

INF01058

**Circuitos Digitais**

Síntese de FSM com FF D

Aula 24b

Circuitos Digitais

**1. Introdução**

- processo de projeto
  - especificação (p.ex. FSM)
  - tabela de estados
  - equações de entrada (para FF's) e de saída
  - síntese a partir das equações (problema de lógica combin.)
- número de flip-flops
  - com codificação:  $n$  FF's  $\rightarrow$   $2^n$  estados
  - sem codificação:  $n$  FF's  $\rightarrow$   $n$  estados
- escolha do tipo dos FF's
  - influencia determinação das equações de entrada

Circuitos Digitais

**2. Projeto com flip-flops tipo D**

**processo de projeto**

- Obter tabela de estados
- Derivar equações de entrada a partir do "próximo estado" na tabela
- Derivar equações de saída a partir da "saída" na tabela
- Simplificar equações de entrada e saída
- Desenhar circuito lógico com FF's D e portas lógicas de acordo com as equações.

**exemplo**

4 estados  $\rightarrow$  2 FF's: A, B

1 entrada X  
1 saída Y

Circuitos Digitais

**Tabela de Estados**

Estado Atual		Entrada	Próx. Estado		Saída	MINTERMO
A	B	X	A	B	Y	
0	0	0	0	0	0	0
0	0	1	0	1	1	1
0	1	0	1	0	0	2
0	1	1	0	1	0	3
1	0	0	1	0	0	4
1	0	1	1	1	1	5
1	1	0	1	1	0	6
1	1	1	0	0	0	7

**equações de entrada para FF's tipo D**

$$A(t+1) = D_A(A,B,X) = \sum m(2,4,5,6)$$

$$B(t+1) = D_B(A,B,X) = \sum m(1,3,5,6)$$

**equação de saída**

$$Y(A,B,X) = \sum m(1,5)$$

Circuitos Digitais

**simplificação das equações**

$D_A = \overline{A}\overline{B} + B\overline{X}$

$D_B = \overline{A}X + \overline{B}X + AB\overline{X}$

$Y = \overline{B}X$

**"desenhar" circuito lógico**

Reconhecer:

- entrada
- saída
- FF's
- realimentações
- lógica combinac.

Circuitos Digitais

**3. Projeto com estados não utilizados**

5 estados  $\rightarrow$  3 FF's

estados não utilizados:

000  
110  
111

saídas do circuito = saída dos FF's

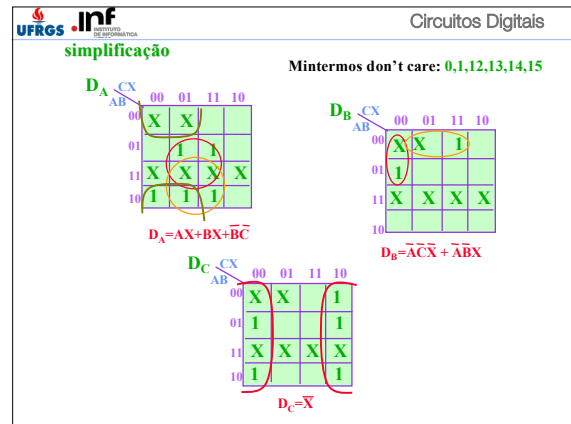
Circuitos Digitais

**Tabela de estados**

Estado Atual				Entrada	Próx. Estado				MINTERMO
A	B	C	X	A	B	C			
0	0	1	0		0	0	1	2	
0	0	1	1		0	1	0	3	
0	1	0	0		0	1	1	4	
0	1	0	1		1	0	0	5	
0	1	1	0		0	0	1	6	
0	1	1	1		1	0	0	7	
1	0	0	0		1	0	1	8	
1	0	0	1		1	0	0	9	
1	0	1	0		0	0	1	10	
1	0	1	1		1	0	0	11	

**equações de entrada**

$A(t+1) = D_A = \sum m(5,7,8,9,11)$   
 $B(t+1) = D_B = \sum m(3,4)$   
 $C(t+1) = D_C = \sum m(2,4,6,8,10)$



- Circuitos Digitais
- 4. Escolha dos flip-flops**
- T** As variáveis de estado vão de 0 → 1 e de volta 1 → 0 (ex. contadores)
  - D** Quando a informação de entrada deve ser armazenada por um tempo
  - SR** Quando sinais diferentes podem dar SET ou RESET nos flip-flops
  - JK** Quando queremos combinar as vantagens de um FF T com SR
  - SR e JK** Tendem a reduzir o custo das equações de entrada, mas demandam até o dobro de conexões do que os FF's D e T
- Como os FF's D e T requerem um número menor de conexões, são preferidos para implementações VLSI