

# SEM0540

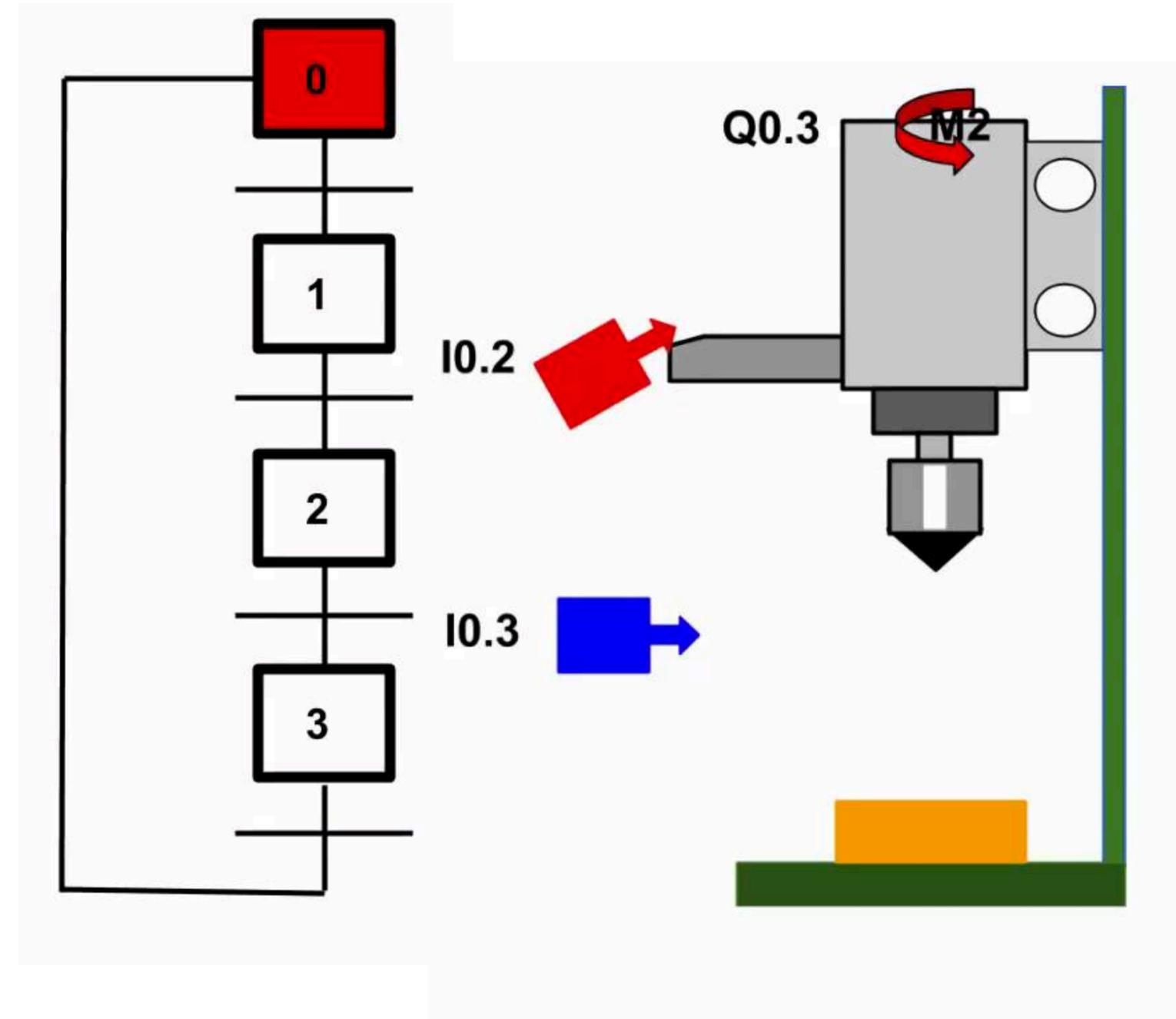
## Elementos de Automação

---

Aula #21: Máquina de Estados e SFC

**Prof. Dr. Thiago Boaventura**

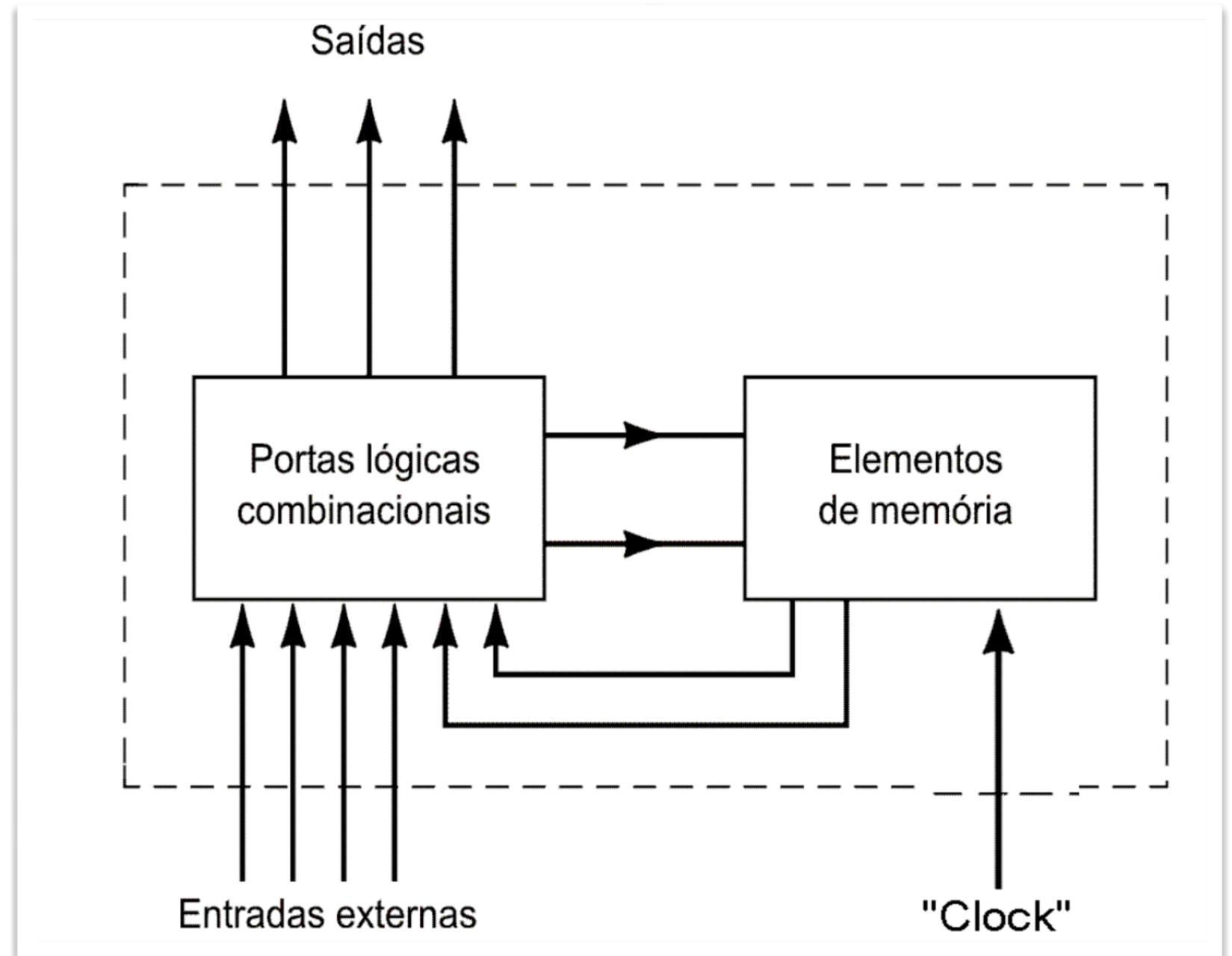
