TURMA _____

b) Sabendo que uma porta XOR de 3 entradas realiza a função detecção de paridade ímpar de uma palavra de 3 bits (ver tabela), construa um circuito utilizando um 74x194 (universal shift-register) que realize a funcionalidade da máquina de estados referida.

A	B	C	$A \oplus B \oplus C$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1



Se o shift-register for configurado em modo shift-right ($S1=0$, $S0=1$) sendo a entrada RIN a entrada X da máquina pretendida, as saídas QA, QB e QC têm, em cada ciclo, os últimos 3 bits recebidos. Se essas saídas do shift-register forem ligadas às entradas do XOR de 3 entradas, a sua saída será 1 quando os últimos 3 bits recebidos apresentarem paridade ímpar. Assim, o circuito completo será:



NOME: _____ 2ª CHAMADA - CORRECÇÃO _____ TURMA _____

4 - A figura seguinte representa a tabela de transição de estados de uma máquina de Moore.

Estado S	Entrada X		Saídas	
	X=0	X=1	Y	Z
E1	E2	E1	0	0
E2	E2	E3	1	1
E3	E3	E4	0	1
E4	E1	E5	0	1
E5	E5	E3	1	1

próximo estado S*

a) Apresente e justifique uma codificação de estados que permita minimizar a complexidade do circuito lógico que realiza as saídas Y e Z.

Como na máquina de Moore as saídas apenas dependem do estado presente, a codificação de estados que minimiza a complexidade do circuito lógico que realiza as saídas Y e Z pode ser obtida incluindo os valores lógicos para as saídas Y e Z no código atribuído a cada estado. Nesse caso não é necessária qualquer função lógica, sendo as saídas iguais a bits da variável de estado. Como temos 5 estados e necessitamos de (pelo menos) 3 bits para os codificar, podemos codificar os estados em Q2, Q1 com os valores pretendidos para as saídas Y e Z, utilizando o 3º bit Q0 para diferenciar entre estados que apresentem saídas iguais:

Fazendo Q2, Q1 iguais a Y, Z

Estado S	Y Z		Q0
	Q2	Q1	
E1	0	0	0
E2	1	1	0
E3	0	1	0
E4	0	1	1
E5	1	1	1

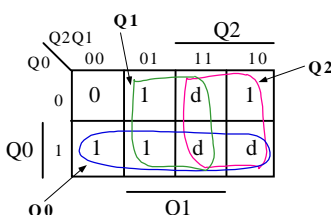
b) Considere a codificação de estados seguinte:

Estado S	Q2, Q1, Q0		
E1	0	0	0
E2	0	0	1
E3	0	1	0
E4	0	1	1
E5	1	0	0

Obtenha uma expressão lógica minimizada para a saída $Z=f(Q2, Q1, Q0, X)$. Admita que Z é indiferente para os estados não especificados na tabela de transição de estados.

Utilizando a codificação de estados dada, vamos construir a tabela de verdade da função $Z=f(Q2, Q1, Q0)$. Note que a saída Z não depende da entrada X, uma vez que se trata de uma máquina de Moore:

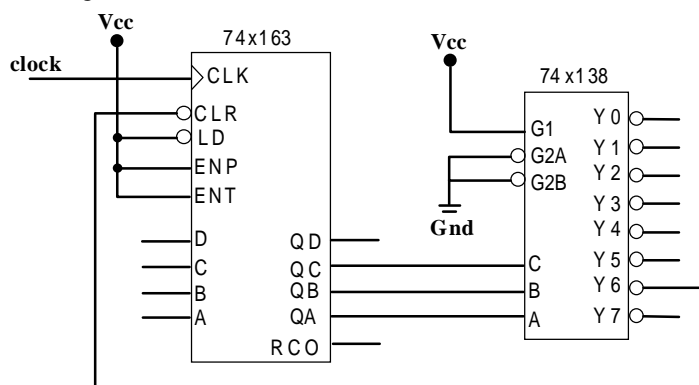
Construindo o mapa de Karnaugh para a função $Z(Q2, Q1, Q0)$:



estado presente			saídas	
Q2	Q1	Q0	Y	Z
0	0	0	0	0
0	0	1	1	1
0	1	0	0	1
0	1	1	0	1
1	0	0	1	1
1	0	1	x	x
1	1	0	x	x
1	1	1	x	x

obtem-se a expressão simplificada (soma-de-produtos): $Z=Q2+Q1+Q0$

NOME: _____ 2ª CHAMADA - CORRECÇÃO _____ TURMA _____

5 - Considere o circuito síncrono da figura, realizado em torno de um contador binário 74x163 e de um decodificador 74x138

