

**Uniube**

UNIUBE – CAMPUS VIA CENTRO – Uberlândia/MG

Curso de Engenharia Elétrica e Engenharia de Computação

Disciplina: Sistemas Digitais

Aula 13

Máquinas de estados finitos

Revisão 2, de 29/05/2025

Prof. João Paulo Seno

joao.seno@uniube.br

1

**Uniube**

Máquinas de estados finitos

Os sistemas sequenciais são caracterizados por circuitos que executam sequências de estados predefinidos de acordo com os pulsos de *clock* e com outros sinais de entrada.

Os contadores, estudados nas aulas anteriores, são um exemplo, pois são circuitos destinados a executar sequências numéricas regulares. Os sistemas sequenciais também podem ser projetados para aplicações específicas, como um sistema de semáforos ou um sistema de controle de uma máquina industrial automática. Nesses casos, estes sistemas são chamados de máquinas de estados finitos.



Máquinas de estados finitos

- Uma máquina de estados finitos, de forma geral, é um conceito abstrato que, por meio de um método estruturado, auxilia a análise e a síntese de sistemas sequenciais sincronizados.
- Representa um sistema digital que, quando acionado por um pulso de um *clock* (transição de borda) move-se de um estado lógico para outro.
- É uma ferramenta para o projeto de sistemas sequenciais.



Máquinas de estados finitos

De onde vem este nome?

- O termo “máquina” é usado no sentido matemático, ou de ciência da computação, sendo um objeto conceitual que pode executar uma linguagem abstrata. Nesse caso, esse sentido de máquina não é o de *hardware*.
- O termo “estado” significa um determinado conjunto de valores para as variáveis de saída do sistema sequencial em questão.
- O termo “finito” está presente para diferenciar as máquinas de estado finito de uma representação similar usada na matemática, que pode ter um número infinito de estados.



Uniube

Projeto de um circuito sequencial

Metodologia de aplicação do modelo de máquinas de estados finitos

- É sempre recomendável solucionar um problema primeiro em um nível funcional mais alto, valendo-se da metodologia das máquinas de estados finitos, para depois implementá-lo no nível do *hardware*.
- O projeto de um circuito sequencial pode ser dividido em cinco etapas:
 1. *Desenhar o diagrama de estados do circuito;*
 2. *Identificação dos estados e cálculo da quantidade de flip-flops necessária;*
 3. *Obter as expressões lógicas das entradas dos flip-flops;*
 4. *Obter as expressões lógicas das saídas dos flip-flops;*
 5. *Desenhar o circuito.*
- A seguir, vamos falar um pouco mais de cada uma destas etapas.



Uniube

Desenhar o diagrama de estados do circuito

Etapas 1

- Mas o que é um diagrama de estado?
- É uma representação gráfica de um sistema sequencial que especifica as transições entre os estados do sistema, as quais, por sua vez, descrevem como um estado particular passa para outro.
- Admite-se que um sistema sequencial pode ser convenientemente representado por um diagrama de estado.
- Um diagrama de estados é composto por um conjunto de círculos em que cada um deles corresponde a um estado e por arcos, que indicam as transições possíveis. A informação anotada junto aos arcos indicam o como tal transição ocorre (valor das entradas). Veja o próximo *slide*.

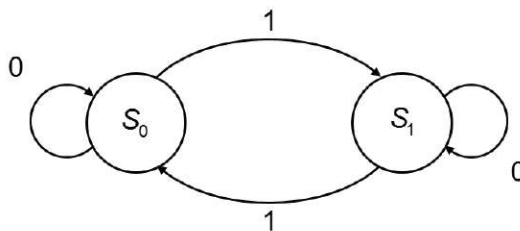


Uniube

Desenhar o diagrama de estados do circuito

Etapa 1

- A figura abaixo traz como exemplo um diagrama de estado genérico para um sistema com uma única entrada e dois estados possíveis, S_0 e S_1 . As setas (ou arcos) indicam as transições entre os estados para cada ação (entrada) em um determinado estado.



Interpretação: O sistema reage à transição de borda do *clock*. Se o sistema encontra-se no estado S_0 e o valor da entrada é 0, ele continua no estado S_0 . Se a entrada for 1, ele muda para o estado S_1 . Já se o sistema se encontra no estado S_1 , e a entrada é zero, ele permanece no S_1 percebe a transição de borda do *clock*. Se a entrada for 1, ele muda para o estado S_0 .



