Nome: ____________________ **Número:** __________

Atenção: Este teste tem 8 questões em 8 páginas, num total de 200 pontos.
Responda diretamente no enunciado. Fundamente todas as respostas.

1. A representação binária de X e Y com seis bits é a seguinte:

$$X: 010110_2 \qquad Y: 101011_2$$

[10] (a) Assumindo que X e Y estão representados em sinal e grandeza, calcular $X+Y$. Comentar o resultado.

> X e Y têm sinais diferentes e $|X| > |Y|$. Logo, $|X+Y| = |X| - |Y|$. O sinal do resultado é positivo.
>
> $|X| - |Y| = 10110 - 01011 = 01011$
>
> Logo: $X+Y = 001011$. O resultado final é representável em 6 bits.

[10] (b) Assumindo que X e Y estão representados em complemento para 2, calcular $Y-X$ e comentar o resultado.

> $Y - X = Y + (-X) = 101011 + 101010 = 010101$
>
> O resultado está incorreto. Os operandos da adição são negativos e o resultado é positivo. Portanto, o resultado correto da operação não está na gama de representação (i.e., ocorreu *overflow*).

[10] 2. O sistema de controlo de um frigorífico regista a temperatura em décimos de grau centígrado usando números inteiros. A gama de temperaturas prevista vai de $-20\,°C$ a $10\,°C$. Indique uma representação binária apropriada com o menor número possível de bits. Cada valor da temperatura deve ter uma representação única.

> Como há valores positivos e negativos, deve usar-se uma representação com sinal. A representação em sinal e grandeza tem dois códigos para o valor 0. Portanto, usar a representação em complemento para dois.
>
> Em décimos de grau, a gama a representar é [-200; 100]. São necessários 9 bits (gama [-256; +255]), porque a gama da representação em complemento para dois com N bits é [-2^{N-1}; $2^{N-1}-1$]

3. A representação em formato IEEE 754 (precisão simples) de B é $40A00000_{16}$. Seja $A = -2{,}5_{10}$.

[10] (a) Converter A para o formato IEEE 754 e apresentar o resultado da conversão em hexadecimal.

A é negativo: sinal é 1.

$2{,}5_{10} = 10{,}1_2 = 1{,}01 \times 2^1$

Expoente codificado é $127+1 = 128 = 10000000_2$

Codificação de A: $1 \mid 10000000 \mid 01000\ldots0 = C0200000_{16}$

[15] (b) Apresentar todos os passos do cálculo de $A \times B$ (em binário).

Codificação de B: $0 \mid 10000001 \mid 01000\ldots0$

1. Sinal: negativo (operandos com sinais diferentes)
2. Expoente: $10000000 + 10000001 = 100000001$
   Subtraindo 127_{10}: $100000001 - 01111111 = 10000010$
3. Produto das mantissas: $1{,}01 \times 1{,}01 = 1{,}1001$
4. Normalização: desnecessária.

O resultado é: $1 \mid 10000010 \mid 100100\ldots0 = C1480000_{16}$.

4. A função booleana $F(X_2, X_1, X_0)$ tem o valor 1 se e só se o número de três bits $X_2X_1X_0$ for múltiplo inteiro (não-nulo) de 2 ou 3.

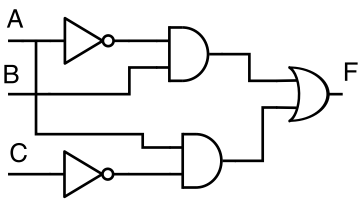
[15] (a) Preencher a tabela de verdade de F apresentada a seguir.

X_2	X_1	X_0	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

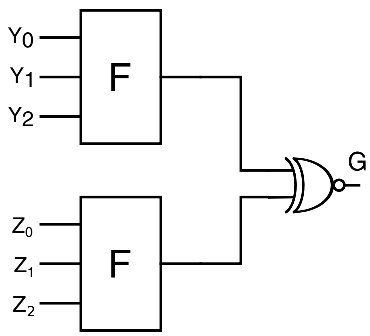
[15] (b) Determinar a representação de F como soma de produtos simplificada.

$$F = \overline{A}{\cdot}B{\cdot}\overline{C} + \overline{A}{\cdot}B{\cdot}C + A{\cdot}\overline{B}{\cdot}\overline{C} + A{\cdot}B{\cdot}\overline{C} = \overline{A}{\cdot}B + A{\cdot}\overline{C}$$

[15] (c) Apresentar o circuito lógico que implementa a função F.

