

# Experimento 10

## Máquina de Estados

Matheus Cardoso de Souza, 202033507  
Ualiton Ventura da Silva, 202033580  
Grupo G42

<sup>1</sup>Dep. Ciência da Computação – Universidade de Brasília (UnB)  
CIC0231 - Laboratório de Circuitos Lógicos

matheus-cardoso.mc@aluno.unb.br, 202033580@aluno.unb.br

**Abstract.** *The present report aims to develop a state machine to describe the behavior of traffic lights in cars and pedestrians.*

**Resumo.** *O presente relatório tem como objetivo a elaboração de uma máquina de estados para descrever o comportamento de semáforos de carros e pedestres.*

### 1. Introdução

Uma máquina de estados do ponto de vista de circuitos lógicos descreve um circuito sequencial em que o mesmo assume um estado específico em determinados momentos e que produz determinadas saídas e diante de situações específicas, sendo que para a composição de um estado é necessário a utilização de seus estados anteriores.[Referencia]

Existem diversos modelos de máquinas de estados que podem ser utilizadas, uma delas muito conhecidas é a de Moore, na qual para a composição de uma saída utiliza-se de seu estado atual. [Mandelli 2021]

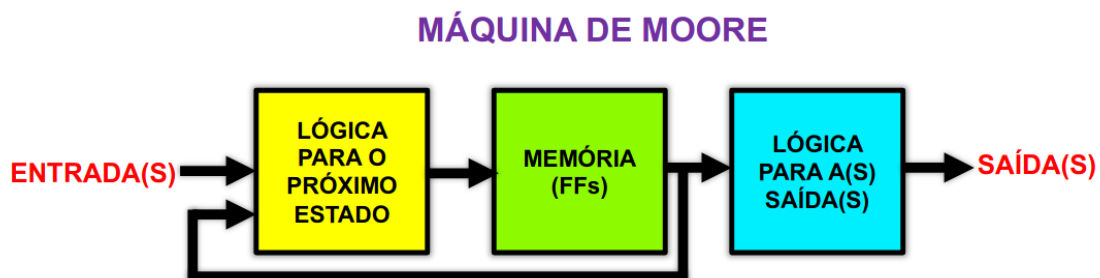


Figura 1. Máquina de Moore - Madelli 2021

Alternativamente também há outro modelo, chamado de máquina de estados de Mealy, na qual para a composição de sua saída utiliza-se não somente o seu estado atual como também a entrada que obtém do estado anterior. [Mandelli 2021]

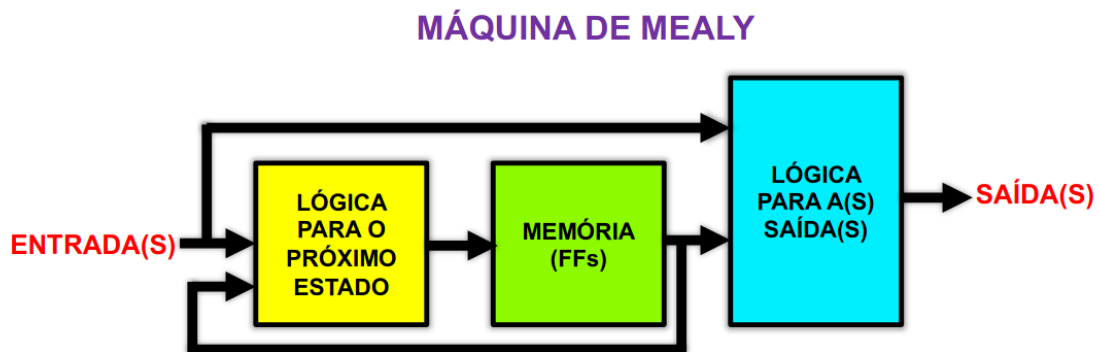


Figura 2. Maquina de Mealy - Mandelli 2021

### 1.1. Objetivos

Utilizando um dos conceitos de máquina de estados devemos descrever o comportamento de um semáforo utilizado para pedestres e carros.

### 1.2. Materiais

Em função da natureza do ensino a distância, os presentes experimentos não foram realizados usando-se materiais e equipamentos físicos, mas sim emulados por meio do software Deeds.

