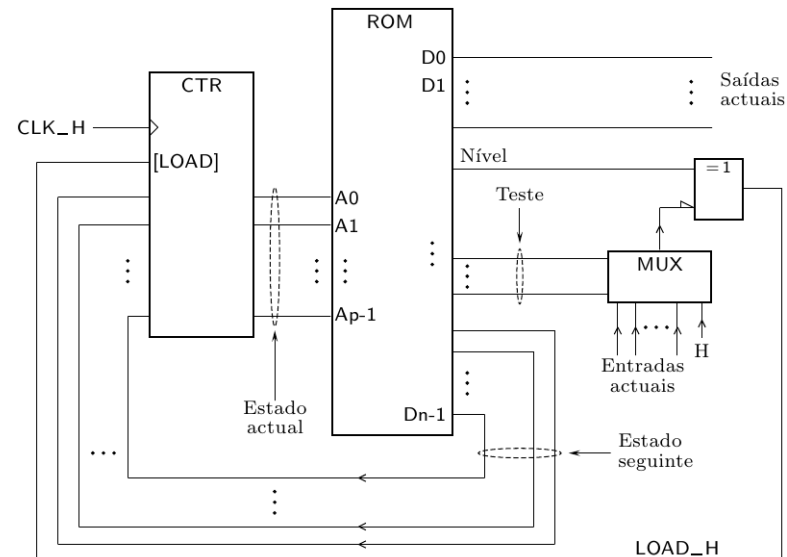


Sistemas Digitais (SD)

Máquinas de Estado Microprogramadas: Microprograma



■ Na aula anterior:

- ▶ Projecto de máquinas de estados microprogramadas:
 - Circuito de dados
 - Circuito de controlo
- ▶ Implementação com ROMs
- ▶ Exemplos



SEMANA	TEÓRICA 1	TEÓRICA 2	PROBLEMAS/LABORATÓRIO
17/Fev a 21/Fev	Introdução	Sistemas de Numeração	
24/Fev a 28/Fev	CARNAVAL	Álgebra de Boole	P0
02/Mar a 06/Mar	Elementos de Tecnologia	Funções Lógicas	VHDL
9/Mar a 13/Mar	Minimização de Funções	Minimização de Funções	L0
16/Mar a 20/Mar	Def. Circuito Combinatório; Análise Temporal	Circuitos Combinatórios	P1
23/Mar a 27/Mar	Circuitos Combinatórios	Circuitos Combinatórios	L1
30/Mar a 03/Abr	Circuitos Sequenciais: Latches	Circuitos Sequenciais: Flip-Flops	P2
06/Abr a 10/Abr	FÉRIAS DA PÁSCOA	FÉRIAS DA PÁSCOA	FÉRIAS DA PÁSCOA
13/Abr a 17/Abr	Caracterização Temporal	Registos	L2
20/Abr a 24/Abr	Contadores	Circuitos Sequenciais Síncronos	P3
27/Abr a 01/Mai	Síntese de Circuitos Sequenciais Síncronos	Síntese de Circuitos Sequenciais Síncronos	L3
04/Mai a 08/Mai	Exercícios	Memórias	P4
11/Mai a 15/Mai	Máq. Estado Microprogramadas: Circuito de Dados e Circuito de Controlo	Máq. Estado Microprogramadas: Microprograma	L4
18/Mai a 22/Mai	Circuitos de Controlo, Transferência e Processamento de Dados de um Processador	Lógica Programável	P5
25/Mai a 29/Mai	P6	P6	L5

Teste 1

■ Tema da aula de hoje:

- ▶ Projecto de máquinas de estados microprogramadas:
 - com endereçamento explícito
 - com endereçamento implícito
- ▶ Exemplos

□ Bibliografia:

- M. Mano, C. Kime: Secção 7.13
- G. Arroz, J. Monteiro, A. Oliveira: Secção 7.5
- G. Arroz, C. Sêro, "Sistemas Digitais: Apontamentos das Aulas Teóricas", IST, 2005: Secções 19.1 a 19.3 (disponível no [Fenix](#))

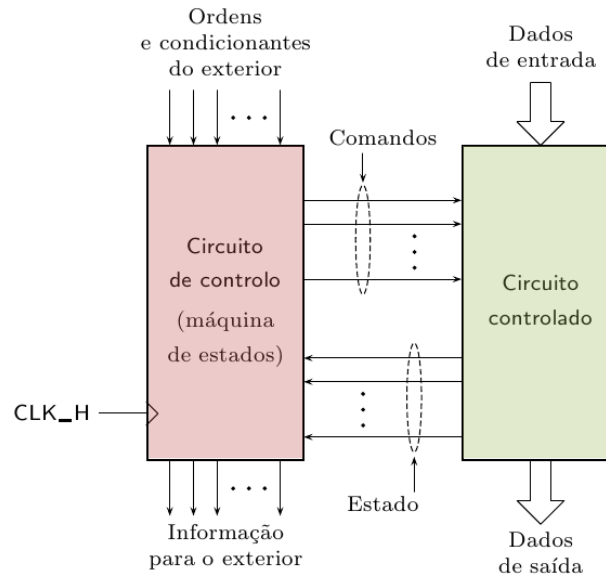
■ Circuito de Dados e Circuito de Controlo (Revisão)

- ▶ Os sistemas digitais com alguma complexidade tornam-se difíceis de ser projectados como vulgares máquinas sequenciais síncronas, porque:
 - Diagramas de estados / tabela de estados de grande dimensão
 - Elevado número de:
 - Entradas
 - Saídas
 - Estados.

Solução: organizar esses sistemas hierarquicamente, estabelecendo uma divisão clara entre:

- **circuito de dados** - dá suporte ao fluxo e à manipulação de dados;
- **circuito de controlo** - controla o circuito de dados.

■ Circuito de Dados e Circuito de Controlo (Revisão)



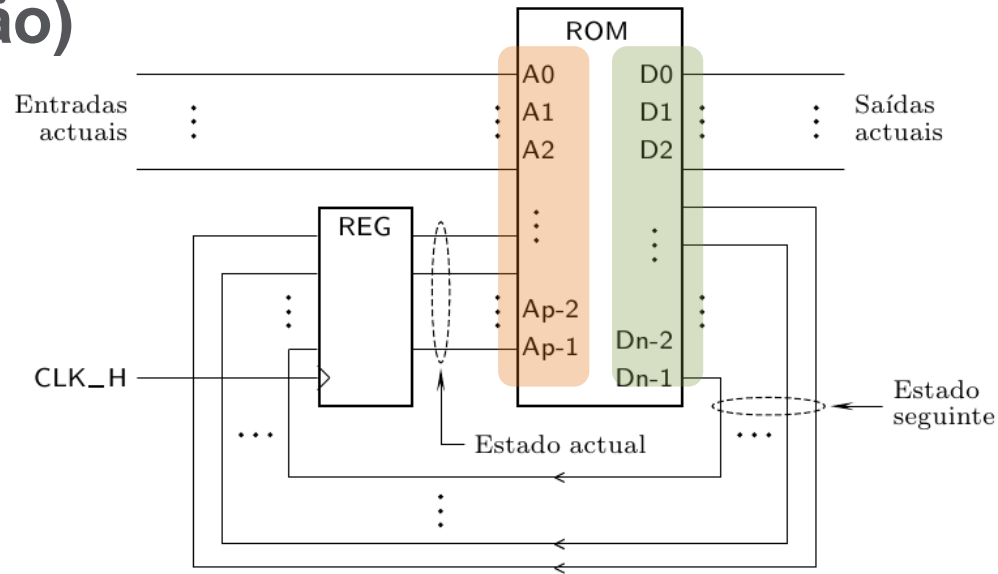
► Em geral:

- O **circuito de dados** (controlado) é formado por um conjunto de módulos simples, tais como contadores, registos, multiplexeres, somadores, comparadores, memórias, algumas portas lógicas, etc, podendo ser combinatório ou sequencial.
- O **circuito de controlo** é sempre um circuito sequencial síncrono.

■ Controlo por ROM (Revisão)

- ▶ A memória **ROM** substitui a lógica combinatória para gerar:

- Estado seguinte
- Saída do circuito



- ▶ Entradas da **ROM** (**barramento de endereços**):

- Entradas externas da máquina de estados
- Estado actual

- ▶ Saídas da **ROM** (**barramento de dados**):

- Saídas para o exterior da máquina de estados
- Saídas (comandos) internas + estado seguinte

Exemplo

■ Exemplo (simples)

► 3 estados:

- A ($Q_1Q_0=00$)
- B ($Q_1Q_0=01$)
- C ($Q_1Q_0=10$)

► 2 entradas: I0 e I1

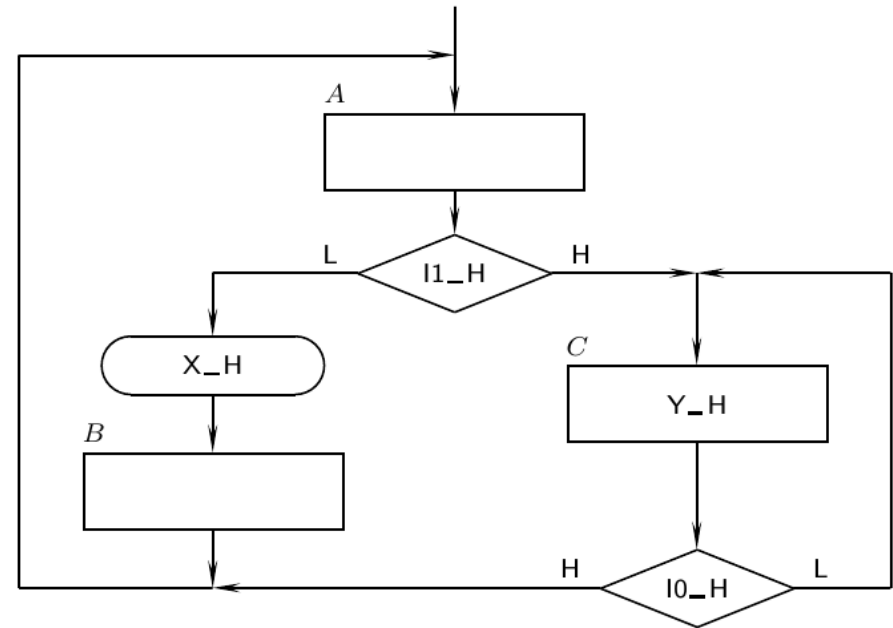
► 2 saídas: X e Y

► Formato da palavra na ROM:

