

NOME: _____ CORRECÇÃO _____ TURMA _____

FEUP
Universidade do Porto
Faculdade de Engenharia

**Departamento de Engenharia Electrotécnica
e de Computadores**

Sistemas Digitais (2001/2002)

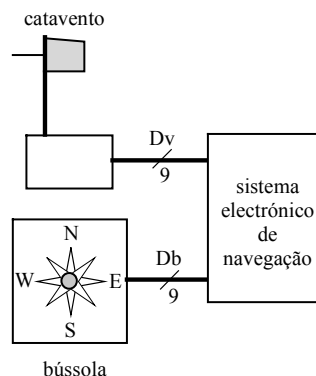
1ª chamada – 26/Junho/2002

Duração: 2 horas, sem consulta.

Antes de iniciar a prova, tenha em atenção as seguintes recomendações:

- Leia atentamente toda a prova antes de a iniciar.
- Mostre e justifique adequadamente todos os passos das suas respostas.
- A prova deverá ser resolvida no enunciado. Se necessário, utilize o verso para continuar a sua resolução.
- Assine todas as folhas que entregar, indicando em cada uma o número de páginas/folhas que entregou.

1 - O sistema electrónico de navegação de um barco à vela é composto, entre outros equipamentos, por uma bússola e um catavento electrónicos (ver figura). A bússola indica a direcção do norte (Db, entre -180 e +180 graus) e o catavento indica a direcção do vento (Dv, entre -180 e +180 graus), ambas em relação ao eixo longitudinal do barco.



- a) Mostre que são necessários 9 bits para representar em complemento para dois cada um dos valores referidos com uma resolução de 1 grau. Utilizando toda a gama permitida por esse número de bits, qual seria a resolução mínima (em graus) com que se poderiam representar essas grandezas?

Ambas as grandezas são representadas por números inteiros no intervalo $[-180, +180]$. Como o sistema complemento para dois com N bits permite representar números com sinal no intervalo $[-2^{(N-1)}, +2^{(N-1)}-1]$, para incluir o intervalo referido é necessário usar, no mínimo, 9 bits (com 9 bits podem-se representar valores inteiros no intervalo $[-256, +255]$):

$$N \geq \max(\log_2(180)+1, \log_2(180+1)+1), N \text{ inteiro} \rightarrow N \geq 9$$

Com 9 bits podem-se representar 511 intervalos $(255-(-256)=511)$; o intervalo referido contém 360 graus $(180-(-180)=360)$. Assim, a resolução mínima que 9 bits permitem representar será dado por:

$$360/511 = 0.70^\circ$$

- b) Quando o barco está a navegar para Este ($Db = -90^\circ$), qual é o valor binário produzido pela bússola electrónica, considerando que a unidade mínima representada é 1° ? E qual é esse valor se for utilizada toda a gama (9 bits) para representar os valores compreendidos entre -180° e $+180^\circ$?

Se a unidade mínima representada for 1° , então cada unidade do sistema de representação equivale a 1° e -90° é representado pelo número inteiro -90. Como em complemento para dois com 9 bits:

$$+90 = 001011010_2$$

a representação de -90 pode ser obtida trocando os bits todos e adicionando 1:

$$-90 = -(001011010_2) = 110100101_2 + 1 = 110100110_2$$

NOME: _____ **CORRECÇÃO** _____ TURMA _____

Se for usada toda a gama de representação permitida por números de 9 bits em complemento para dois, então o valor mais negativo da grandeza a representar (-180) corresponderá ao extremo mais negativo da gama representável por 9 bits (-256). Como $-90 = -180 / 2$, então a quantidade -90° seria representada por $-256/2 = -128$. A grandeza -128 representa-se em complemento para dois com 9 bits como:

$$-128 = - (010000000_2) = 101111111_2 + 1 = 110000000_2$$

- c) A direcção real do vento (em relação ao norte) pode ser obtida calculando a diferença entre os valores da direcção do barco (Db) e da direcção do vento em relação ao barco (Dv). Determine a direcção real do vento quando o barco navega para Sul (Db=-180°) e o catavento indica uma direcção do vento igual a Dv=+170°, efectuando a operação aritmética em binário com 9 bits. Comente o resultado obtido (utilize o verso da folha).