74x163												74x138													
												G1	/G2A	/G2B	C	B	A	/Y7	/Y6	/Y5	/Y4	/Y3	/Y2	/Y1	/Y0
/CLR	/LD	ENT	ENP	QD	QC	QB	QA	QD*	QC*	QB*	QA*	0	x	x	x	x	x	1	1	1	1	1	1	1	1
0	x	x	x	x	x	x	x	0	0	0	0	0	x	x	x	x	x	1	1	1	1	1	1	1	1
1	0	x	x	x	x	x	x	D	C	B	A	x	1	x	x	x	x	1	1	1	1	1	1	1	1
1	1	0	x	x	x	x	x	QD	QC	QB	QA	x	x	1	x	x	x	1	1	1	1	1	1	1	1
1	1	x	0	x	x	x	x	QD	QC	QB	QA	1	0	0	0	0	0	1	1	1	1	1	1	0	1
1	1	1	0	x	x	x	x	QD	QC	QB	QA	1	0	0	0	1	0	1	1	1	1	0	1	1	1
1	1	1	1	0	x	x	x	QD	QC	QB	QA	1	0	0	1	0	0	1	1	1	0	1	1	1	1
1	1	1	1	1	N (se N<15)			N + 1				1	0	0	1	1	0	1	0	1	1	1	1	1	1
1	1	1	1	1	1	1	1	0	0	0	0	1	0	0	1	1	1	0	1	1	1	1	1	1	1

a) Admitindo um estado inicial igual a $Q_D, Q_C, Q_B, Q_A = 0000$, determine a sequência de valores que ocorrem nas saídas do contador.

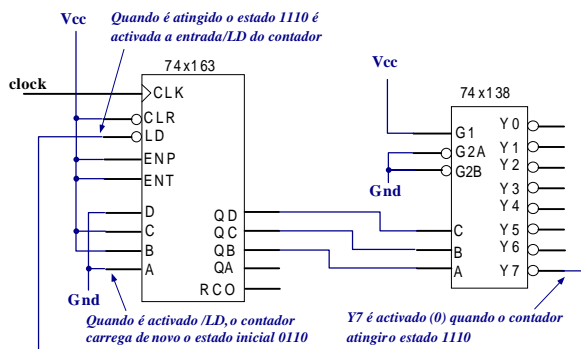
No circuito apresentado, a entrada /CLR do contador é activada quando o estado presente é $Q_D, Q_C, Q_B, Q_A = x110$ (ver tabela do decodificador 74x138). Admitindo o estado inicial igual a 0000 e como o 74x163 funciona como um contador binário, a entrada /CLR será activada logo que é atingido o estado 0110, sendo o estado seguinte igual a 0000. Assim, a sequência nas saídas do contador será:

0000, 0001, 0010, 0011, 0100, 0101, 0110, 0000, 0001, ...
 /CLR é activado aqui, ^^^^
 próximo estado é 0000

(ou em decimal: 0, 1, 2, 3, 4, 5, 6, 0, 1, 2 ...)