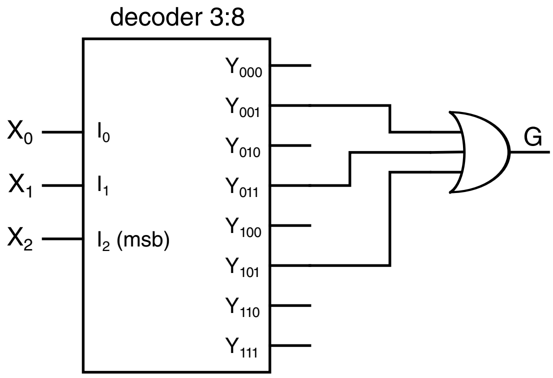


[10] (d) No circuito da figura, o bloco `F` implementa a função F. Explicar em que situações é que a saída $G(Y_2, Y_1, Y_0, Z_2, Z_1, Z_0)$ toma o valor 1.



A porta XNOR é 1 sempre que as suas entradas são iguais.
Portanto, $G = 1$ sempre que os valores A_2, A_1, A_0 e B_2, B_1, B_0 são ambos múltiplos de 2 ou 3, ou então quando nenhum dos valores é múltiplo de 2 ou 3. No primeiro caso, não é necessário que sejam o mesmo múltiplo. Por exemplo, o número A_2, A_1, A_0 pode ser múltiplo de 3 e B_2, B_1, B_0 múltiplo de 2.

5. A figura mostra um descodificador binário 3:8 com algumas saídas ligadas a uma porta OR.



[15] (a) Determinar a tabela de verdade da saída G em função de X_2, X_1 e X_0. Deduzir a partir da tabela a expressão da função $G(X_2, X_1, X_0)$ na forma de soma de produtos não simplificada.

Como

$$G = Y_{001} + Y_{011} + Y_{101},$$

então $G = 1$ sempre que uma das saídas Y_{001}, Y_{011} ou Y_{101} for 1. As condições para isso são:

- saída Y_{001}: $X_2 = 0, X_1 = 0, X_0 = 1$
- saída Y_{011}: $X_2 = 0, X_1 = 1, X_0 = 1$
- saída Y_{101}: $X_2 = 1, X_1 = 0, X_0 = 1$

Para estes combinações de valores de entrada, a função G é 1; para as restantes é zero.

X_2	X_1	X_0	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

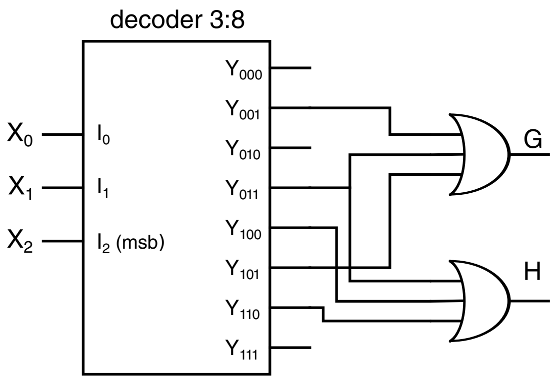
$$G(X_2, X_1, X_0) = \overline{X_2}\cdot\overline{X_1}\cdot X_0 + \overline{X_2}\cdot X_1\cdot X_0 + X_2\cdot\overline{X_1}\cdot X_0$$

[10] (b) Mostrar como alterar o circuito para implementar adicionalmente a função