A seguir estão enumerados os materiais utilizados:

- Software Deeds
- Portas lógicas
  - *Flip-flops Ds*
  - *Display de saída de 8 Segmentos*
  - *Display de saída de 1 bit*
  - *Memória ROM de 64x16*
  - *Decodificador 2x4*
- *Clocks*

## 2. Procedimentos

Passaremos a apresentar os experimentos requeridos.

### 2.1. Semáforo de Carros e Pedestres

Para descrever o diagrama de estados podemos realizar a seguinte análise:

O semáforo dos carros é composto por 3 lâmpadas, portanto, uma forma de descrever os seus possíveis estados seria através de 2 bits, podendo então definir-se:

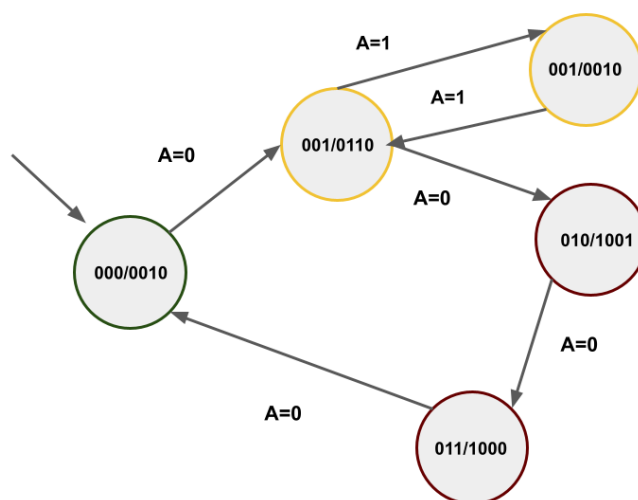
Lâmpada	Notação
Verde	00
Amarelo	01
Vermelho	10
Indefinido(Ou Apagado)	11

Analisando cada um dos semáforos que compõe o de pedestre, temos que um é independente do outro, sendo assim, será a cada um deles associado 1 bit individualmente.

Assim, temos que para compor um estado poderá ser utilizado 2 bits referentes ao estado do semáforo dos carros e outros 2 para compor o dos pedestres. Ou seja, a informação de saída utilizará 4 bits, sendo denotado por  $b_3, b_2, b_1, b_0$ , de maneira que define-se os 2 primeiros para o semáforo de carros e o terceiro referente ao de pedestre vermelho e quarto e último para o pedestre vermelho.

Como há 5 estados descritos, temos então que para compor os mesmos são utilizados 3 bits, já que para 2 bits há 4 combinações possíveis e para 3 bits 8.

Para a composição final poderíamos dizer que o diagrama que melhor atende aos requisitos definidos poderia ser:



**Figura 3. Diagrama de Estados**

Um fator que foi desconsiderado neste diagrama, trata-se da existência do tempo, assim, como descrevemos o tempo utilizando somente a memória ROM apresentada?

Uma possível solução e que será adotada é a de criarmos sub-estados, ou seja, entre o estado 1 e 2 por exemplo, haverá estados intermediários que descrevem o estado atual um certo número de vezes. Assim, se quisermos descrever que 1 dura 3 segundos, podemos definir como as sequências 1, 1.1, 1.2, 2, sendo que os estados entre 1 e 2 tem como objetivo apenas manterem 1 por um certo período de tempo, e para que dure 1 segundo, cada estado deverá ser mantido por 1 segundo.

Utilizando esta análise devemos então não utilizar somente 3 bits para a descrição de estados, mas sim 5, analisando a tabela abaixo podemos verificar a motivação:

Estado	Notação
1	00000
1.1	00001
1.2	00010
1.3	00011
1.4	00100
1.5	00101
2	00110
3	01111
3.1	01000
3.2	01001
3.3	01010
4	01011
4.1	01100
4.2	01101
4.3	01110
4.4	00111
5	10000
5.1	10001

Portanto, adotou-se os 5 bits mais significativos para descrição dos estado e os outros 4 utilizados para suas saídas. Deve-se perceber que para a criação de um momento onde as lâmpadas acendem e apagam em um intervalo, basta variar os 4 bits de saída nos estados intermediários.