$NQ1$	$NQ0$	Y	X
-------	-------	-----	-----

► Dimensão mínima da ROM:

- 12 endereços de 4 bits



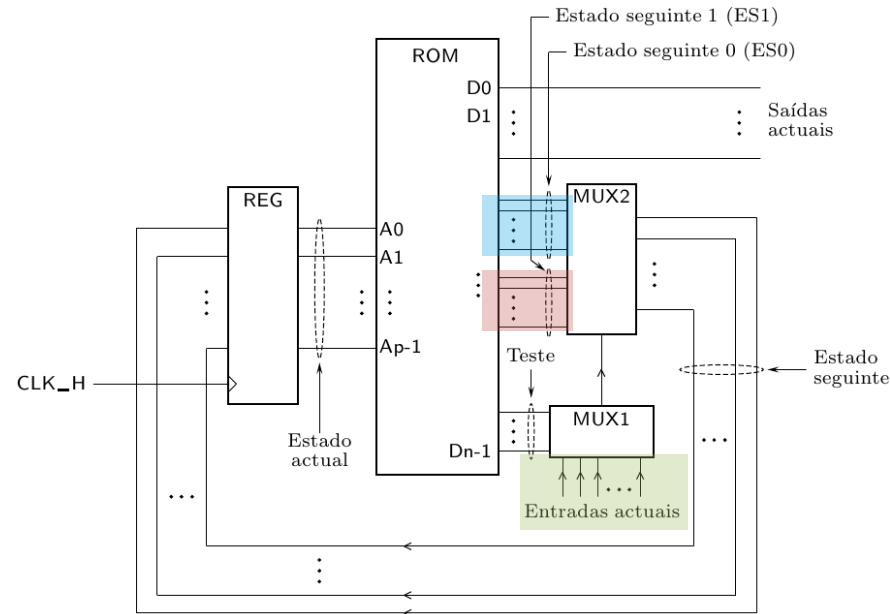
**Como reduzir a dimensão da ROM
(nº de endereços)?**



CONTROLO POR ROM COM ENDEREÇAMENTO EXPLÍCITO

Controlo por ROM com Endereçamento Explícito

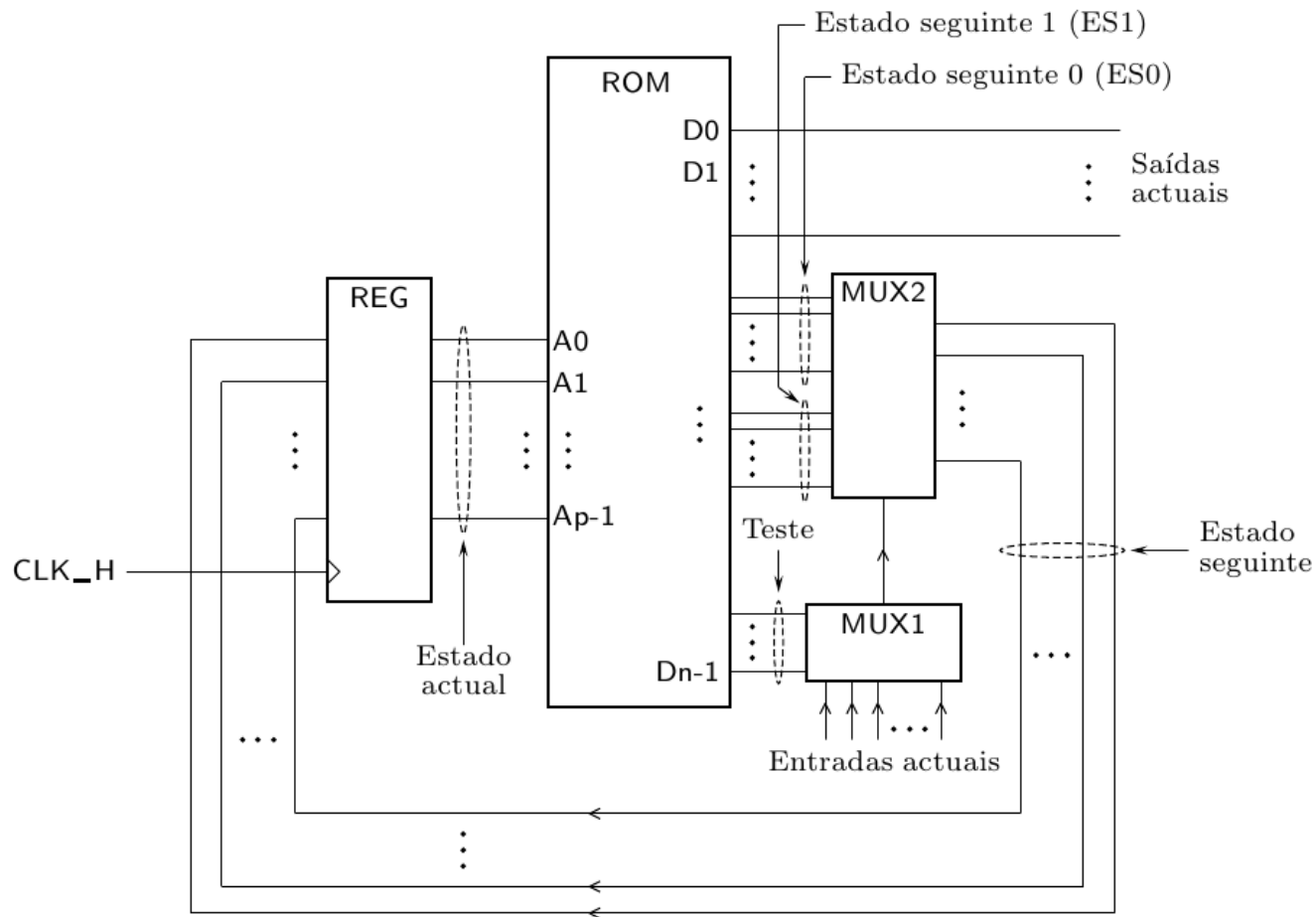
■ Controlo por ROM com Endereçamento Explícito



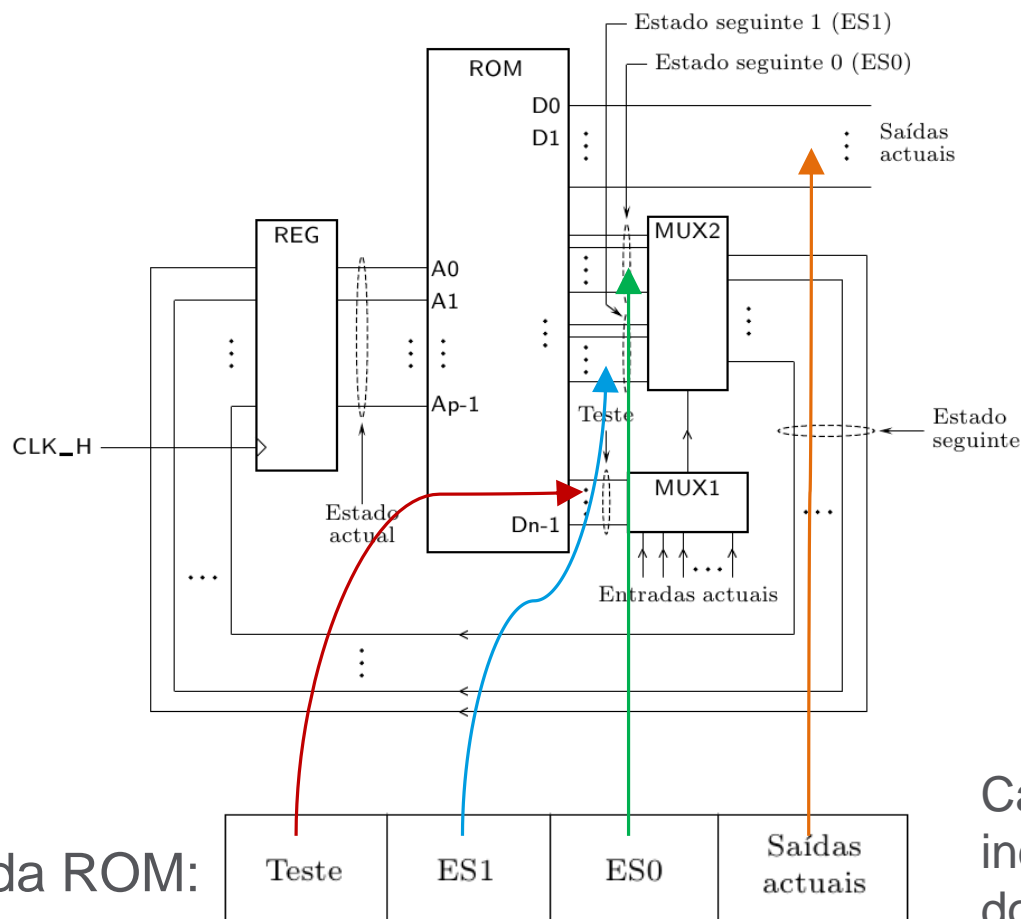
- Para reduzir o número de endereços, as entradas externas são retiradas do barramento de endereços:
 - Eliminar o efeito das entradas nas saídas, transformando a máquina de **Mealy** em máquina de **Moore**;
 - Cada estado actual só pode evoluir para um de dois estados seguintes (incluindo, eventualmente, o próprio).