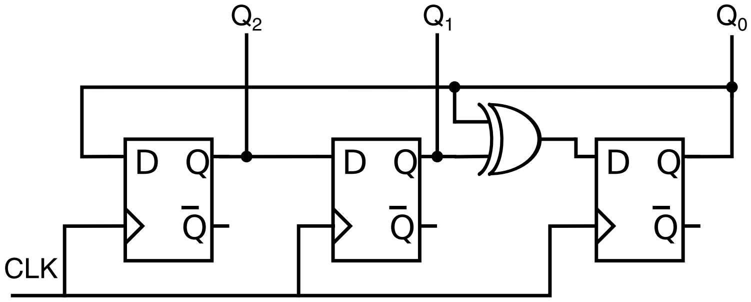
$$H(X_2, X_1, X_0) = \overline{X_2}\cdot X_1\cdot X_0 + X_2\cdot\overline{X_0}.$$

$H(X_2, X_1, X_0) = \overline{X_2}\cdot X_1\cdot X_0 + X_2\cdot\overline{X_0} = \overline{X_2}\cdot X_1\cdot X_0 + X_2\cdot\overline{X_1}\cdot\overline{X_0} + X_2\cdot X_1\cdot\overline{X_0}$

Usando a mesma abordagem que na alínea anterior, $H = Y_{011} + Y_{100} + Y_{110}$. O circuito modificado é o seguinte:



6. Considere o circuito sequencial indicado na figura. Este circuito tem três saída (Q_2, Q_1 e Q_0) e uma entrada para o sinal de relógio.



[15] (a) Para o estado inicial $Q_2 = Q_1 = 0$ e $Q_0 = 1$, determinar os valores de Q_2, Q_1 e Q_0 nos 8 ciclos seguintes.

As saídas dos *flip-flops* no ciclo seguinte são dadas pelas respetivas entradas no ciclo corrente. Assim temos:

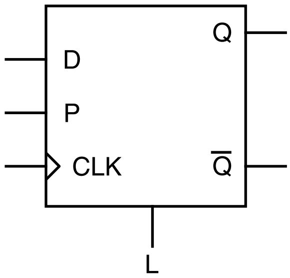
- $D_2 = Q_0$
- $D_1 = Q_2$
- $D_0 = Q_0 \oplus Q_1$

Ciclo	Q_2	Q_1	Q_0	
-	0	0	1	estado inicial
1	1	0	1	
2	1	1	1	
3	1	1	0	
4	0	1	1	
5	1	0	0	
6	0	1	0	
7	0	0	1	volta ao estado inicial
8	1	0	1	

[10] (b) Explicar o comportamento do circuito, quando o estado inicial é $Q_2 = Q_1 = Q_0 = 0$.

Quando o estado é $Q_2 = Q_1 = Q_0 = 0$, o valor das saídas (e o valor do estado seguinte do circuito) permanece sempre a 0. Para os dois *flip-flops* mais à esquerda, entrada é sempre igual à saída de outro *flip-flop*. A entrada do terceiro *flip-flop* vem de uma porta lógica do tipo ou-exclusivo, para a qual temos sempre $0 \oplus 0 = 0$. As saídas são determinadas de novo a cada ciclo de relógio, mas os seus valores são sempre 0.

[10] (c) O circuito usado não permite definir o valor inicial. Para eliminar essa deficiência, vai-se substituir o *flip-flop* tipo D pelo módulo apresentado na figura seguinte. Este módulo tem mais duas entradas designadas por L e P.

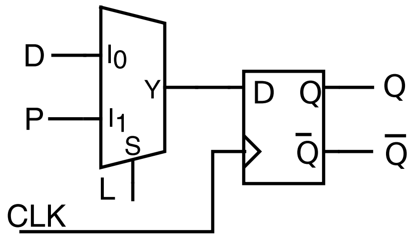


O funcionamento do módulo é o seguinte:

- Se $L = 0$, o circuito funciona como um *flip-flop* tipo D (o valor da entrada P é ignorado);
- Se $L = 1$, após o flanco ativo do relógio (transição 0→1) tem-se $Q = P$ (o valor da entrada D é ignorado).

Apresentar uma implementação do módulo pretendido usando um *flip-flop* tipo D e outro(s) componente(s).

