GLOBAL – 13 FEV 2020

GRUPO I

1) $F(A,B,C,D) = \sum m(5,6,11,14) + \sum md(3,4,8,9,13,15)$

$\Rightarrow \sum M(0,1,2,7,10,12)$

	TERMO	A	B	C	D	F
	0	0	0	0	0	0
	1	0	0	0	1	0
	2	0	0	1	0	0
	3	0	0	1	1	X
	4	0	1	0	0	X
	5	0	1	0	1	1
	6	0	1	1	0	1
	7	0	1	1	1	0
	8	1	0	0	0	X
	9	1	0	0	1	X
A	10	1	0	1	0	0
B	11	1	0	1	1	1
C	12	1	1	0	0	0
D	13	1	1	0	1	X
E	14	1	1	1	0	1
F	15	1	1	1	1	X

F

AB \ CD	00	01	11	10
00	0	0	X	0
01	X	1	0	1
11	0	X	X	1
10	X	X	1	0

X, Y, Z

$F = X + Y + Z$:

$= B\overline{C}D + AD + BC\overline{D}$

2) D28h ⟶ OCTAL

$D28h = \underbrace{1101}_{D}\,\underbrace{0010}_{2}\,\underbrace{1000}_{8}{}_b$

$\underbrace{110}_{6}\,\underbrace{100}_{4}\,\underbrace{101}_{5}\,\underbrace{000}_{0}{}_b = 6450_8$

3) -40_d ⟶ BINÁRIO, 8 BITS, COMP. P/2

40 | 2
00 20 | 2
0 00 10 | 2
0 0 5 | 2
1 2 | 2
0 1 | 2
1 0

$40_d = 101000_b$

$= 00101000_b$

(8 BITS)

COMP. P/1 ⇒ INVERTER BIT-A-BIT $= 11010111_b$

COMP P/2 = COMP. P/1 + 1 = 11010111 + 1

$$\begin{array}{r} 11010111 \\ +\quad\quad\quad 1 \\ \hline 11011000 \end{array}$$

$-40_d = 11011000_b$ EM COMP. P/2, 8 BITS

GRUPO II

1) $F(A,B,C) = \underbrace{(A+\overline{B}C)(\overline{A}+\overline{B}C)}_{\text{ADJACENCIA}} + \overline{\overline{A}+\overline{B}}\,(A+B)$

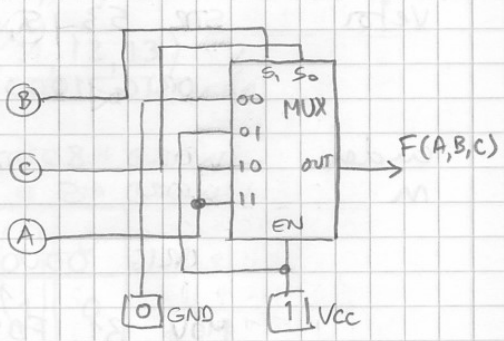
$= \overline{B}C + AB(A+B) =$

$= \overline{B}C + ABA + ABB = \overline{B}C + \underbrace{AB + AB}_{\text{IDEMPOTÊNCIA}}$

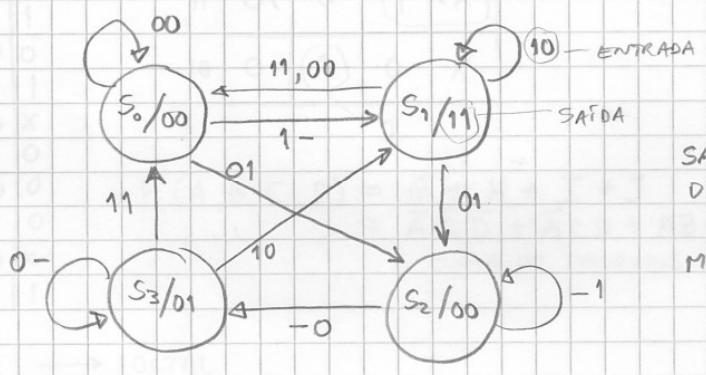
$= \overline{B}C + AB$

4) MUX $S_1 = B$, $S_0 = C$

B	C	$\overline{B}C + AB$
0	0	$1\cdot 0 + A\cdot 0 = 0+0 = 0$
0	1	$1\cdot 1 + A\cdot 0 = 1+0 = 1$
1	0	$0\cdot 0 + A\cdot 1 = 0+A = A$
1	1	$0\cdot 1 + A\cdot 1 = 0+A = A$