Controlo por ROM com Endereçamento Explícito

■ Controlo por ROM com Endereçamento Explícito

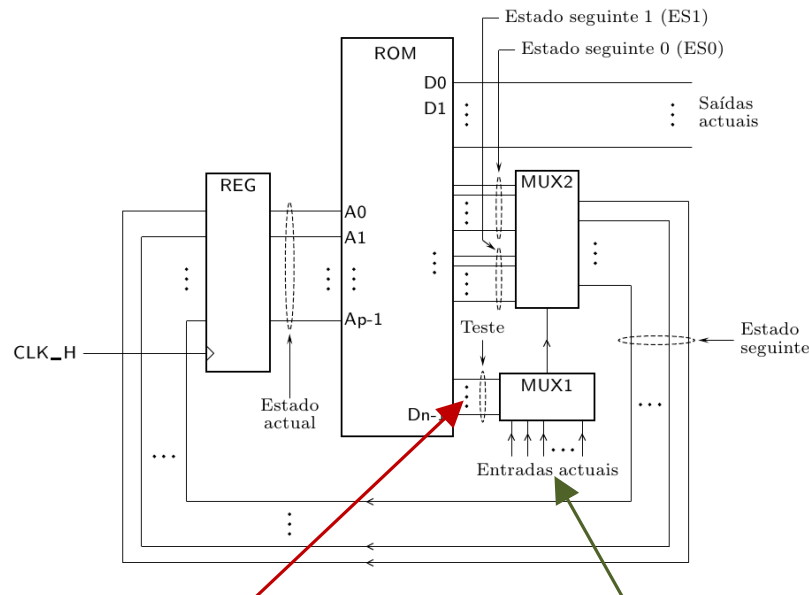


■ Controlo por ROM com Endereçamento Explícito



Cada palavra da ROM indica **explicitamente** dois estados seguintes

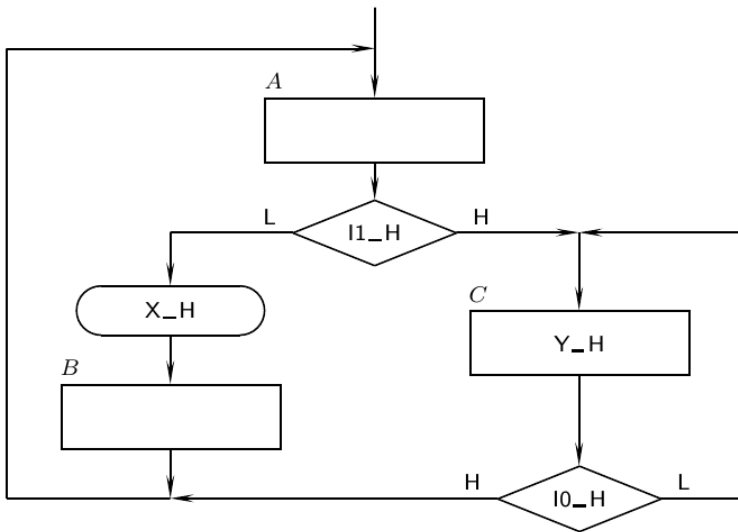
■ Controlo por ROM com Endereçamento Explícito



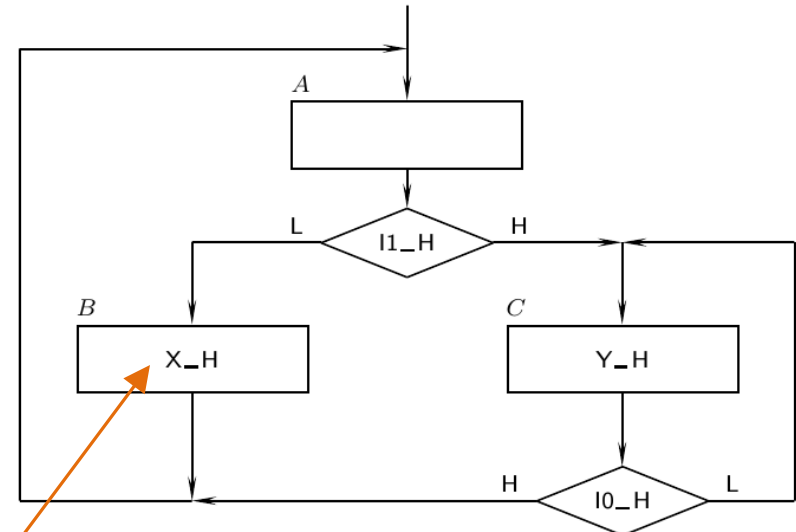
- ▶ O **MUX1** tem as suas entradas ligadas às **entradas primárias** do circuito de controlo. O campo de **teste** da ROM permite, para cada estado actual, escolher a entrada ou combinações de entradas a testar.
 - Se a entrada seleccionada tiver o valor 0, o estado seguinte escolhido é o que vier indicado no campo ES0;
 - No caso contrário, será o estado seguinte ES1.

■ Exemplo (simples)

- O fluxograma original (máq. **Mealy**) vai ter de ser transformado, de modo a assumir um comportamento do tipo máquina de **Moore**:



Máquina de Mealy

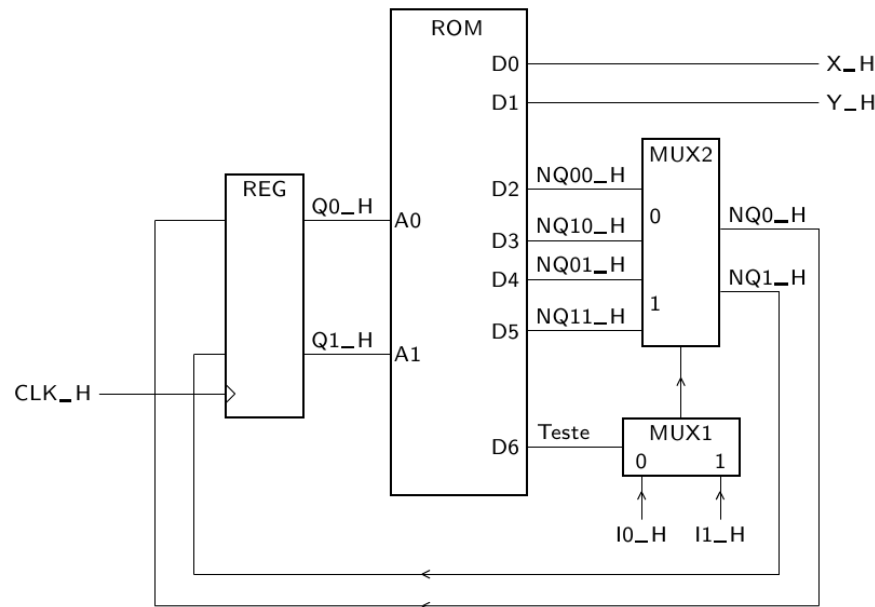


Máquina de Moore

A saída X vem activada
um ciclo mais tarde...

■ Exemplo (simples)

- Diagrama de blocos de um controlador implementado com ROM com **endereçamento explícito**:



- Formato de cada palavra da ROM:

6	5	4	3	2	1	0
Teste	NQ11	NQ01	NQ10	NQ00	Y	X
Teste	ES1		ES0		Saídas actuais	

Exemplo (simples)

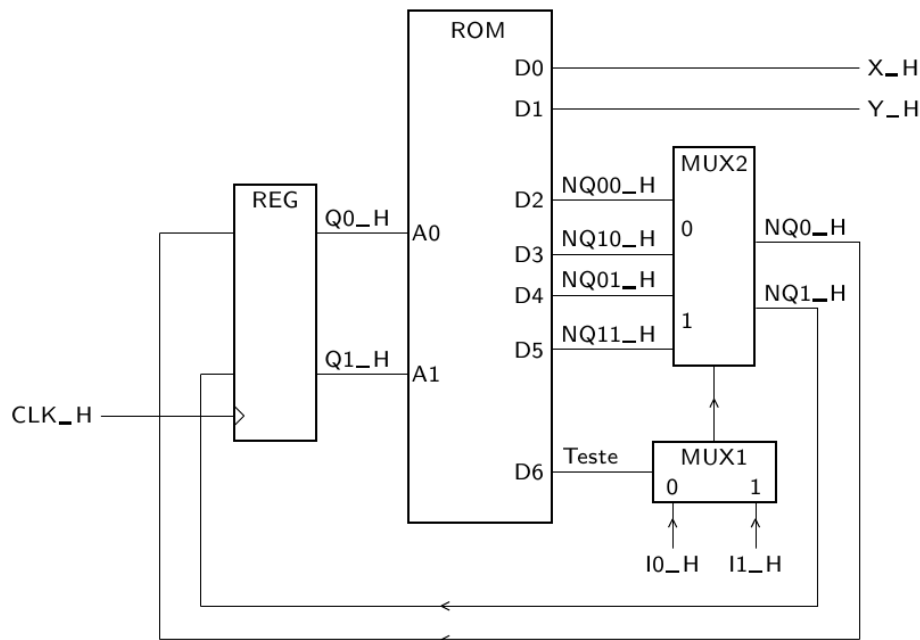
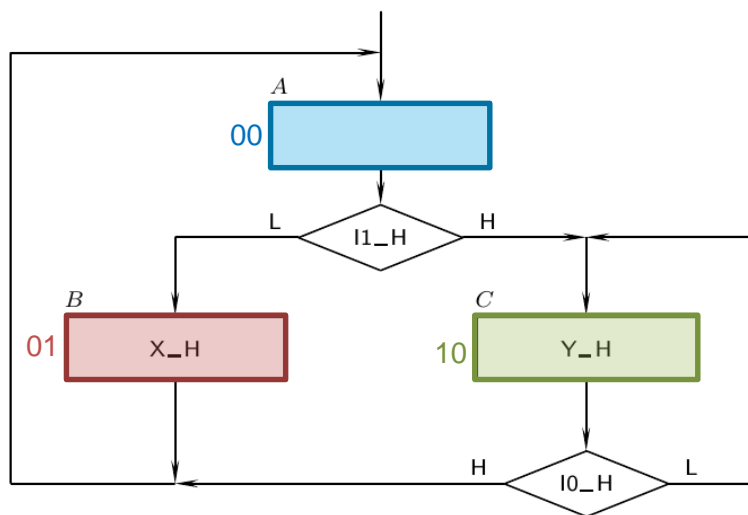


Tabela de transição de estados

Estado	Q1 A1	Q0 A0	Teste D6	NQ11 D5	NQ01 D4	NQ10 D3	NQ00 D2	Y D1	X D0
A	0	0	1	1	0	0	1	0	0
B	0	1	x	0	0	0	0	0	1
C	1	0	0	0	0	1	0	1	0
	1	1	x	x	x	x	x	x	x

■ Exemplo (simples)