Pretende-se realizar a operação aritmética Db-Dv, com Db=-180 e Dv=+170. Como Db-Dv se pode escrever Db+(-Dv), vamos obter primeiro a representação de Db=-180 e de -Dv=-170 em complemento para dois com 9 bits, e depois realizar a operação de adição em vez da subtracção:

$$Db = -180 = - (010110100_2) = 101001011_2 + 1 = 101001100_2$$

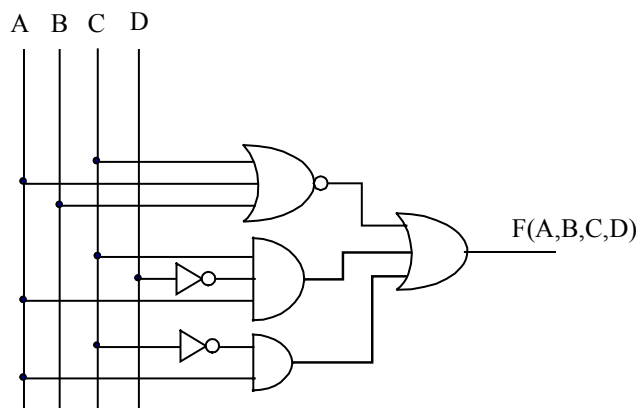
$$-Dv = -170 = - (010101010_2) = 101010101_2 + 1 = 101010110_2$$

Realizando agora a operação de adição em binário:

$$\begin{array}{r} 101001100 \\ + 101010110 \\ \hline 1010100010 \end{array}$$

Podemos concluir que o resultado com 9 bits (010100010) não é correcto porque ocorre overflow porque é excedida a gama de representação de números inteiros com sinal, usando 9 bits em complemento para dois.

2 - Considere o seguinte circuito lógico que realiza a função F(A,B,C,D):



- a) Obtenha uma expressão booleana da função F(A,B,C,D) e construa a sua tabela de verdade.

Traduzindo o circuito dado podemos escrever directamente a expressão booleana:

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

Aplicando as leis de DeMorgan podemos transformar esta expressão numa forma soma-de-produtos, que é mais conveniente para construir a tabela de verdade pedida:

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

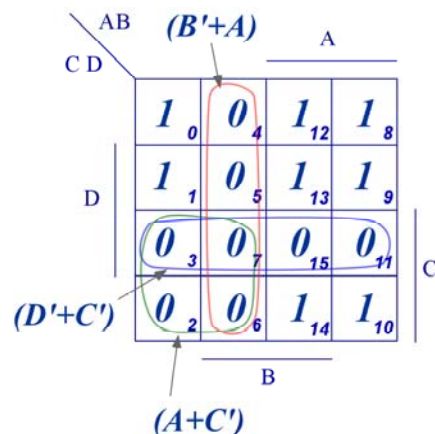
NOME: _____ **CORRECÇÃO** _____ TURMA _____

Com a função representada na forma soma-de-produtos, pode-se construir a tabela de verdade sabendo que a função vale 1 quando $C=B=A=0$ (1º termo de produtos) ou quando $C=A=1$ e $D=0$ (2º termo) ou quando $C=0$ e $A=1$ (último termo):

A	B	C	D	F(A,B,C,D)
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

- b) Utilizando mapas de Karnaugh, obtenha a forma simplificada produto de somas (POS) da função $F(A,B,C,D)$.

Para obter a função mínima produto-de-somas vamos representar a função $F(A,B,C,D)$ no mapa de Karnaugh e agrupar os zeros:

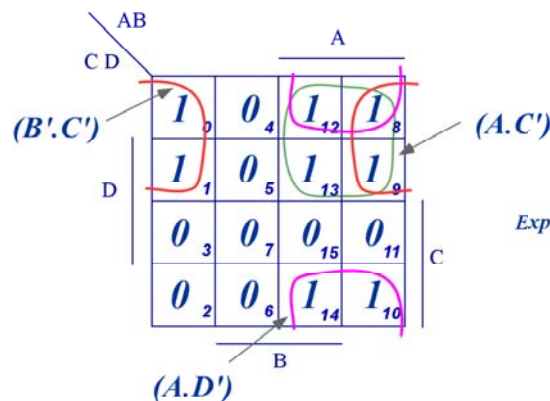


Expressão mínima produto-de-somas:

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

- c) Realize a função $F(A,B,C,D)$ utilizando apenas portas lógicas do tipo NAND de duas entradas (utilize o verso da folha).