Uniube

Identificação dos estados e cálculo da quantidade de *flip-flops* necessária – Etapa 2

- A quantidade de *flip-flops* necessária, dada por:

$$n_{ff} = \text{int}(\log_2 n_s)$$

- O valor n_{ff} é o número de *flip-flops* e n_s é o número de estados do sistema sequencial.
- A função $\text{int}()$ indica o número inteiro menor ou igual à variável.
- Exemplo: O número de *flip-flops* necessários para um sistema sequencial de 10 estados (n_s) é dado por:

$$n_{ff} = \text{int}(\log_2 10) = \text{int}(3,322) = 3$$

**Uniube**

Obter as expressões lógicas das entradas dos *flip-flops* – Etapa 3

- Obter as expressões lógicas das variáveis de entrada dos *flip-flops* em função do estado atual e das variáveis de entrada.
- Usar, para isso, tabelas verdade e/ou mapas de Karnaugh (que já estudamos nas aulas anteriores).

**Uniube**

Obter as expressões lógicas das saídas dos *flip-flops* – Etapa 4

- Obter as expressões lógicas das saídas dos *flip-flops* em função do estado atual.
- Usar, para isso, tabelas verdade e/ou mapas de Karnaugh.

**Uniube**

Desenhar o circuito

Etapa 5

- Desenhar o circuito lógico com os *flip-flops* e portas lógicas.

**Uniube**

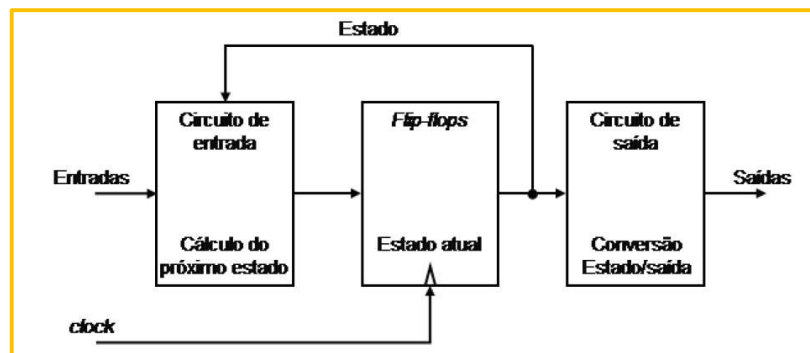
Modelos para Implementação

Máquinas Moore e máquinas Mealy

- *Máquinas Moore e máquinas Mealy* são dois modelos tradicionais para a implementação de máquinas de estados.
- Vamos conhecer as características de cada uma delas para depois fazer uma aplicação prática da metodologia apresentada anteriormente e compreender melhor como funciona.

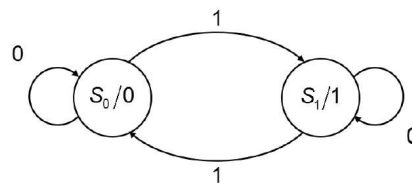
Máquinas Moore

- Nas *máquinas Moore*, as saídas dependem apenas do estado atual registrado internamente no sistema.
- As mudanças das saídas ocorrem apenas na próxima transição de borda do *clock*. Abaixo, tem-se o diagrama de blocos de uma *máquina Moore*.



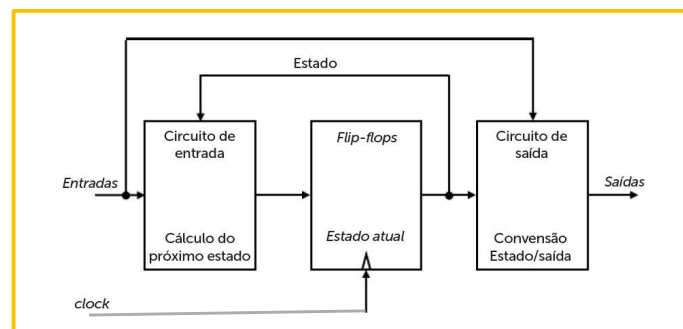
Máquinas Moore

- Analisando o diagrama anterior, vê-se que o *clock* comanda a memória que envia para o circuito de saída o estado atual, que é resultado da combinação entre as entradas e o estado anterior.
- Assim, as saídas assumem um novo estado somente na próxima transição de borda do *clock*.
- A figura abaixo traz um diagrama de estado genérico de uma *máquina Moore* para um sistema com uma entrada, uma saída e dois estados possíveis. Note que a saída é indicada em cada estado.