Tabela de transição de estados

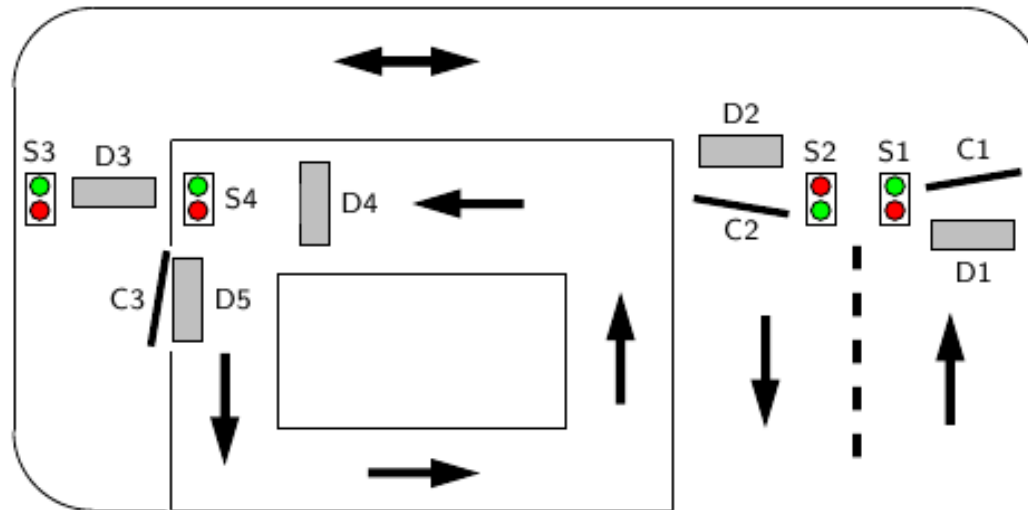
	Q1	Q0	Teste	NQ11	NQ01	NQ10	NQ00	Y	X
Estado	A1	A0	D6	D5	D4	D3	D2	D1	D0
A	0	0	1	1	0	0	1	0	0
B	0	1	×	0	0	0	0	0	1
C	1	0	0	0	0	1	0	1	0
	1	1	×	×	×	×	×	×	×

Conteúdo da ROM

Endereço	Dados
0h	1100100
1h	0000001
2h	0001010
3h	0000000

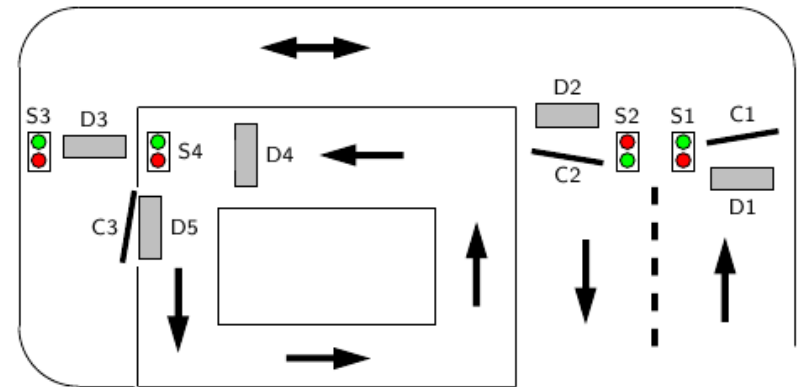
- ▶ Circuito controlador original:
 - 12 endereços de 4 bits (48 bits)
- ▶ Circuito controlador com endereçamento explícito da ROM:
 - 4 endereços de 7 bits (28 bits)

■ Exemplo: acesso a um parque de estacionamento



- ▶ O acesso faz-se por uma via de sentido único, controlada na entrada e na saída pelas cancelas C1 a C3, pelos semáforos S1 a S4, e pelos sensores D1 a D5.
- ▶ O controlador contém um contador ascendente/descendente, que guarda a informação sobre o número de carros estacionados no parque.

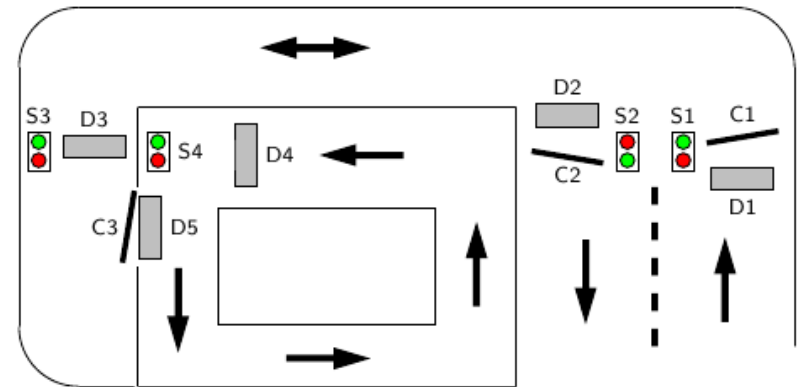
■ Exemplo: acesso a um parque de estacionamento



► Funcionamento:

- A **saída** é detectada pela presença de um carro que pisa D4. Se não há entrada em curso, o semáforo S4 fica verde e a cancela C3 abre. Em seguida, espera-se que o carro pise D5 e saia, para se fechar a cancela e colocar o semáforo S4 em vermelho. Entretanto, coloca-se o semáforo S2 a verde. Quando o carro pisa D2, abre-se C2, que se mantém aberta enquanto a viatura estiver a pisar D2. Quando o carro deixar de pisar D2, o semáforo S2 passa a vermelho e C2 fecha. Nessa altura desconta-se uma unidade no **contador** de lugares ocupados no parque.

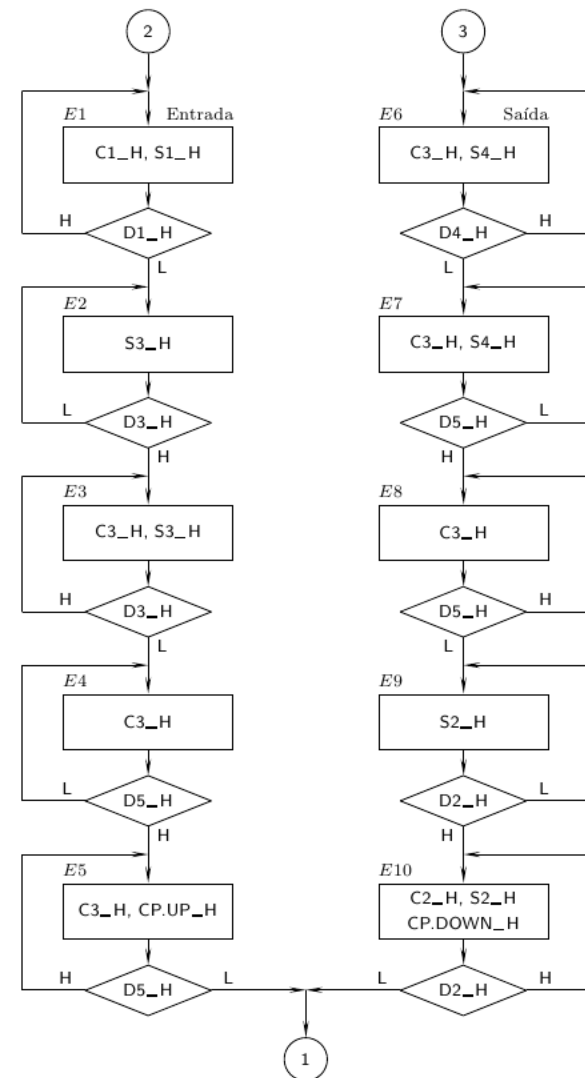
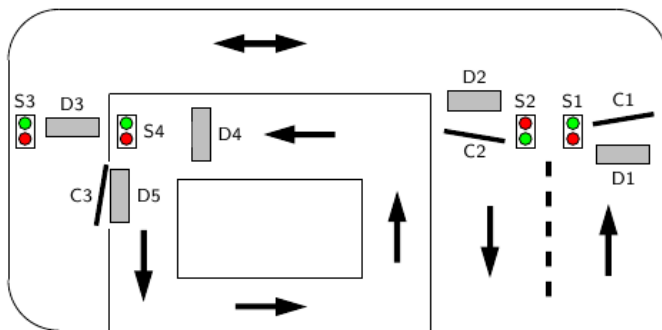
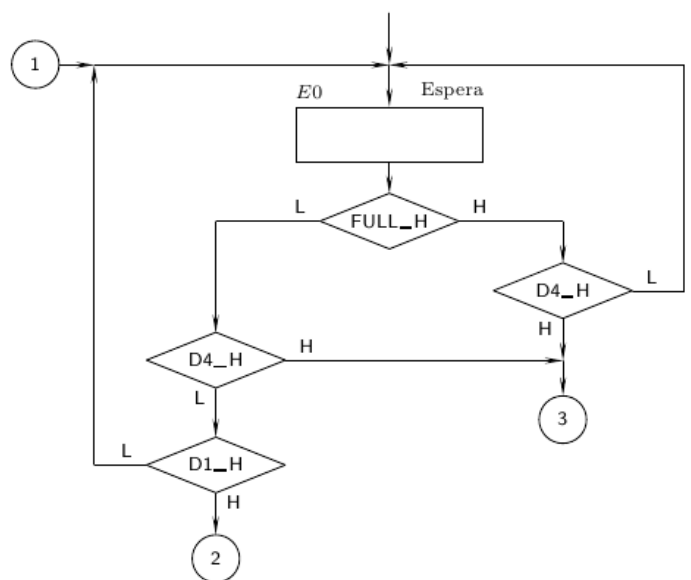
■ Exemplo: acesso a um parque de estacionamento



► Funcionamento:

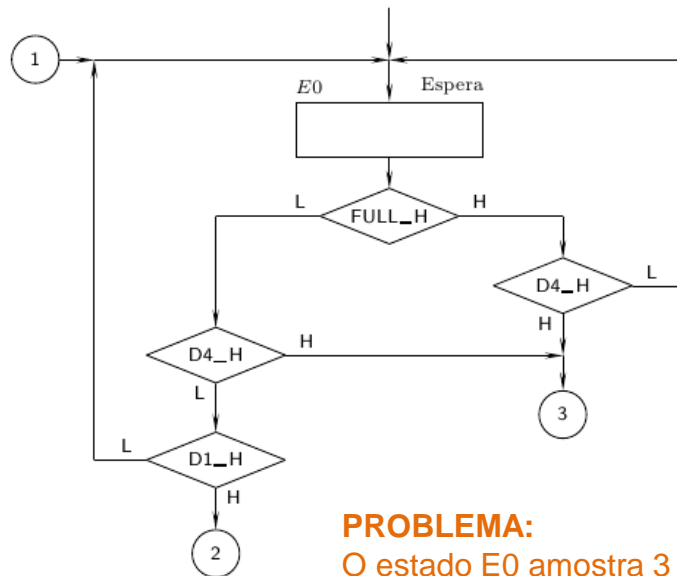
- A **entrada** começa com um carro a pisar D1. Se não há saída em curso, o semáforo S1 fica verde e a cancela C1 abre, ficando aberta enquanto o carro é detectado por D1. Quando o carro deixa D1, S3 fica a verde, e quando chega a D3 a cancela C3 é aberta e o carro entra, passando S3 a vermelho e ficando o circuito à espera que D5 seja pisado. Só depois de D5 deixar de ser pisado é que C3 fecha. Nessa altura, o contador é incrementado.