Para construir um circuito lógico apenas com portas lógicas do tipo NAND é conveniente partir de um circuito AND-OR e para isso devemos obter a expressão (mínima) do tipo soma-de-produtos:

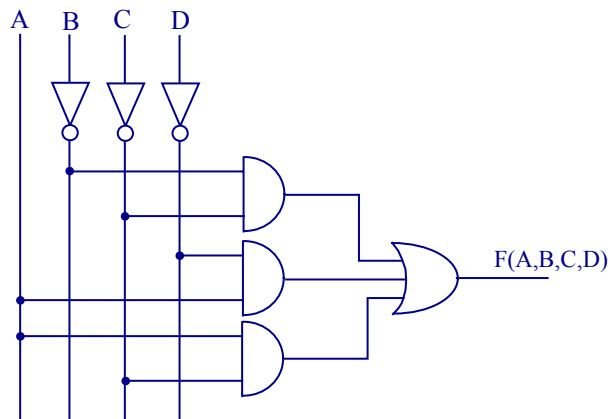


Expressão mínima soma-de-produtos:

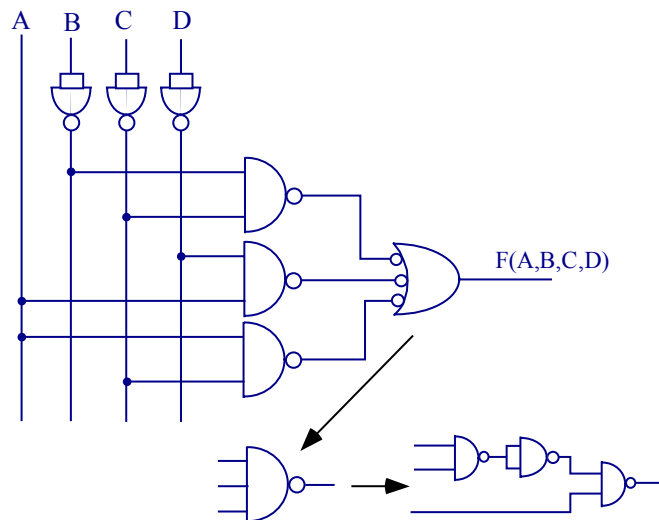
$$Z = B' \cdot C' + A \cdot D' + A \cdot C'$$

NOME: CORRECÇÃO TURMA

O circuito AND-OR que realiza a expressão obtida é:

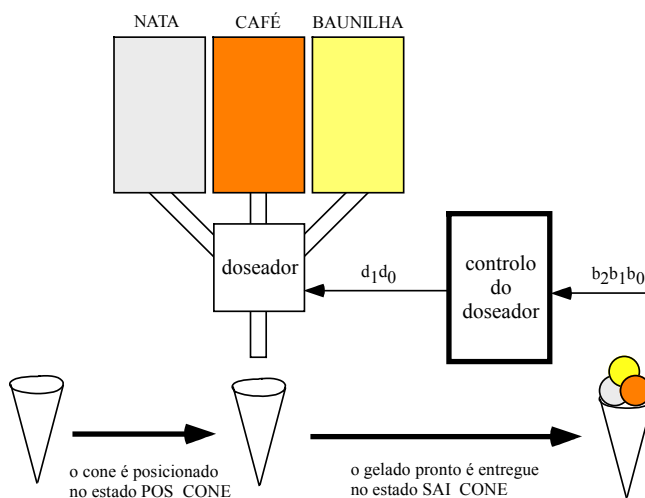


Transformando este circuito de forma a utilizar apenas portas NAND com duas entradas:



NOME: _____ **CORRECÇÃO** _____ TURMA _____

3 — Considere uma máquina para fazer gelados constituídos por 3 doses (uma dose é uma bola de gelado) que podem ser de nata, café ou baunilha. Quando recebe a ordem para fazer um gelado, o sistema de controlo desta máquina coloca um cone sob a unidade de doseamento, inicia o processo de saída dos diversos sabores de gelado, e finalmente coloca o gelado pronto na saída da máquina (ver figura).



A unidade de controlo do doseador que se pretende projectar é uma máquina de estados cuja entrada é um código de 3 bits ($b_2b_1b_0$) que especifica o tipo de gelado pretendido. O período do sinal de relógio que comanda esta máquina de estados (2 segundos) corresponde ao tempo necessário para depositar no cone uma das doses de gelado. De acordo com o código $b_2b_1b_0$ recebido, a máquina deve dosear cada um dos sabores segundo as regras seguintes:

- o custo dos sabores é, por ordem crescente, nata (o mais barato), café e baunilha (o mais caro).
- a ordem do doseamento dos sabores seleccionados deverá ser, quando existam, primeiro nata, depois café e no fim baunilha.
- o bit $b_2=1$ selecciona o sabor nata, $b_1=1$ selecciona o sabor café e $b_0=1$ selecciona o sabor baunilha.
- se o código tiver um único bit igual a 1, o gelado é composto por 3 bolas desse sabor.
- se o código tiver 2 bits iguais a 1, é repetido o sabor mais barato.
- se o código tiver 3 bits iguais a 1 o gelado terá os 3 sabores.
- o código 000 é ilegal (nunca ocorre).