GRUPO III



SAÍDAS NÃO DEPENDEM DAS ENTRADAS

⇓

MÁQUINA DE MOORE

1) 4 ESTADOS → CODIFICAÇÃO BINÁRIA ⟹ 2 F.F TIPO D

ESTADO	EST. INICIAL Q_1	Q_0	ENTRADAS E_1	E_0	EST. SEGUINTE D_1	D_0	SAÍDAS O_1	O_0	
S_0	0	0	0	0	0	0	0	0	S_0
	0	0	0	1	1	0			S_2
	0	0	1	0	0	1			S_1
	0	0	1	1	0	1			S_1
S_1	0	1	0	0	0	0	1	1	S_0
	0	1	0	1	1	0			S_2
	0	1	1	0	0	1			S_1
	0	1	1	1	0	0			S_0
S_2	1	0	0	0	1	1	0	0	S_3
	1	0	0	1	1	0			S_2
	1	0	1	0	1	1			S_3
	1	0	1	1	1	0			S_2
S_3	1	1	0	0	1	1	0	1	S_3
	1	1	0	1	1	1			S_3
	1	1	1	0	0	1			S_1
	1	1	1	1	0	0			S_0

→ NÃO DEPENDEM DAS ENTRADAS
MÁQUINA DE MOORE

2) VAR. ESTADO SEGUINTE

D_1

AB \ CD	00	01	11	10
00	0	1	0	0
01	0	1	0	0
11	1	1	0	0
10	1	1	1	1

$D_1 = G + H + I = A\overline{C} + A\overline{B} + \overline{C}D$

D_0

AB \ CD	00	01	11	10
00	0	0	1	1
01	0	0	0	1
11	1	1	0	1
10	1	0	0	1

$D_0 = J + K + L + M = C\overline{D} + \overline{A}\,\overline{B}C + AB\overline{C} + A\overline{D}$

SAÍDAS: $O_0 = Q_0$, $O_1 = \overline{Q}_1 Q_0$

Grupo IV

```
;AC:21010 2020 - PFolio
; --- E --- 1902295

; O programa recebe um valor em R1 com um endereço de memória,
; e em R2 com o número de posições a processar a partir do endereço em R1.
; O programa deve colocar a 0 as posições de memória a processar
; em que o número de bits seja ímpar.

; ----------- valores para teste ----------------
                ORIG 8000h

valores         STR 16384,16383,32768,32767,6,7,8,256,4095,4096

; ----------- variáveis -------------------------
                ORIG 1000h

ender           WORD 8000h
n               WORD 10

; ----------- início do programa ----------------
                ORIG 0000h

                MOV R1, M[ender]    ;endereço
                MOV R2, M[n]        ;nº de elementos
                MOV R3, R0          ;índice
                MOV R7, R0          ;conta_zeros

ciclo:          MOV R4, M[R1]   ;coloca em R4 o valor do vetor a testar
;1º) vai rodando bit-a-bit para esquerda o nº a testar, enquanto o carry for 0,
;    em R7 conta zeros (conta 1 zero a mais, pq só pára depois de aparecer o 1)
primeiro:       INC R7          ;teste do nº da esquerda para a direita
                ROLC R4, 1      ;enquanto não aparecer um 1, conta_zeros
                BR.NC primeiro
;2º) subtrai o nº de zeros contados, a 16, e esse diferencial é igual
;    à quantidade de dígitos do nº
                DEC R7          ;foi detetatado 1, conta_zeros tem 1 a mais
                MOV R6, 16      ;R6 = 16
                SUB R6, R7      ;16-R7= nº de bits do nº testado
                RORC R6, 1      ;LSB no Carry: C=1 nº é ímpar, se C=0 nº é par
                BR.C impar
                BR par

;3º) se for par, não mexe no vetor e prepara registos para testar o
;    próximo nº (verifica se já acabou o vetor)
par:            INC R1        ;vai para o índice de vetor seguinte
                INC R3        ;incrementa índice de ciclo
                MOV R7, R0    ;reset conta_zeros
                CMP R3, R2    ;verifica se já analizou todos os nºs do vetor
                BR.NZ ciclo
                BR fim
```

```
;4º) se for ímpar, coloca esse valor do vetor a 0 e prepara registos para
;    testar o próximo nº (verifica se já acabou o vetor)
impar:          MOV M[R1], R0 ;quantidade ímpar de dígitos, valor colocado a 0
                INC R1        ;vai para o índice de vetor seguinte
                INC R3        ;incrementa índice de ciclo
                MOV R7, R0    ;reset conta_zeros
                CMP R3, R2    ;verifica se já analizou todos os nºs do vetor
                BR.NZ ciclo

fim:            JMP fim
```