■ Fluxograma do circuito de controlo



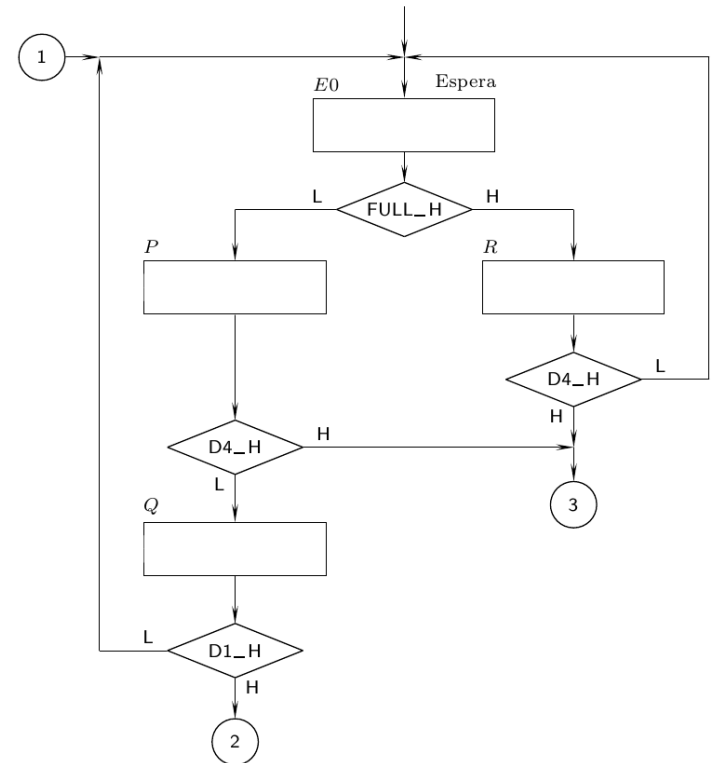
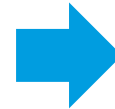
■ Fluxograma do circuito de controlo

- É necessário acrescentar alguns estados para garantir que, de qualquer estado actual, apenas se prossegue para um de dois estados seguintes:

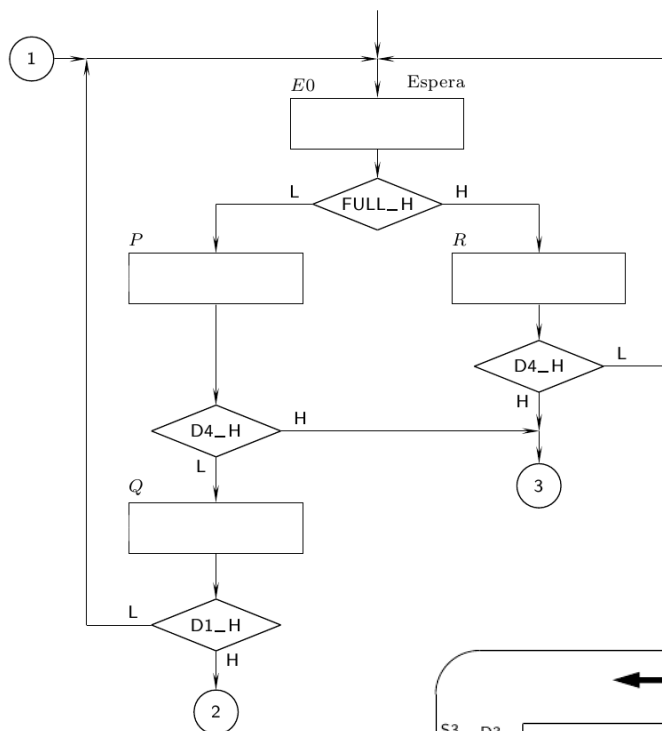


PROBLEMA:

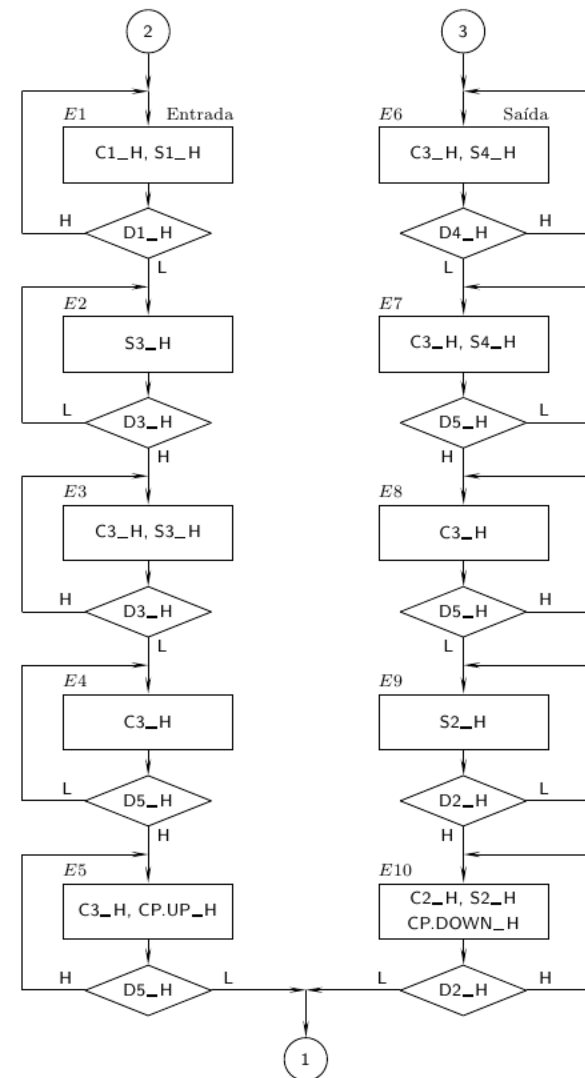
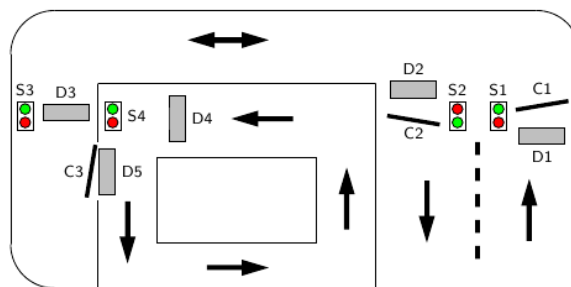
O estado E0 amostra 3 entradas (D1, D4, FULL) e transita para 3 estados possíveis (E0, E1 e E6)