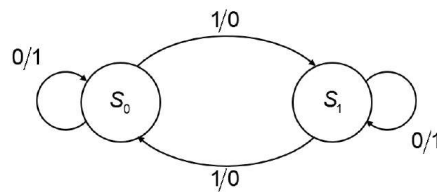
Máquinas Mealy

- Nas *máquinas Mealy*, as saídas dependem tanto das entradas quanto do estado atual registrado internamente no sistema. Se houver mudanças nas entradas, as saídas também se alterarão. A figura abaixo apresenta em blocos a estrutura de um sistema sequencial que utiliza o modelo de *Mealy*.



Máquinas *Mealy*

- Observando a figura anterior, pode-se notar que o *clock* comanda o registrador que envia os estados armazenados resultantes das entradas e estado anteriormente existente para estágio de saída, porém, as saídas também recebem as informações diretamente das entradas.
- Nesse caso, as saídas podem ser alteradas pela mudança dos estados de entradas, mesmo antes da atuação do *clock*. A figura abaixo traz um diagrama de estado genérico de uma *máquina Mealy* para um sistema com uma entrada, uma saída e dois estados possíveis. Note que agora a saída é indicada em cada transição de estado.



Resumindo

Nas **máquinas Moore**, as saídas dependem **apenas** do estado atual registrado internamente no sistema. Já nas **máquinas Mealy**, as saídas dependem tanto das entradas quanto do estado atual registrado internamente no sistema.

Máquinas de estado síncronas e assíncronas

19

Máquinas de estado síncronas e assíncronas

- Os sistemas sequenciais síncronos são aqueles em que os circuitos operam de maneira sincronizada com a entrada *clock*, ou seja, todos os elementos, *flip-flops* ou registradores de estado são comandados pelo mesmo pulso de *clock*, pois estes têm ligação dessa entrada em comum.
- Já os sistemas sequenciais assíncronos são aqueles que atuam de maneira assíncrona, ou seja, sem sincronismo entre os elementos do circuito, sendo que as variações internas e de saída ocorrem em instantes diferenciados com relação ao pulso de *clock*.
- Observando as estruturas estudadas que a atuação do modelo de *Moore* é síncrona, e no modelo de *Mealy* as saídas podem atuar assincronamente



Exemplo de aplicação

- Vamos projetar uma máquina de estados para atuar como contador de três bits para efetuar a contagem crescente (Se a entrada $E = 0$, conta de 0 a 7) ou decrescente (Se Se a entrada $E = 1$, conta de 7 a 0).
- O circuito deverá possuir também uma saída S que sinaliza se a contagem é crescente ($S = 1$) ou decrescente ($S = 0$).
- Para isso, usar *flip-flops* do tipo J-K.
- A Tabela 4.10 traz a sequência de contagem proposta e a Tabela 4.11 traz a tabela verdade do *flip-flop* J-K, como apoio, para relembrarmos seu funcionamento.
- Veja o próximo *slide*.



Exemplo de aplicação

Tabela 4.10 | Sequência de contagem proposta

E	Q_2	Q_1	Q_0	Decimal
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	1	1	1	7
1	1	1	0	6
1	1	0	1	5
1	1	0	0	4
1	0	1	1	3
1	0	1	0	2
1	0	0	1	1
1	0	0	0	0

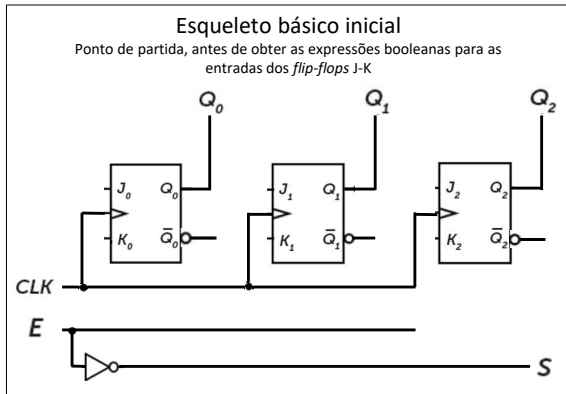
Tabela 4.11 | Tabela verdade do *flip-flop* J-K

$Q(t)$	$Q(t+1)$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0



Exemplo de aplicação

- Passo 3:** a Tabela 4.12 reproduz a máquina de estados, apresentando os estados e sua tradução para a saída dos *flip-flops* J-K.