Assim, sua memória ROM será descrita por:

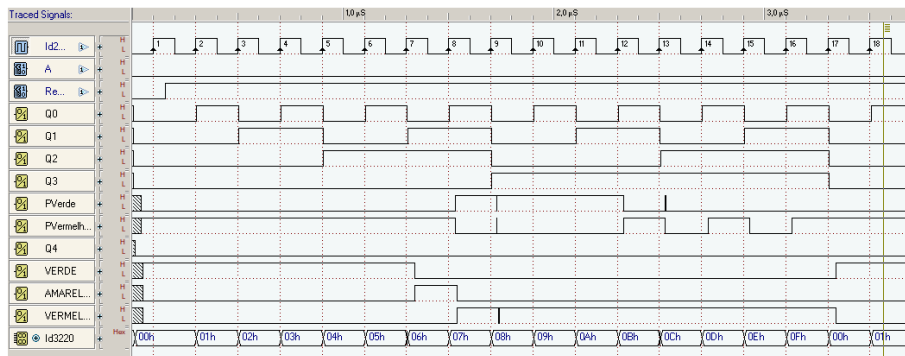
Addr	+0	+1	+2	+3
00	0000.1100.0111.1111	0001.0100.0011.1111	0001.1100.0111.1111	0010.0100.0111.1111
04	0010.1100.0111.1111	0011.0100.0111.1111	0011.1100.1111.1111	0100.0011.0111.1111
08	0100.1011.0111.1111	0101.0011.0111.1111	0101.1011.0111.1111	0110.0101.0111.1111
0C	0110.1001.0111.1111	0111.0101.0111.1111	0111.1001.0111.1111	0000.0101.0111.1111
10	1000.1000.1111.1111	1000.0001.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111
14	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111
18	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111
1C	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111
20	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111
24	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111
28	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111
2C	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111
30	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111
34	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111
38	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111
3C	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111	1111.1111.1111.1111

**Figura 4. Memória ROM**

Simulando temos: <https://youtu.be/XqisP-9VnYE>

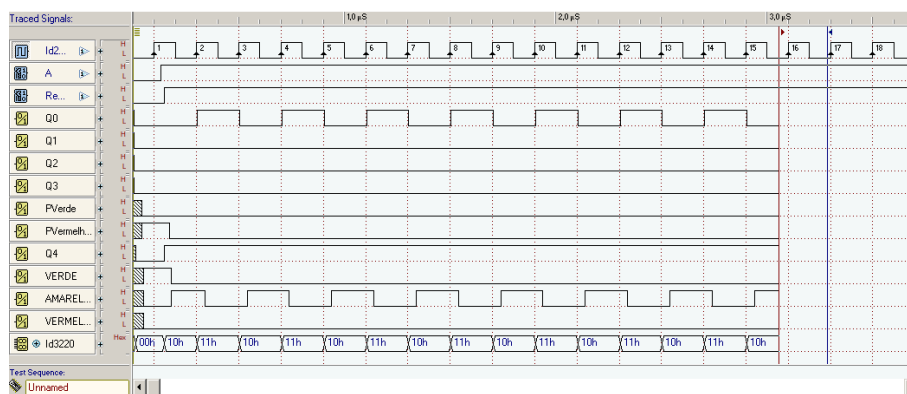
Sendo que em sua simulação de onda será:

Para A=0:



**Figura 5. Simulação de Onda para A=0**

Para A=1:



**Figura 6. Simulação de Onda para A=1**

### 3. Análise dos Resultados

Passaremos a analisar individualmente cada um dos tópicos anteriores, levantando observações pertinentes para cada um deles.

#### 3.1. Análise do tópico 2.1

Os resultados obtidos no tópico 2.1 são os de fato esperados e verificados sendo que alguns detalhes poderiam ser corrigidos por questões de otimização poderiam ser utilizados 3 bits para a composição dos estados do semáforo de carros por exemplo, assim, além de minimizar a complexidade retirando o uso de decodificadores permitiria que cada lâmpada pudesse ter um estado independente.

### 4. Conclusão

A utilização de máquinas de estados finitos possibilita a criação de comportamentos complexos e não necessariamente lineares, pois, apesar da dependência de estado e saída, estes não necessariamente precisam ser iguais.

Fator a ser considerado é o fato da memória ROM ser capaz de descrever variados estados de funções lógicas, porém, apesar do ganho em versatilidade há a perda em velocidade.

### **Referências**

[Mandelli 2021] Mandelli, M. G. (2021). Maquinas de estado. <https://aprender3.unb.br/mod/resource/view.php?id=386117>.

## Auto-Avaliação

Respostas:

Questão	Resposta
1	V
2	V
3	V
4	F
5	F
6	V
7	V
8	V
9	F
10	V