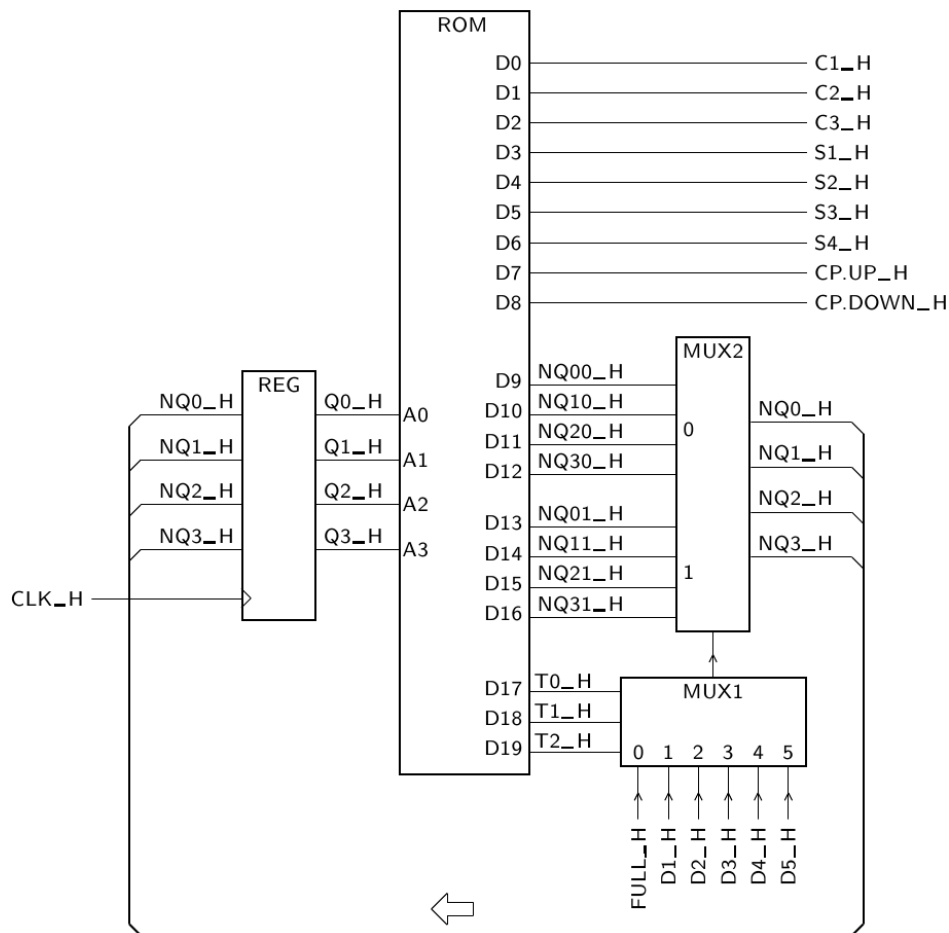
■ Fluxograma do circuito de controlo



14 Estados!!!



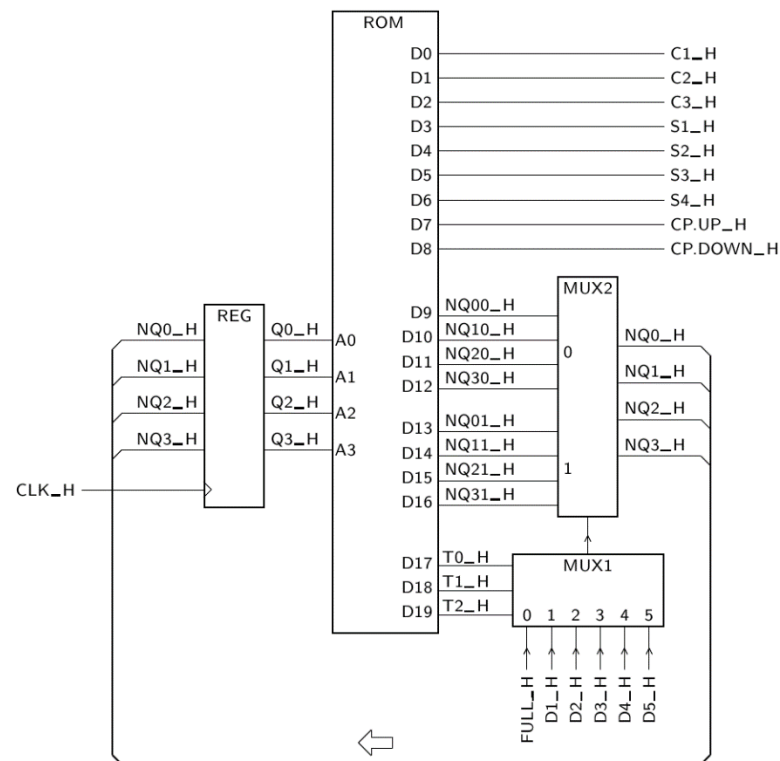
■ Diagrama de blocos da máquina de estados



■ Diagrama de blocos da máquina de estados

Codificação dos estados:

Estado	Q3	Q2	Q1	Q0
<i>E0</i>	0	0	0	0
<i>E1</i>	0	0	0	1
<i>E2</i>	0	0	1	0
<i>E3</i>	0	0	1	1
<i>E4</i>	0	1	0	0
<i>E5</i>	0	1	0	1
<i>E6</i>	0	1	1	0
<i>E7</i>	0	1	1	1
<i>E8</i>	1	0	0	0
<i>E9</i>	1	0	0	1
<i>E10</i>	1	0	1	0
<i>P</i>	1	0	1	1
<i>Q</i>	1	1	0	0
<i>R</i>	1	1	0	1
	1	1	1	0
	1	1	1	1



Palavra da ROM:

19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
<i>T2</i>	<i>T1</i>	<i>T0</i>	<i>NQ31</i>	<i>NQ21</i>	<i>NQ11</i>	<i>NQ01</i>	<i>NQ30</i>	<i>NQ20</i>	<i>NQ10</i>	<i>NQ00</i>	<i>CP.DOWN</i>	<i>CP.UP</i>	<i>S4</i>	<i>S3</i>	<i>S2</i>	<i>S1</i>	<i>C3</i>	<i>C2</i>	<i>C1</i>
Teste			ES1				ES0				Saídas actuais								

■ Diagrama de blocos da máquina de estados

Tabela de Transição de Estados

Estado					Teste			ES1				ES0				Saídas atuais									
	Q3	Q2	Q1	Q0	T2	T1	T0	NQ31	NQ21	NQ11	NQ01	NQ30	NQ20	NQ10	NQ00	CP.DOWN	CP.UP	S4	S3	S2	S1	C3	C2	C1	
	A3	A2	A1	A0	D19	D18	D17	D16	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	
E0	0	0	0	0	0	0	0	1	1	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0	
E1	0	0	0	1	0	0	1	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	1	
E2	0	0	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0	0	
E3	0	0	1	1	0	1	1	0	0	1	1	0	1	0	0	0	0	0	1	0	0	1	0	0	
E4	0	1	0	0	1	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	
E5	0	1	0	1	1	0	1	0	1	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0	
E6	0	1	1	0	1	0	0	0	1	1	0	0	1	1	1	0	0	1	0	0	0	1	0	0	
E7	0	1	1	1	1	0	1	1	0	0	0	0	1	1	1	0	0	1	0	0	0	1	0	0	
E8	1	0	0	0	1	0	1	1	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	
E9	1	0	0	1	0	1	0	1	0	1	0	1	0	0	1	0	0	0	0	1	0	0	0	0	
E10	1	0	1	0	0	1	0	1	0	1	0	0	0	0	0	1	0	0	0	1	0	0	1	0	
P	1	0	1	1	1	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	
Q	1	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
R	1	1	0	1	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
—	1	1	1	0	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	
—	1	1	1	1	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	

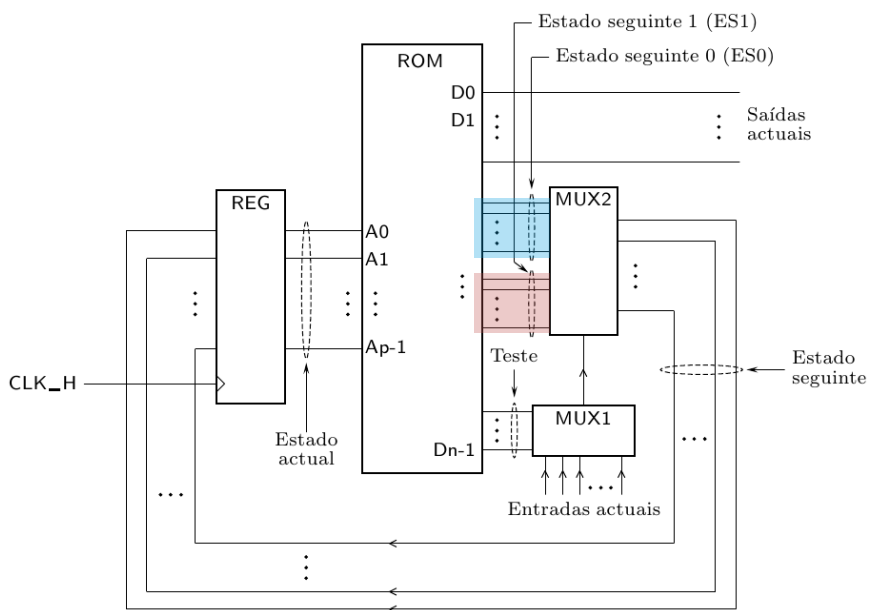
Dimensão da ROM: 16 endereços de 20 bits (320 bits)



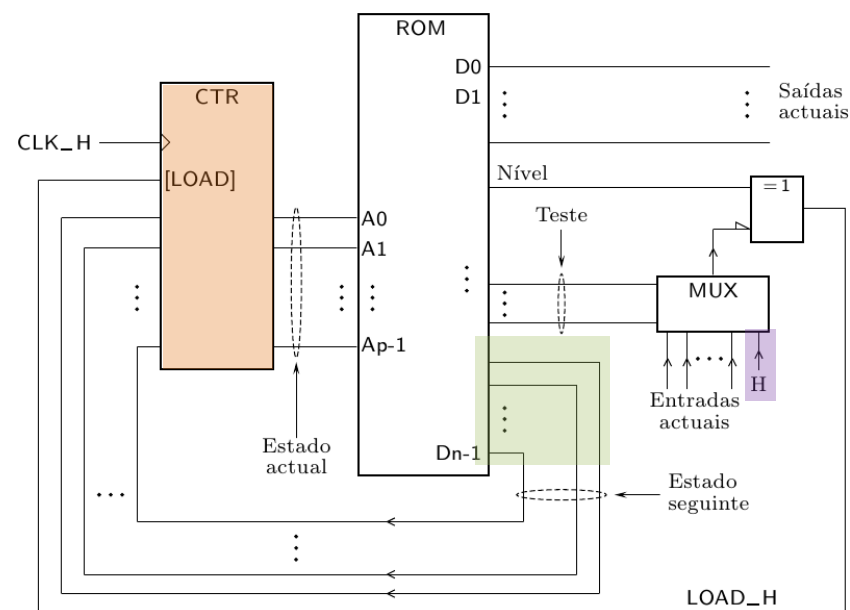
CONTROLO POR ROM COM ENDEREÇAMENTO IMPLÍCITO

■ Controlo por ROM com Endereçamento Implícito

Endereçamento Explícito



Endereçamento Implícito



- Para reduzir as dimensões da ROM, um dos endereços de estado seguinte está **implícito**: **Estado Seguinte = Estado Actual + 1**
 - O registo é substituído por um **contador com carregamento paralelo**.

■ Controlo por ROM com Endereçamento Implícito

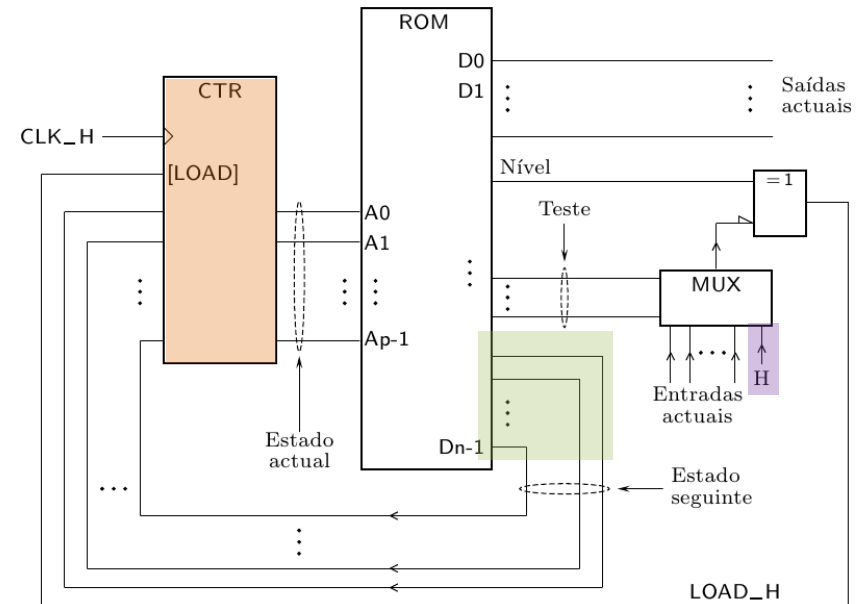
Alterações:

- ▶ O registo é substituído por um **contador com carregamento paralelo**.
- ▶ O **fluxograma** é ajustado de modo a garantir que cada estado evolui para:

- Estado seguinte da contagem:
EstadoSeguinte=EstadoActual+1

ou

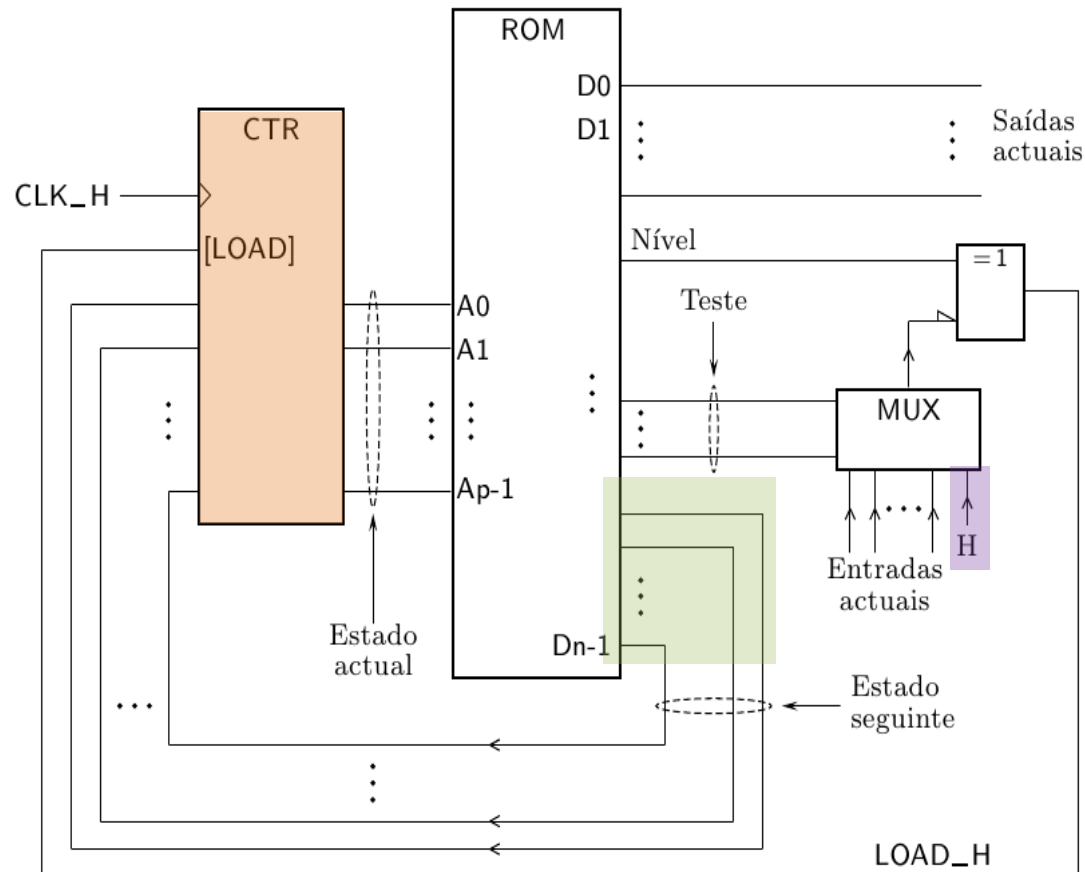
- Saltar para um outro estado qualquer, não imediatamente a seguir em termos de contagem



Não é necessário ter dois campos de estado seguinte (ES0 e ES1) na ROM

Palavras mais curtas

■ Controlo por ROM com Endereçamento Implícito



■ Controlo por ROM com Endereçamento Implícito

► Modos de contagem:

[INC] - O estado seguinte corresponde ao valor seguinte da contagem:

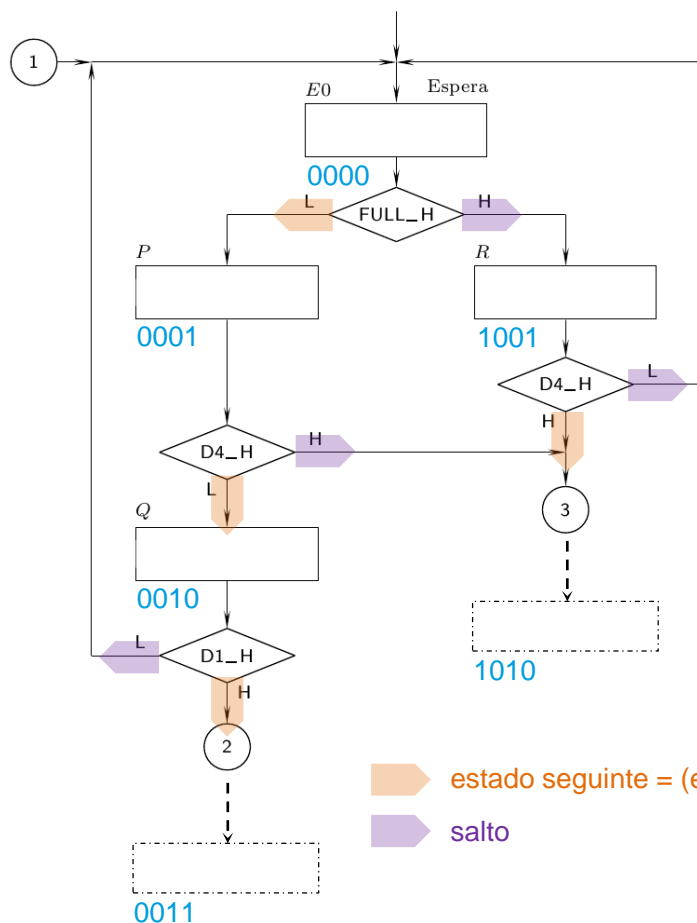
$$\text{Estado Seguinte} = \text{Estado Actual} + 1$$

[LOAD] - O estado seguinte corresponde a um estado que não corresponde ao estado seguinte de contagem:

$$\text{Estado Seguinte} = \text{Outro Estado} \quad (\text{salto})$$

- **Salto condicional** - depende do valor de uma variável de entrada:
 - campo **Teste** selecciona a variável de entrada;
 - campo **Nível** decide se o salto se deve efectuar quando ela tiver o valor 1 ou o valor 0;
- **Salto incondicional**:
 - Selecção da entrada H do MUX (sempre activa);
 - campo **Nível** a 1.

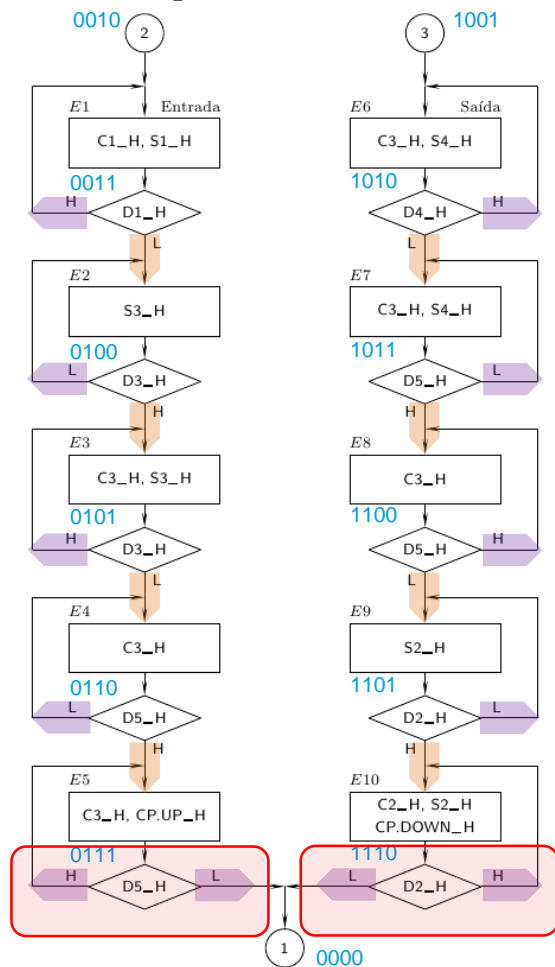
■ Exemplo: acesso a um parque de estacionamento



► Novos cuidados a ter na **codificação dos estados**:

- Sempre que possível: é necessário garantir que o estado seguinte a cada estado corresponde à codificação dada por:
estado seguinte = (estado actual + 1);
- Caso não seja possível: codificar com um **salto**

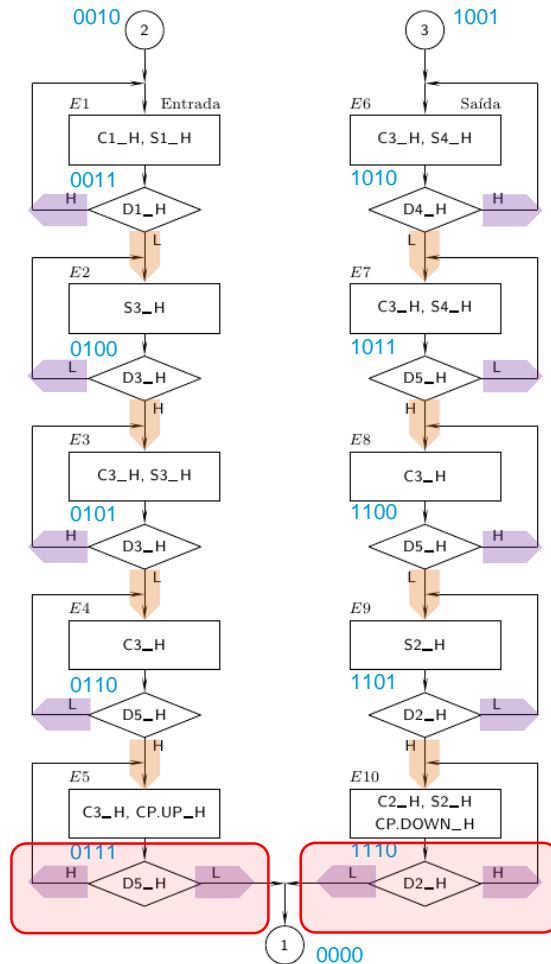
■ Exemplo: acesso a um parque de estacionamento



PROBLEMA:

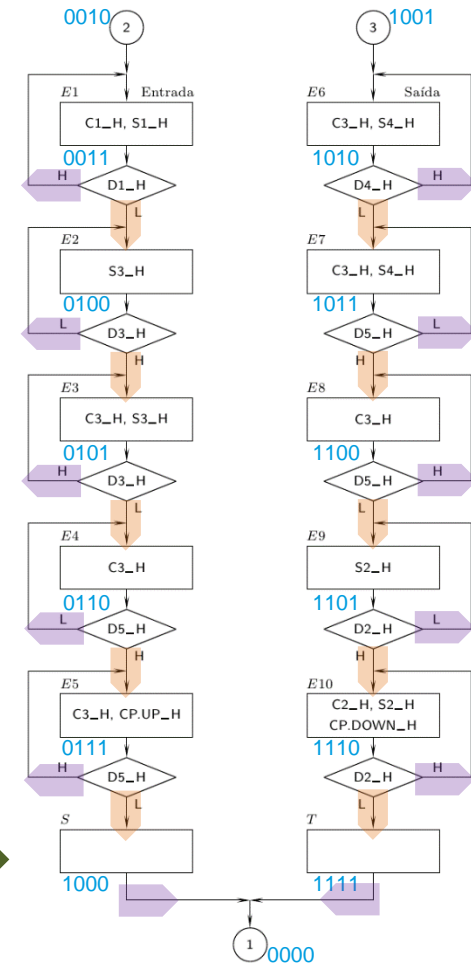
Os estados E5 e E10 transitam sempre através de um salto!!!