Por exemplo, se o código for 011 deverá ser feito um gelado com duas bolas de café e uma bola de baunilha; se for 100 deverá ter 3 bolas de nata.

As saídas d_1d_0 da máquina de estados comandam o doseador para controlar a saída de cada tipo de gelado, de acordo com a tabela seguinte:

d_1d_0	tipo de gelado
00	nenhum
01	nata
10	café
11	baunilha

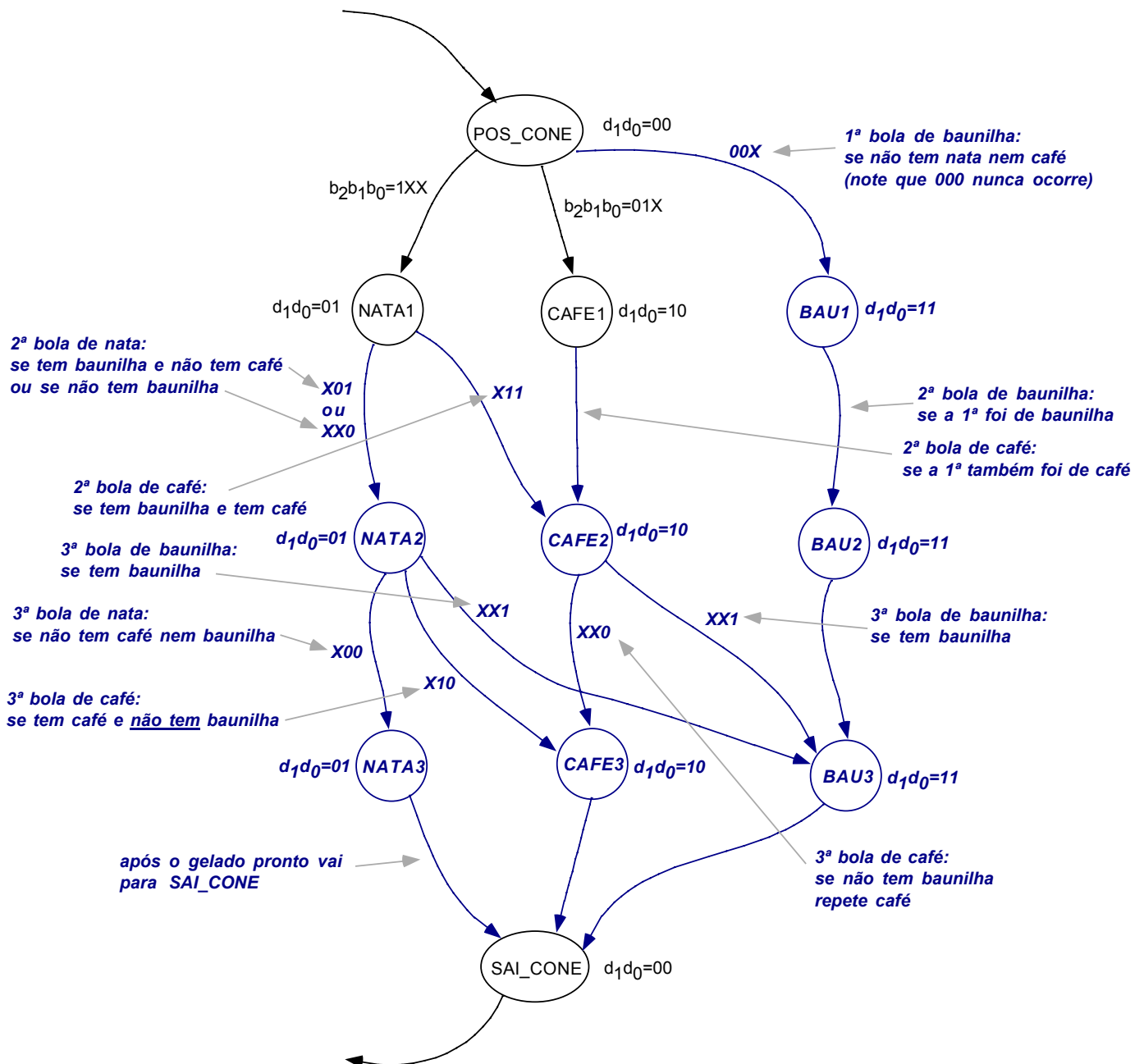
A figura (ver próxima folha) representa um diagrama de estados incompleto do sistema de controlo, relativo à unidade de doseamento de gelado. No estado POS_CONE é posicionado um cone sob a saída da unidade de doseamento e no estado SAI_CONE o gelado pronto é colocado na saída da máquina. Complete a parte do diagrama entre estes dois estados (POS_CONE e SAI_CONE), que gera a sequência de sinais de controlo apropriada (d_1d_0) para fabricar o gelado seleccionado pelo código $b_2b_1b_0$.

