Relatório de Sistemas Digitais L1
Funções Combinatórias

João Oliveira
Tomás A. Reis

Instituto Superior Técnico
Universidade de Lisboa

21 de Março de 2014
Quinta-Feira LSD1

1 Introdução

O objectivo deste trabalho é concepção de um circuito que recebendo um número A no intervalo [0;3] realiza sobre ele uma operação indicada por um número B também no intervalo [0;3] segundo o seguinte código:

B	Operação
0	Deslocamento à direita
1	Deslocamento à esquerda
2	Operação indefinida
3	Identidade

Tabela 1: Operações codificadas

Retorna assim um número S também no intervalo [0;3] e um *bit* C_0 (*"Carry Out"*) que representa o digito que é perdido no deslocamento de A.

Na operação "Deslocamento à direita" cada *bit* é movido para a posição à sua direita. O *bit* mais à direita é assim "perdido" sendo o retorno de C_0, já o *bit* mais à esquerda passa a 0. Neste caso, isto resum-se a que S_0 é igual a A_1, C_0 a A_0 e S_1 é 0.

A operação "Deslocamento à esquerda" é semelhante, sendo que em cada *bit* é movido para a posição à sua esquerda. O *bit* mais à esquerda é assim "perdido" sendo o retorno de C_0, já o *bit* mais à direita passa a 0. Neste caso, isto resum-se a que S_1 é igual a A_0, C_0 a A_1 e S_0 é 0.

A "Operação indefinida" não tem interesse, e como tal o seu retorno é escolhido consoante o mais útil à economização de portas lógicas. A operação "Identidade" retorna o mesmo número, sendo que, neste caso, o "*Carry Out*" não faz sentido, visto não existir deslocamento. Para esta operação, então, S_1 será idêntico a S_1 e S_0 idêntico a A_0.

Tem-se também como alvo conseguir este circuito utilizando o mínimo de recursos.

2 Projecto

2.1 Entradas e Saídas

Estando A, B e S no intervalo [0;3] cada um será representado por dois bits, enquanto C_0 apenas necessitará de um. A codificação de A,B e S seguirão a conversão habitual de binário para decimal, como apresentado na seguintes tabela:

A_0	A_1	Valor de A	B_0	B_1	Valor de B	S_0	S_1	Valor de s
0	0	0	0	0	0	0	0	0
0	1	1	0	1	1	0	1	1
1	0	2	1	0	2	1	0	2
1	1	3	1	1	3	1	1	3

Tabela 2: Codificação das entradas

2.2 Tabela de verdade

A_1	A_0	B_1	B_0	S_1	S_0	C_0
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	1	0	X	X	X
0	0	1	1	0	0	X
0	1	0	0	0	0	1
0	1	0	1	1	0	0
0	1	1	0	X	X	X
0	1	1	1	0	1	X
1	0	0	0	0	1	0
1	0	0	1	0	0	1
1	0	1	0	X	X	X
1	0	1	1	1	0	X
1	1	0	0	0	1	1
1	1	0	1	1	0	1
1	1	1	0	X	X	X
1	1	1	1	1	1	X

Tabela 3: Tabela de Verdade

2.3 Simplificação das funções algébricas

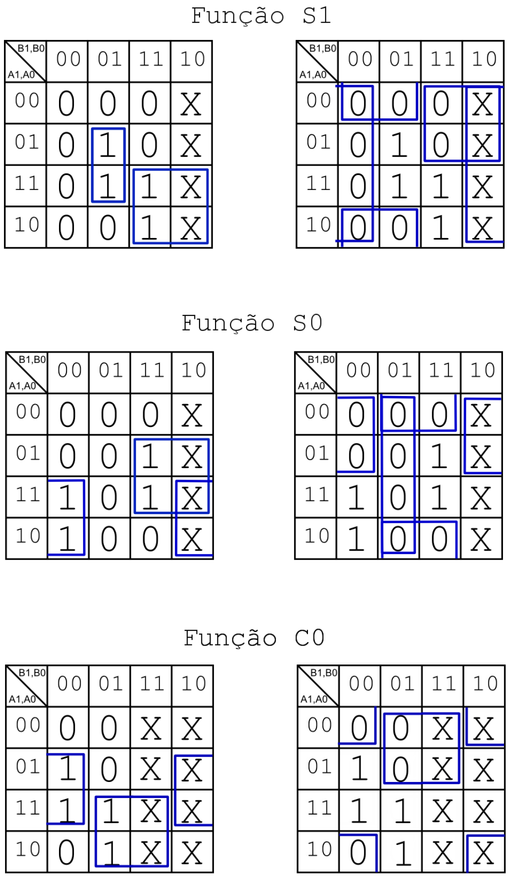
Segundo os quadros de Karnaugh apresentados (Figura 1.) com os implicantes assinalados, podemos exprimir as funções na forma de soma de produtos como:

$$S_1(A_1, A_0, B_1, B_0) = A_0\overline{B_1}B_0 + A_1B_1 \tag{1}$$

$$S_0(A_1, A_0, B_1, B_0) = A_1\overline{B_0}B_1 + A_0B_1 \tag{2}$$

$$C_0(A_1, A_0, B_1, B_0) = A_0\overline{B_0} + A_1B_0 \tag{3}$$

Figura 1: Quadros de Karnaugh



Alternativamente, podemos expressá-las como produto de somas tendo em conta os quadros com implicados marcados, obtendo:

$$S_1(A_1, A_0, B_1, B_0) = (B_0)(A_1 + \overline{B_1})(A_0 + B_1) \tag{4}$$

$$S_0(A_1, A_0, B_1, B_0) = (B_1 + \overline{B_0})(A_1 + \overline{B_0})(A_1 + B_0) \tag{5}$$

$$C_0(A_1, A_0, B_1, B_0) = (A_0 + B_0)(A_1 + \overline{B_0}) \tag{6}$$

2.4 Funções a construir

Sendo os nossos números 79144 e 78811 respetivamente então:

$$ \tag{7} $$

$$157955/3 = 52651 + 2/3 \tag{8}$$

Sendo o último algarismo de um número inteiro em base três dado pelo resto da primeira divisão desse número por três, podemos ver na operação aritmética apresentada em (8) que o algarismo menos significativo da soma dos nossos números é 2. Logo, apenas realizaremos a função S_1 e C_0.

2.5 Transformação das expressões algébricas

2.5.1 De forma a serem concretizadas com portas NAND-2, NAND-3 e NOT

1. A partir da forma disjuntiva

$$\begin{aligned} S_1(A_1, A_0, B_1, B_0) &= A_0\overline{B_1}B_0 + A_1B_1 \\ &= \overline{\overline{((A_0\overline{B_1}B_0)}\ \overline{(A_1B_1))}} \end{aligned} \tag{9}$$

$$\begin{aligned} C_0(A_1, A_0, B_1, B_0) &= A_0\overline{B_0} + A_1B_0 \\ &= \overline{\overline{((A_0\overline{B_0})}\ \overline{(A_1B_0))}} \end{aligned} \tag{10}$$

   Requisitos de implementação:

   (a) 1x NAND-3
   (b) 5x NAND-2
   (c) 2x NOT

2. A partir da forma conjutiva

$$\begin{aligned} S_1(A_1, A_0, B_1, B_0) &= B_0\ (A_0 + B_1)\ (\overline{B_1} + A_1) \\ &= \overline{\overline{B_0\ \overline{\overline{(A_0}\ \overline{B_1)}}\ \overline{(B_1\ \overline{A_1})}}} \end{aligned} \tag{11}$$

$$\begin{aligned} C_0(A_1, A_0, B_1, B_0) &= (A_0 + B_0)\ (A_1 + \overline{B_0}) \\ &= \overline{\overline{\overline{\overline{(A_0}\ \overline{B_0)}}\ \overline{\overline{(A_1}\ B_0)}}} \end{aligned} \tag{12}$$

Requisitos de implementação:

(a) 1x NAND-3

(b) 5x NAND-2

(c) 6x NOT

2.5.2 De forma a serem concretizadas com portas NOR-2, NOR-3 e NOT

1. A partir da forma disjuntiva

$$\begin{aligned} S_1(A_1, A_0, B_1, B_0) &= A_0\overline{B_1}B_0 + A_1B_1 \\ &= \overline{\overline{((A_0 \cdot \overline{B_1} \cdot B_0) + (A_1 \cdot B_1))}} \\ &= \overline{\overline{(\overline{(\overline{A_0} + B_1 + \overline{B_0})} + \overline{(\overline{A_1} + \overline{B_1})})}} \end{aligned} \tag{13}$$

$$\begin{aligned} C_0(A_1, A_0, B_1, B_0) &= A_0\overline{B_0} + A_1B_0 \\ &= \overline{\overline{(A_0 \cdot \overline{B_0}) + (A_1 \cdot B_0)}} \\ &= \overline{\overline{\overline{(\overline{\overline{A_0} + B_0}) + (\overline{\overline{A_1} + \overline{B_0}})}}} \end{aligned} \tag{14}$$

2. A partir da forma conjutiva

$$\begin{aligned} S_1(A_1, A_0, B_1, B_0) &= (B_0)(A_1 + \overline{B_1})(A_0 + B_1) \\ &= \overline{(\overline{B_0} + \overline{(A_1 + \overline{B_1})} + \overline{(A_0 + B_1)})} \end{aligned} \tag{15}$$

$$\begin{aligned} C_0(A_1, A_0, B_1, B_0) &= (A_0 + B_0)(A_1 + \overline{B_0}) \\ &= \overline{(\overline{(A_0 + B_0)} + \overline{(A_1 + \overline{B_0})})} \end{aligned} \tag{16}$$

Requisitos de implementação:

(a) 1x NOR-3

(b) 5x NOR-2

(c) 2x NOT