Exemplo: acesso a um parque de estacionamento



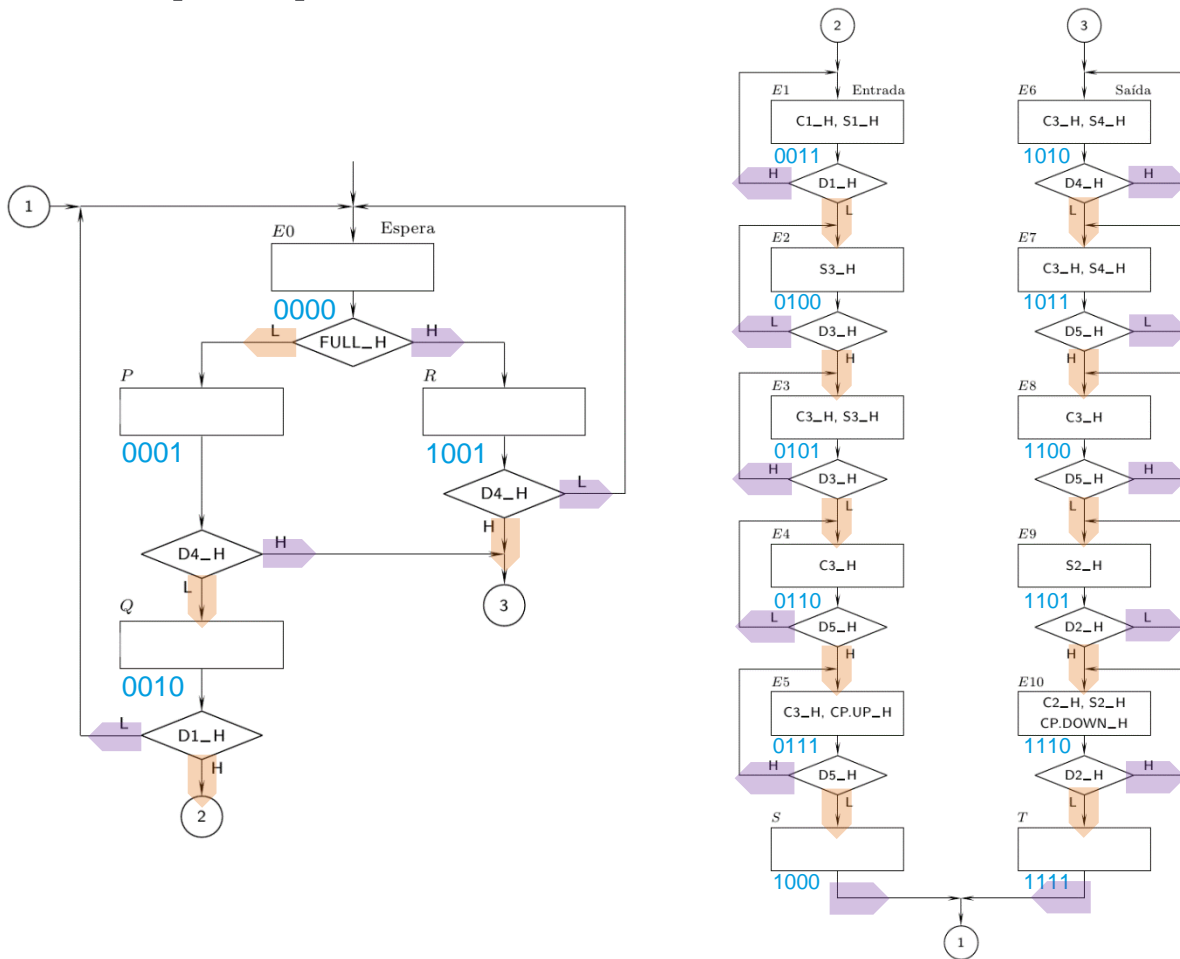
SOLUÇÃO:

Criar mais dois estados
“seguintes” extra: S e T



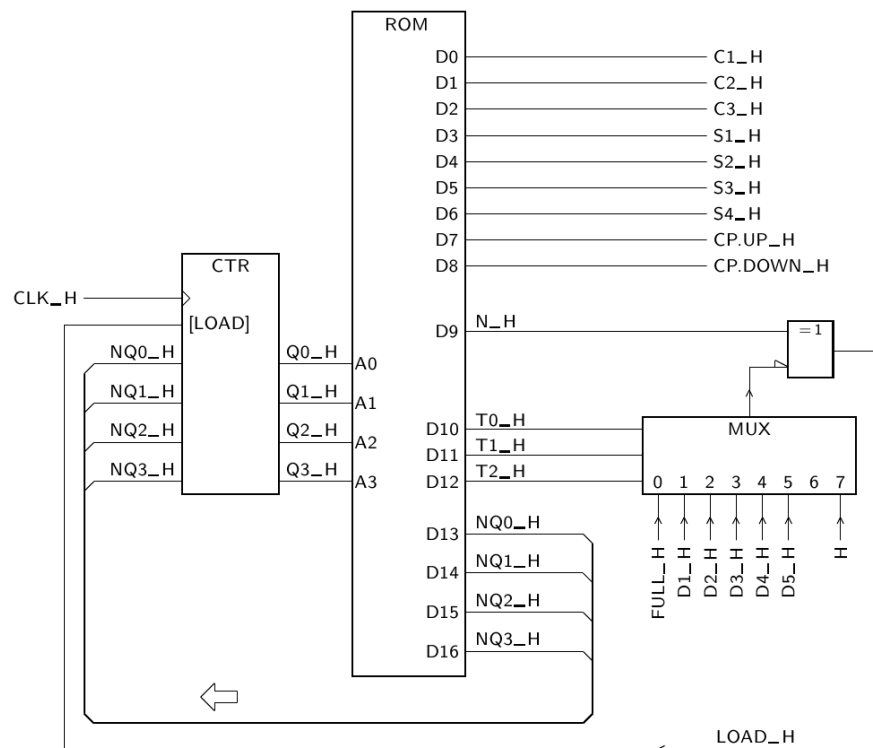
Exemplo: acesso a um parque de estacionamento

Estado	Q3	Q2	Q1	Q0
E0	0	0	0	0
P	0	0	0	1
Q	0	0	1	0
E1	0	0	1	1
E2	0	1	0	0
E3	0	1	0	1
E4	0	1	1	0
E5	0	1	1	1
S	1	0	0	0
R	1	0	0	1
E6	1	0	1	0
E7	1	0	1	1
E8	1	1	0	0
E9	1	1	0	1
E10	1	1	1	0
T	1	1	1	1



■ Exemplo: acesso a um parque de estacionamento

Diagrama de blocos:



Palavra da ROM:

16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NQ3	NQ2	NQ1	NQ0	T2	T1	T0	N	CP.DOWN	CP.UP	S4	S3	S2	S1	C3	C2	C1
ES				Teste			Nível	Saídas actuais								

■ Diagrama de blocos da máquina de estados

Tabela de Transição de Estados

Estado					ES				Teste			Nível	Saídas atuais											
	Q3	Q2	Q1	Q0	NQ3	NQ2	NQ1	NQ0	T2	T1	T0	N	CP.DOWN	CP.UP	S4	S3	S2	S1	C3	C2	C1			
	A3	A2	A1	A0	D16	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0			
E0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0			
P	0	0	0	1	1	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0			
Q	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0			
E1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0	1	0	1	1	0	1			
E2	0	1	0	0	0	1	0	0	0	1	1	0	0	0	0	1	0	0	1	0	0			
E3	0	1	0	1	0	1	0	1	0	1	1	1	0	0	0	1	0	0	1	0	0			
E4	0	1	1	0	0	1	1	0	1	0	1	0	0	0	0	0	0	0	1	0	0			
E5	0	1	1	1	0	1	1	1	1	0	1	1	0	1	0	0	0	0	1	0	0			
S	1	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0			
R	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0			
E6	1	0	1	0	1	0	1	0	1	0	0	1	0	0	1	0	0	0	1	0	0			
E7	1	0	1	1	1	0	1	1	1	0	1	0	0	0	1	0	0	0	1	0	0			
E8	1	1	0	0	1	1	0	0	1	0	1	1	0	0	0	0	0	0	1	0	0			
E9	1	1	0	1	1	1	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0			
E10	1	1	1	0	1	1	1	0	0	1	0	1	1	0	0	0	1	0	0	1	0			
T	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0			

Dimensão da ROM: 16 endereços de 17 bits (272 bits)

■ Tema da Próxima Aula:

- ▶ Circuitos de Controlo, Transferência e Processamento de Dados
- ▶ Exemplo de uma arquitectura simples de um processador

Agradecimentos

Algumas páginas desta apresentação resultam da compilação de várias contribuições produzidas por:

- Nuno Roma
- Guilherme Arroz
- Horácio Neto
- Nuno Horta
- Pedro Tomás