NOME: _____

CORRECÇÃO _____

TURMA _____

Como a duração de cada estado corresponde à colocação de uma dose de gelado de cada sabor, o diagrama de estados terá um total de 9 estados (para além de POS_CONE e SAI_CONE), um para uma dose de cada sabor. Os valores indicados junto de cada transição representam as condições de transição de estado para $b_2b_1b_0$; as transições de estado que não apresentam condição associada são transições incondicionais.



NOME: _____ CORRECÇÃO _____ TURMA _____

4 — Considere a máquina sequencial cuja tabela de transições de estados se mostra. Os estados S0, S1 e S2 são codificados respectivamente na forma $Q_1Q_0 = 00, 01$ e 10 .

S	X	
	0	1
S0	S0,0	S1,1
S1	S2,0	S1,0
S2	S0,0	S1,1
	S*,Z	

- a) Identifique as entrada e saídas da máquina e diga, justificando, se se trata de uma máquina de Moore ou Mealy.

A entrada é X e a saída é Z. A máquina de estados representada corresponde a uma máquina de Mealy porque a saída Z é função do estado actual S e da entrada X.

- b) Obtenha o circuito respectivo utilizando *flip-flops* tipo D e um critério de custo mínimo (utilize o verso da folha).

Primeiro vamos representar a tabela de transição de estados com a codificação dada para os estados, considerando dois flip-flops do tipo D para representar o estado actual. Para minimizar o custo (complexidade) do circuito, vamos considerar que, quando o estado actual é 11 (não pertence à máquina de estados) o próximo estado e a saída Z são indiferentes:

Q_1Q_0	X	Q_1^*	Q_0^*	Z
0 0	0	0	0	0
0 0	1	0	1	1
0 1	0	1	0	0
0 1	1	0	1	0
1 0	0	0	0	0
1 0	1	0	1	1
1 1	0	x	x	x
1 1	1	x	x	x

Vamos agora, com recurso a mapas de Karnaugh, obter expressões minimizadas para $Z(Q_1, Q_0, X)$, $Q_1^(Q_1, Q_0, X)$ e $Q_0^*(Q_1, Q_0, X)$:*

$Z(Q_1, Q_0, X)$

Q_1Q_0		Q_1			
X		00	01	11	10
	0	0	0	x	0
	1	1	0	x	1
		Q_0		$Q_0'X$	

$Q_1^*(Q_1, Q_0, X)$

Q_1Q_0		Q_1			
X		00	01	11	10
	0	0	1	x	0
	1	0	0	x	0
		Q_0		$Q_0'X'$	

$Q_0^*(Q_1, Q_0, X)$

Q_1Q_0		Q_1			
X		00	01	11	10
	0	0	0	x	0
	1	1	1	x	1
		Q_0		X	

NOME: _____ CORRECÇÃO _____ TURMA _____

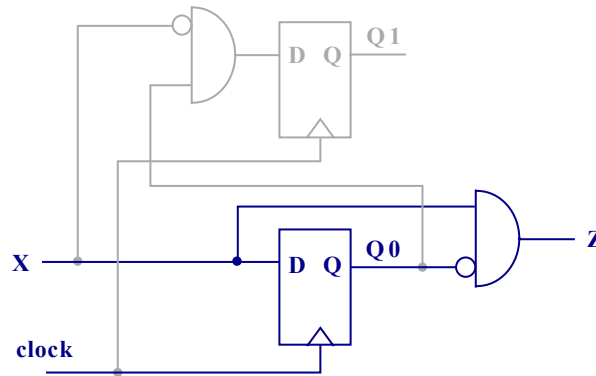
As equações minimizadas na forma soma-de-produtos são:

$$Q1^*(Q1, Q0, Z) = Q0.X'$$

$$Q0^*(Q1, Q0, X) = X$$

$$Z(Q1, Q0, X) = Q0'.X$$

Note-se que, como nestas equações nunca aparece $Q1$, então o circuito pode ser simplificado removendo o flip-flop $Q1$ e todo o circuito lógico que produz $Q1^$ (desenhado a cinzento na figura):*



- c) Supondo que a máquina arranca no estado não definido (em que $Q_1Q_0 = 11$), e tendo por base a solução encontrada, indique, justificando, quais os estados seguintes se a entrada X for igual a 0 ou igual a 1.