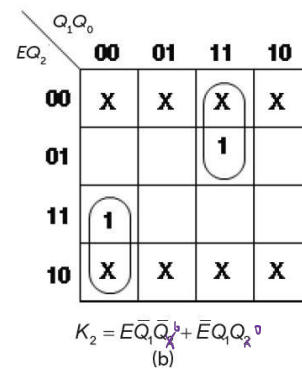
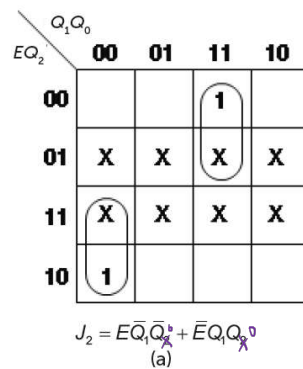
E	Q ₂	Q ₁	Q ₀	J ₂	K ₂	J ₁	K ₁	J ₀	K ₀	S
0	0	0	0	0	X	0	X	1	X	1
0	0	0	1	0	X	1	X	X	1	1
0	0	1	0	0	X	X	0	1	X	1
0	0	1	1	1	X	X	1	X	1	1
0	1	0	0	X	0	0	X	1	X	1
0	1	0	1	X	0	1	X	X	1	1
0	1	1	0	X	0	X	0	1	X	1
0	1	1	1	X	1	X	1	X	1	1
1	1	1	1	X	0	X	0	X	1	0
1	1	1	0	X	0	X	1	1	X	0
1	1	0	1	X	0	0	X	X	1	0
1	1	0	0	X	1	1	X	1	X	0
1	0	1	1	0	X	X	0	X	1	0
1	0	1	0	0	X	X	1	1	X	0
1	0	0	1	0	X	0	X	X	1	0
1	0	0	0	1	X	1	X	1	X	0



Exemplo de aplicação

- Passo 4:** A Figura 4.38 apresenta os mapas de Karnaugh correspondentes a cada entrada dos *flip-flops* e da saída S e também as expressões simplificadas para cada entrada dos *flip-flops* e da saída S.

Figura 4.38 | Mapas de Karnaugh com simplificações (a) J₂ (b) K₂ (c) J₁ (d) K₁ (e) J₀ (f) K₀ (g) S





Exemplo de aplicação

- **Passo 4:** continuação...

		Q_1Q_0			
		00	01	11	10
EQ_2	00		1	X	X
	01		1	X	X
	11	1		X	X
	10	1		X	X

$J_1 = E\bar{Q}_0 + \bar{E}Q_0 = E \oplus Q_0$

(c)

		Q_1Q_0			
		00	01	11	10
$\bar{E}Q_2$	00	X	X	1	
	01	X	X	1	
	11	X	X		1
	10	X	X		1

$K_1 = E\bar{Q}_0 + \bar{E}Q_0 = E \oplus Q_0$

(d)



Exemplo de aplicação

- **Passo 4:** continuação...

		Q_1Q_0			
		00	01	11	10
EQ_2	00	1	X	X	1
	01	1	X	X	1
	11	1	X	X	1
	10	1	X	X	1

$J_0 = 1$

(e)

		Q_1Q_0			
		00	01	11	10
EQ_2	00	X	1	1	X
	01	X	1	1	X
	11	X	1	1	X
	10	X	1	1	X

$K_0 = 1$

(f)

