

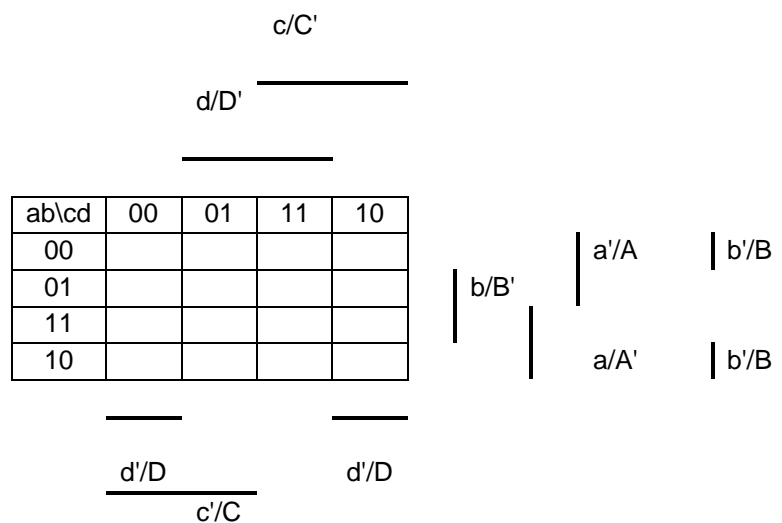
ARQ1 - Recuperação 01

Tema: Sistemas de Numeração e circuitos combinatórios

Exercícios:

01.) Dado o mapa de Veitch-Karnaugh:

n	m	M	f(a,b,c,d)
0	a'b'c'd'	A+B+C+D	0
1	a'b'c'd	A+B+C+D'	0
2	a'b'c d'	A+B+C'+D	1
3	a'b'c d	A+B+C'+D'	1
4	a'b c'd'	A+B'+C+D	0
5	a'b c'd	A+B'+C+D'	1
6	a'b c d'	A+B'+C'+D	0
7	a'b c d	A+B'+C'+D'	1
8	a b'c'd'	A'+B+C+D	0
9	a b'c'd	A'+B+C+D'	1
A	a b'c d'	A'+B+C'+D	1
B	a b'c d	A'+B+C'+D'	1
C	a b c'd'	A'+B'+C+D	0
D	a b c'd	A'+B'+C+D'	1
E	a b c d'	A'+B'+C'+D	0
F	a b c d	A'+B'+C'+D'	0



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
mintermos																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
MAXTERMOS																

Determinar e implementar os circuitos equivalentes em Verilog e no Logisim:

- expressão canônica para SoP(a,b,c,d)
- expressão canônica para PoS(A,B,C,D)
- simplificação de mintermos por mapa de Veitch-Karnaugh
- simplificação de MAXTERMOS por mapa de Veitch-Karnaugh
- expressão SoP equivalente com portas NAND de 2 entradas (usar dupla negação)
- expressão PoS equivalente com portas NOR de 2 entradas (usar dupla negação)

02. Implementar no Verilog e no Logisim as expressões abaixo para obter as tabelas verdade:

a.)

```
module f ( output s, input a, input b, input c );  
  wire w1, w2, w3, w4;  
  not NOT_1 (w1,b);  
  not NOT_1 (w2,c);  
  and AND_1 (w3,a,w2);  
  and AND_2 (w4,a,w1,c);  
  or OR_1 (s,w3,w4);  
endmodule // s = f (a,b,c)
```

b.) a expressão SoP simplificada e implementada apenas com portas NAND

c.) a expressão PoS simplificada

d.)  **$\text{mux}(\text{mux}(a,b',a),\text{mux}(a,c',a),b)$**

03. Implementar no Verilog e no Logisim a expressão  **$S=(A'+C).(A+B).(B+C)$**

04. Expressar a representação segundo a norma IEEE-754 de 18.625 em 32 bits.

05. Calcular:

a.)  $1/4$  de  $\text{FACE}_{16}$  já em complemento de 2

b.) quociente e o resto de  $235_8$  dividido por  $17_{16}$

c.)  $a^b$  e  $(-a)^{-b}$  em um byte para  $a=5$  e  $b=3$

d.)  $(-11)_{10}$  em complemento de 2 com tamanhos iguais a 7, 6 e 5 bits