Substituindo os valores para $Q1$ e $Q0$ nas equações obtidas na alínea anterior obtemos:

Se $X=0$ então $Q1^ = 1.1 = 1$ e $Q0^*=0$, logo $S^* = 10$ (estado $S2$)*

Se $X=1$ então $Q1^ = 1.0 = 0$ e $Q0^*=1$, logo $S^* = 01$ (estado $S1$)*

Outra forma de obter este resultado: como a máquina de estados não depende de $Q1$, então o "estado" $Q1Q0=11$ é equivalente ao estado $Q1Q0=01$ (estado $S1$); pela tabela dada podemos ver que de $S1$ passa para $S2$ se $X=0$ e passa para $S1$ se $X=1$.

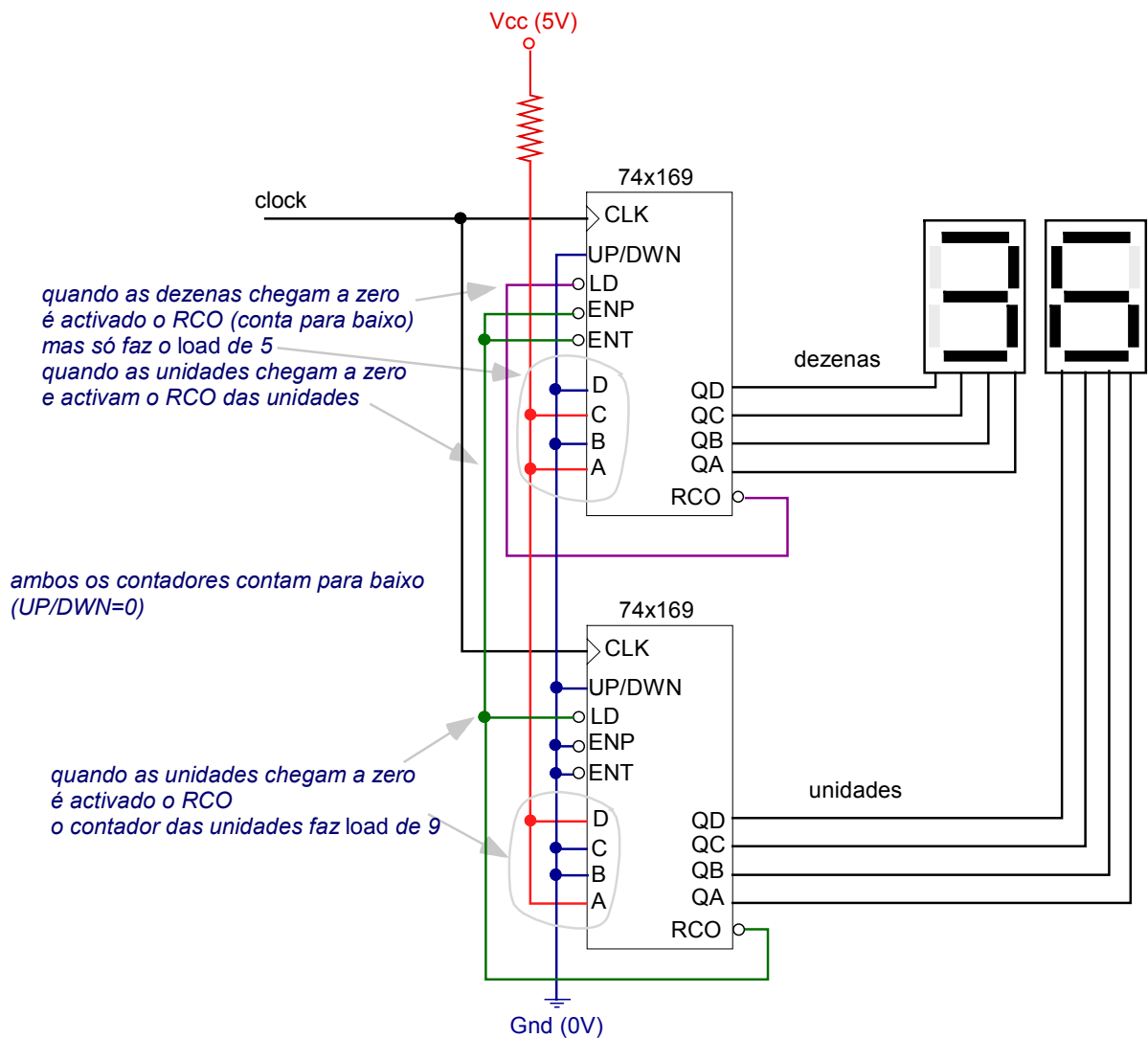
NOME: CORRECÇÃO TURMA

5 — Pretende-se projectar um circuito sequencial que conte ciclicamente de 59 até zero, em formato BCD (dois dígitos decimais codificados em binário), afixando o resultado em dois mostradores de 7 segmentos (por exemplo, como se mostra na figura, quando o contador das dezenas tem as saídas $Q_D Q_C Q_B Q_A = 0011$ e o contador das unidades tem as saídas $Q_D Q_C Q_B Q_A = 0110$, o valor afixado nos mostradores é 36).

a) Projecte o circuito referido utilizando apenas 2 contadores UP/DOWN do tipo 74x169.