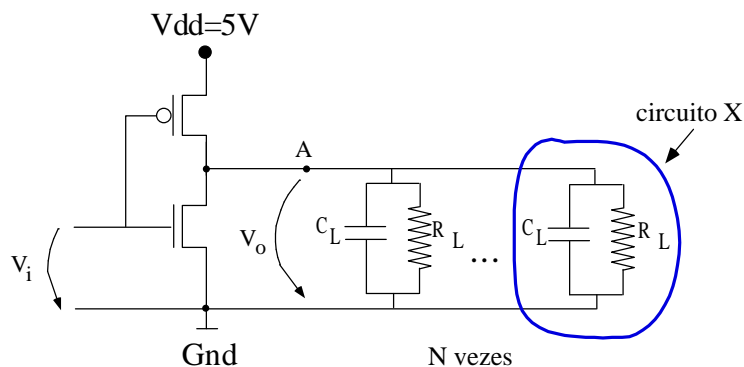
b) Utilizando apenas o contador 74x163 e o decodificador 74x138, modifique o circuito anterior de forma a obter um contador que conte ciclicamente de 6 a 14, admitindo que o estado inicial é $Q_D, Q_C, Q_B, Q_A = 0110$.

Admitindo que o estado inicial é $Q_D, Q_C, Q_B, Q_A = 0110$ (6 em decimal), é necessário utilizar o decodificador 74x138 para detectar o valor máximo da sequência pretendida (14 ou 1110 em binário), de forma a activar a entrada /LD do contador para carregar de novo o estado inicial 0110. Ligando os bits mais significativos das saídas do contador (Q_D, Q_C, Q_B) às entradas C, B, A do 74x138, o sinal /LD pretendido será a saída /Y7 do decodificador. Para que seja carregado o valor 0110 quando é activada a entrada /LD, é ainda necessário ligar as entradas D, C, B, A às constantes lógicas 1 e 0 apropriadas:



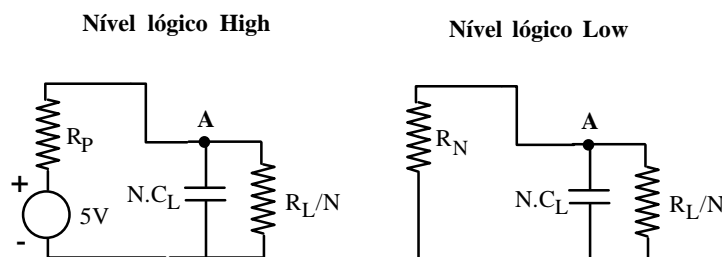
NOME: _____ 2ª CHAMADA - CORRECÇÃO _____ TURMA _____

6 - O circuito da figura representa um inversor CMOS cuja saída está ligada às entradas de N circuitos digitais do tipo X, conforme se mostra na figura. Cada entrada do circuito X pode ser representada pelo paralelo de uma resistência R_L e um condensador C_L .



a) Explique justificando, de que forma o número N de circuitos do tipo X afecta os níveis lógicos na saída do inversor (ponto A)

O circuito equivalente à associação de N circuitos do tipo X em paralelo é representado por uma resistência R_L/N em paralelo com um condensador $N.C_L$. Assim, os circuitos equivalentes para os 2 níveis lógicos são:



Os níveis lógicos (estáticos) na saída do inversor dependem apenas da carga resistiva vista da sua saída. O nível lógico low não será afectado com o número de circuitos do tipo X ligados à saída do inversor, já que a tensão no ponto A será sempre zero qualquer que seja N.

No nível lógico high, a tensão no ponto A será definida pelo divisor de tensão formado pela resistência equivalente do transistor PMOS no estado de condução (R_P na figura) e a resistência equivalente da associação em paralelo de N circuitos do tipo X (R_L/N na figura). Quando N aumenta R_L/N diminui e consequentemente a tensão no ponto A diminui reduzindo com isso a margem de ruído estática para o nível lógico alto.

b) Escreva a expressão que relaciona a constante de tempo associada ao tempo de subida da tensão V_o (no ponto A) com N, C_L , R_L e a resistência de condução do transistor PMOS, R_P

A análise do tempo de subida da tensão no ponto A é feita sobre o circuito equivalente para o nível lógico alto. A constante de tempo associada ao tempo de subida neste circuito será o produto da capacidade equivalente $N.C_L$ pela resistência vista dos terminais do condensador (R_L/N em paralelo com R_P):

$$\tau = N.C_L \times (R_L/N \parallel R_P) = N.C_L \times R_L/N \times R_P / (R_L/N + R_P)$$

-FIM -