		Q_1Q_0			
		00	01	11	10
$\bar{E}Q_2$	00	1	1	1	1
	01	1	1	1	1
	11				
	10				

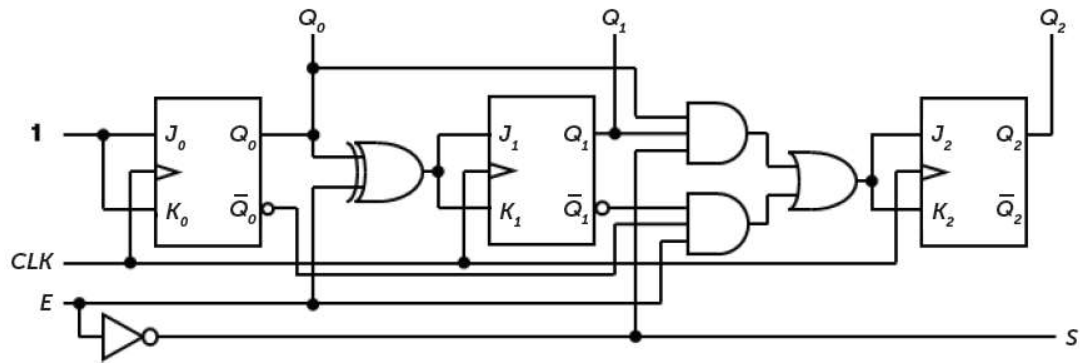
$S = \bar{E}$

(g)



Exemplo de aplicação

- **Passo 5:** o desenho do circuito final é apresentado na figura abaixo.



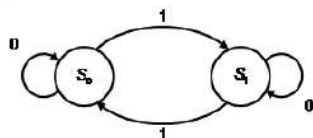
Uniube

Testes do final da seção



1. Um *diagrama de estado* representa um modelo gráfico de um sistema sequencial e especifica as transições entre os estados do sistema, as quais, por sua vez, descrevem como um estado particular passa para outro. Um sistema sequencial pode convenientemente ser representado por um diagrama de estado, como o diagrama de estado genérico representado na Figura 3.57.

Figura 3.57 | Diagrama lógico para o circuito contador de carros no estacionamento



Fonte: elaborada pelo autor.

Nesse contexto, avalie as afirmativas a seguir:

I. Em um diagrama de estados, cada círculo representa um estado, de modo que, no diagrama do sistemas representado na Figura 3.57, há dois estados, S_0 e S_1 .

II. As setas representam as transições de estado, de um determinado sistema, devido a uma ação (entrada) em um certo estado.

III. O sistema da Figura 3.57, por possuir apenas dois estados, pode ser representado com apenas 1 bit, ou seja, apenas 1 *flip-flop*.

IV. Considerando o sistema no estado S_0 , uma entrada igual a 1 leva o sistema para o estado S_1 .

É correto o que se afirma em:

- I e III, apenas.
- II e IV, apenas.
- II, III e IV, apenas.
- I, II e III, apenas.
- I, II, III e IV.



2. Uma máquina de estados finitos é um sistema algébrico usado para modelar programas de computador ou circuitos lógicos. Em tese, todos os circuitos síncronos, incluindo os registradores e contadores movidos por um *clock*, podem ser representados por máquinas de estados finitos.

Sobre os modelos de máquinas de estado finito, avalie as afirmações a seguir:

I. Existem dois modelos tradicionais para a implementação de máquinas de estados. São eles: as máquinas Moore e as máquinas Mealy. Embora a implementação desses dois modelos ocorra de forma diferente, o comportamento das duas máquinas modelando um mesmo sistema é idêntico.

II. Nas máquinas Moore, a saída depende da entrada e do estado atual, de modo que o uso de uma máquina Moore geralmente leva a uma redução no número de estados.

III. Nas máquinas Mealy, a saída depende somente da entrada, o que costuma simplificar o comportamento da máquina.

É correto o que se afirma em:

- I, apenas.
- II e III, apenas.
- I e II, apenas.
- I e III, apenas.
- I, II e III.



3. As máquinas de estado também podem ser classificadas como sistemas síncronos ou assíncronos, conforme as características de aplicação do sistema de clock nos circuitos. Os sistemas sequenciais síncronos são aqueles em que os circuitos operam de maneira sincronizada com a entrada clock. Já os sistemas sequenciais assíncronos são aqueles atuam de maneira assíncrona, ou seja, sem sincronismo entre os elementos do circuito.

Nesse contexto, avalie as afirmativas a seguir:

I. Em um sistema assíncrono, todos os elementos, *flip-flops* ou registradores de estado são comandados pelo mesmo pulso de *clock*, em função do aclonamento no tempo, uma vez que têm ligação dessa entrada em comum.

II. Em um sistema síncrono, as variações internas e de saída ocorrem em instantes diferenciados com relação ao pulso de *clock*.