74x169				estado presente					próximo estado			
UP/DWN	LD	ENT	ENP	QD	QC	QB	QA	RCO	QD*	QC*	QB*	QA*
x	0	x	x	x	x	x	x	a)	D	C	B	A
x	1	1	x	x	x	x	x	1	QD	QC	QB	QA
x	1	x	1	x	x	x	x	a)	QD	QC	QB	QA
1	1	0	0	N (se $N < 15$)				1	N + 1			
1	1	0	0	1	1	1	1	0	0	0	0	0
0	1	0	0	N (se $N > 0$)				1	N - 1			
0	1	0	0	0	0	0	0	0	1	1	1	1

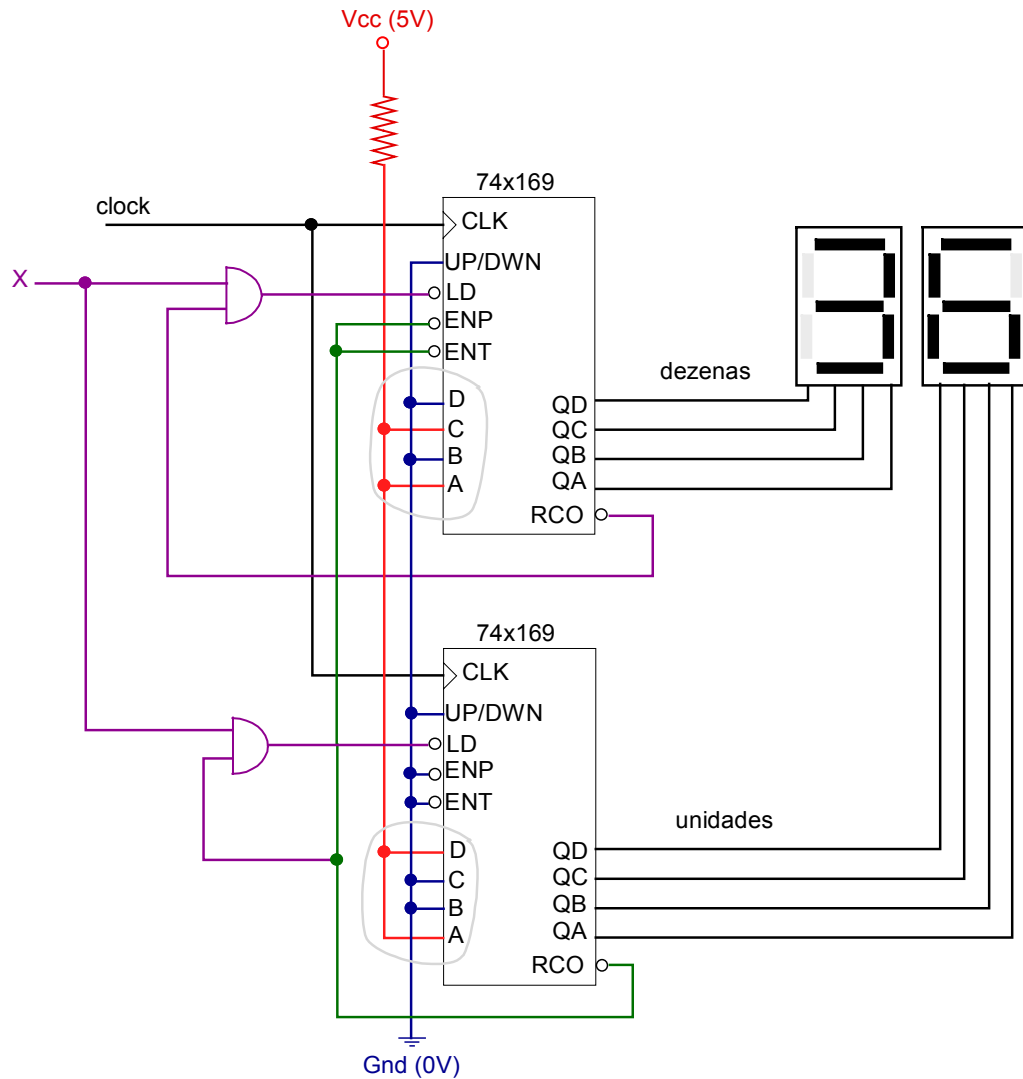
a) $\overline{RCO} = 0$ se $\overline{ENT} = 0$ e ($(N=15 \text{ e } \overline{UP/DWN}=1)$ ou $(N=0 \text{ e } \overline{UP/DWN}=0)$)