A implementação pedida pode ser feita com um *multiplexer*, conforme indicado na figura. Quando $L = 0$, o módulo funciona como um *flip-flop* do tipo D. Quando $L = 1$, o valor da entrada P é ligado à entrada D do *flip-flop*, pelo que será "capturado" no próximo flanco ascendente do relógio.



7. Um processador suporta instruções de três classes (A, B e C). O valor de CPI para cada classe de instruções está indicado na tabela. Um dado programa muito usado foi compilado com o compilador C_1, resultando nas taxas de ocorrência de instruções de cada classe também indicadas na tabela.

   Pretende-se avaliar o efeito de usar um novo compilador C_2. Esse compilador consegue fazer melhor utilização das instruções de classe A, que passam a ser usadas mais frequentemente. (As taxas de ocorrência correspondentes estão também indicadas na tabela.)

Compilador	Taxa de ocorrência		
	A	B	C
C_1	30 %	35 %	35 %
C_2	50 %	25 %	25 %
CPI	1	2	4

[10] (a) Determinar o CPI médio obtido com cada compilador.

Para o compilador C_1:

$$\mathrm{CPI}_1 = 0{,}3 \times 1 + 0{,}35 \times 2 + 0{,}35 \times 4 = 0{,}3 + 0{,}7 + 1{,}4 = 2{,}4$$

Para o compilador C_2:

$$\mathrm{CPI}_2 = 0{,}5 \times 1 + 0{,}25 \times 2 + 0{,}25 \times 4 = 0{,}5 + 0{,}5 + 1 = 2$$

[10] (b) Estima-se que os programas gerados por C_2 executam mais 10 % de instruções que os programas gerados por C_1. Determinar se será vantajoso utilizar o programa gerado pelo compilador C_2.

Como é usado sempre o mesmo computador, o valor de F mantém-se. Comparando os tempos de execução em cada caso:

$$\frac{T_{\mathrm{exec1}}}{T_{\mathrm{exec2}}} = \frac{N \times 2{,}4}{F} \times \frac{F}{1{,}1 \times N \times 2} = \frac{2{,}4}{2{,}2} > 1$$

Como $T_{\mathrm{exec1}} > T_{\mathrm{exec2}}$, é, de facto, mais vantajoso usar o programa gerado por C_2.

[10] 8. Um computador executa repetidamente três tarefas em sequência:

1. obter informação de um sistema remoto;
2. comprimir essa informação;
3. guardar a informação comprimida num sistema de armazenamento.

O computador gasta 30 % do tempo na tarefa 1 e 50 % na tarefa 2.

Existe a possibilidade de introduzir um algoritmo de compressão mais sofisticado, que reduz a quantidade de informação a armazenar. Estima-se, por isso, que a tarefa 3 possa ser executada 1,6 vezes mais depressa. Contudo, a compressão é mais complexa, o que faz o tempo da tarefa 2 aumentar 1,25 vezes. [Nota: 1,25 = 5/4.]

Determinar se, do ponto de vista do desempenho do sistema, é vantajoso adotar o novo algoritmo.

Seja T o tempo original necessário para realizar as três tarefas e T' o tempo usando o novo algoritmo de compressão.
Então:

$$T' = 0{,}3 \times T + \frac{5}{4} \times 0{,}5 \times T + \frac{0{,}2 \times T}{1{,}6} = T \times \left(0{,}3 + \frac{5}{8} + \frac{0{,}2}{1{,}6}\right) = T \times \left(0{,}3 + \frac{10}{16} + \frac{2}{16}\right)$$

Simplificando:

$$T' = T \times \left(0{,}3 + \frac{12}{16}\right) = T \times \left(0{,}3 + \frac{3}{4}\right) = T \times (0{,}3 + 0{,}75) = T \times 1{,}05$$

Como $T' > T$, o desempenho não seria aumentado pela adoção do novo algoritmo.

Fim

Questão	1	2	3	4	5	6	7	8	Total
Pontos	20	10	25	55	25	35	20	10	200
Nota									