III. Uma máquina Moore é considerada um sistema síncrono.

IV. Uma máquina Mealy é considerada um sistema assíncrono.

É correto o que se afirma em:

- a) I e II, apenas.
- b) III e IV, apenas.
- c) I e III, apenas.
- d) II e IV, apenas.
- e) I, II, III e IV.



4. Todos os circuitos síncronos, incluindo os registradores e contadores movidos por um *clock*, podem ser representados por máquinas de estados finitos. É sempre recomendável solucionar um problema em um nível funcional mais alto, valendo-se da metodologia das máquinas de estados finitos.

O projeto de um circuito sequencial pode ser dividido em cinco etapas:

I. Obter as expressões lógicas das entradas em função do estado atual.

II. Obter as expressões lógicas das saídas em função do estado atual.

III. Desenhar o diagrama de estados do circuito.

IV. Identificar os estados e calcular a quantidade de *flip-flops* necessários.

V. Desenhar o circuito.

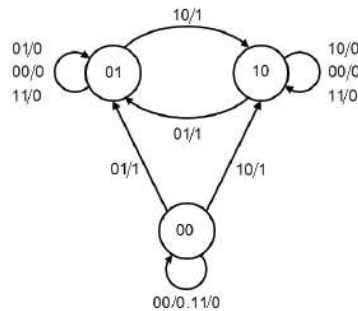
Assinale a opção que apresenta a ordem correta dos procedimentos realizados.

- a) I; II; III; IV; V.
- b) V; III; I; II; IV.
- c) III; IV; I; II; V.
- d) V; IV; I; II; III.
- e) IV; I; I; III; V.



5. Um **diagrama de estado** representa um modelo gráfico de um sistema sequencial e especifica as transições entre os estados do sistema, as quais, por sua vez, descrevem como um estado particular passa para outro. Considere o diagrama de estado hipotético da Figura 4.42.

Figura 4.42 | Diagrama de estado hipotético



Fonte: elaborada pelo autor.

Sobre este diagrama de estado, avalie as afirmações a seguir:

- I. A máquina de estado representada por esse diagrama possui quatro estados, sendo que o estado 11 foi suprimido do diagrama.
- II. Essa máquina de estado possui duas entradas, e todas as possibilidades foram representadas para cada estado.
- III. A máquina de estado apresenta uma saída que indica um nível lógico ALTO quando há transição de um estado para outro.
- IV. No estado 01, as entradas 01 e 10 produzem o mesmo efeito.

Quais afirmações estão corretas?

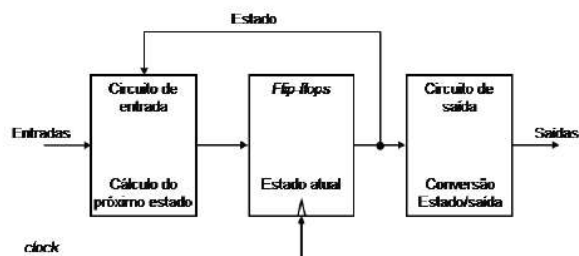
- a) I, II e III.
- b) I, III e IV.
- c) II e III.
- d) III.
- e) II, III e IV.



6. Existem dois modelos tradicionais para a implementação de máquinas de estados, são eles: as **máquinas Moore** e as **máquinas Mealy**. Sobre os modelos de máquinas de estado finito, avalie as afirmações a seguir:

- I. Nas máquinas Moore, as saídas dependem apenas do estado atual registrado internamente no sistema.
- II. Nas máquinas Mealy, as saídas dependem das entradas e do estado atual registrado internamente no sistema.
- III. O diagrama de blocos da Figura 4.43 corresponde a uma máquina Mealy.

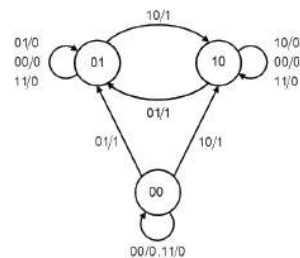
Figura 4.43 | Diagrama de blocos



Fonte: elaborada pelo autor.

IV. O diagrama de estado da Figura 4.44 é referente a uma máquina Moore.

Figura 4.44 | Diagrama de estado



Fonte: elaborada pelo autor.

É correto o que se afirma em:

- a) I e II.
- b) III e IV.
- c) I e III.
- d) II e IV.
- e) I, II, III e IV.



Fim

37