NOME: _____ **CORRECÇÃO** _____ TURMA _____

- b) Modifique o circuito anterior por forma a acrescentar uma entrada (activa no nível lógico zero) que permita iniciar o contador no valor 59 (identifique apenas as alterações a efectuar no circuito anterior e se necessário utilize o verso da folha).

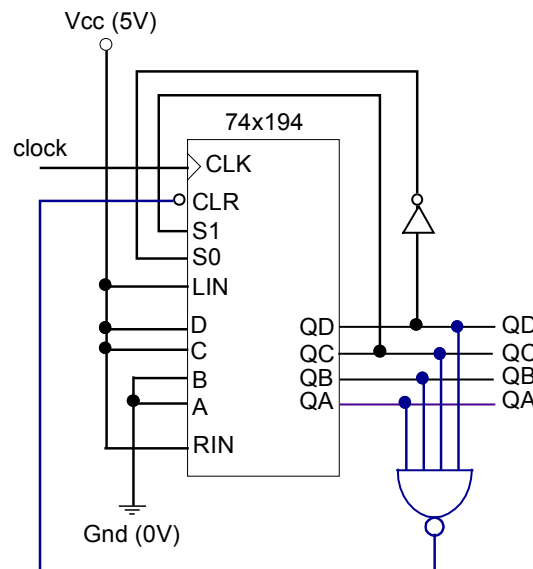
Para iniciar os contadores em 59 é necessário activar as entradas LD (Load) de ambos os contadores. Para isso basta acrescentar uma porta AND nas entradas LD de cada contador, com uma das suas entradas ligada ao sinal X que provoca a iniciação em 59:



NOME: CORRECÇÃO TURMA **6** – Considere o circuito seguinte baseado no registo de deslocamento 74x194:

Universal Shift-register 74x194

função	S1	S0	QA*	QB*	QC*	QD*
hold	0	0	QA	QB	QC	QD
shift right	0	1	RIN	QA	QB	QC
shift left	1	0	QB	QC	QD	LIN
load	1	1	A	B	C	D



- a) Determine a sequência nas saídas $Q_A Q_B Q_C Q_D$, supondo que o estado inicial é $Q_A Q_B Q_C Q_D = 0000$.

Partindo do estado inicial $Q_A Q_B Q_C Q_D = 0000$, vamos analisar o modo de funcionamento do shift-register para determinar o próximo estado (note que RIN e LIN estão permanentemente ligados a 1, e as entradas ABCD estão também sempre ligadas a 0011):

Q_A	Q_B	Q_C	Q_D	$S1$	$S0$	Modo
0	0	0	0	0	1	shift right (de QA para QD, RIN=1)
1	0	0	0	0	1	shift right (de QA para QD, RIN=1)
1	1	0	0	0	1	shift right (de QA para QD, RIN=1)
1	1	1	0	1	1	load de ABCD=0011
0	0	1	1	1	0	shift left (de QD para QA, LIN=1)
0	1	1	1	1	0	shift left (de QD para QA, LIN=1)
1	1	1	1	1	0	shift left (de QD para QA, LIN=1)
1	1	1	1	1	0	shift left (de QD para QA, LIN=1)
...

Após ser atingido o estado $Q_A Q_B Q_C Q_D = 1111$, o shift-register mantém-se no estado $Q_A Q_B Q_C Q_D = 1111$.

- b) Indique o estado seguinte a $Q_A Q_B Q_C Q_D = 1111$ e altere o circuito por forma a que esse estado seguinte passe a ser $Q_A Q_B Q_C Q_D = 0000$ (mostre as alterações na figura).

Pelos resultados obtidos na alínea anterior, o estado a seguir a $Q_A Q_B Q_C Q_D = 1111$ é $Q_A Q_B Q_C Q_D = 1111$. Para que o estado seguinte a $Q_A Q_B Q_C Q_D = 1111$ seja $Q_A Q_B Q_C Q_D = 0000$, basta ligar à entrada CLR (clear) do shift-register a saída (activa com o nível lógico zero) de um comparador com 1111, que pode ser feito com uma porta NAND de 4 entradas. A alteração está indicada a azul no desenho.

- FIM -