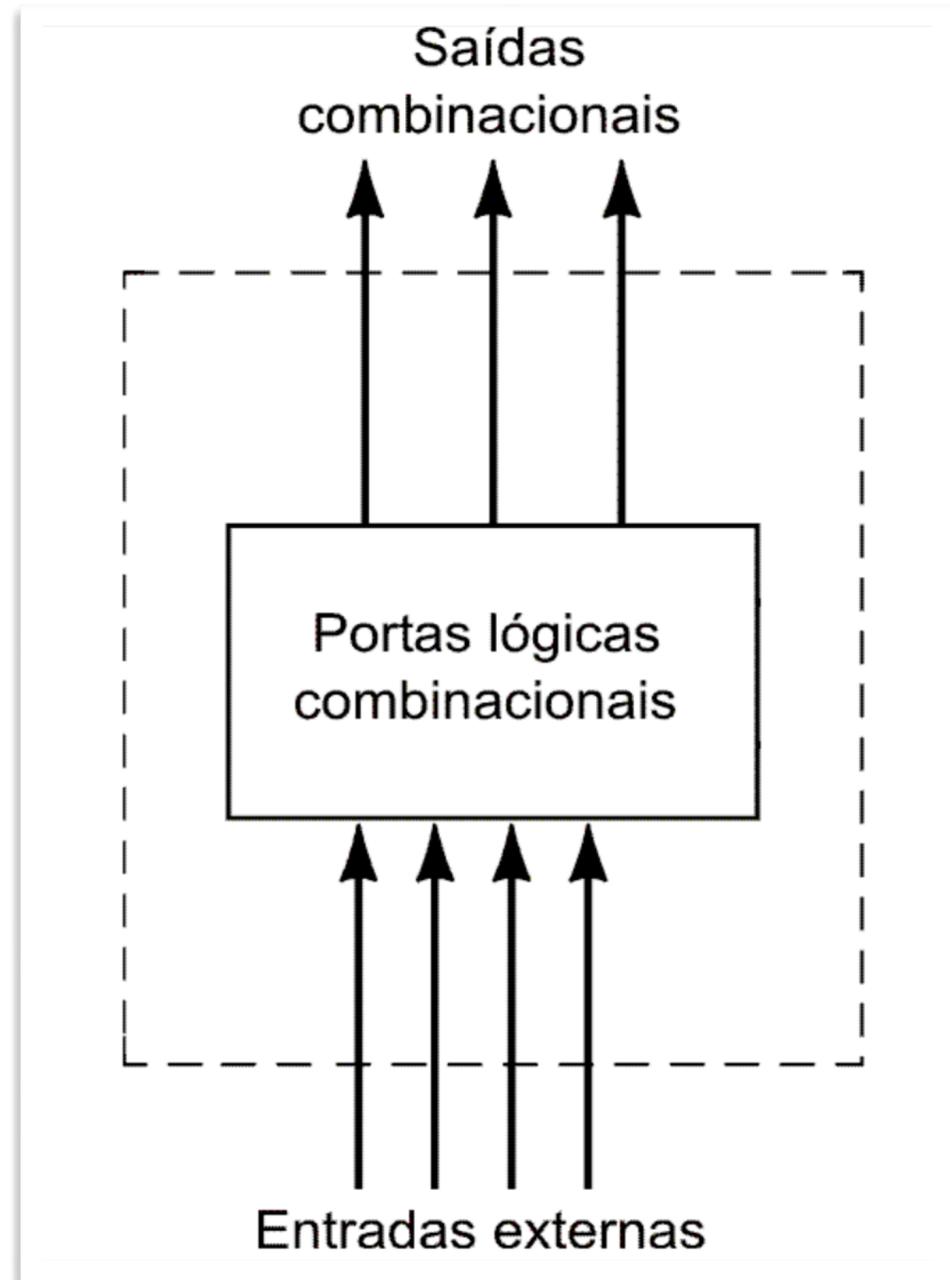
[tboaventura@usp.br](mailto:tboaventura@usp.br)



# Circuito combinacional vs. sequencial

Máquina de Estados Finitos

SFC



# Máquinas de estado finito

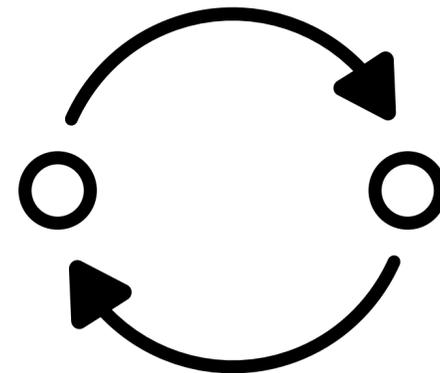
modelo de comportamento  
de um determinado processo

Máquina de  
Estados Finitos

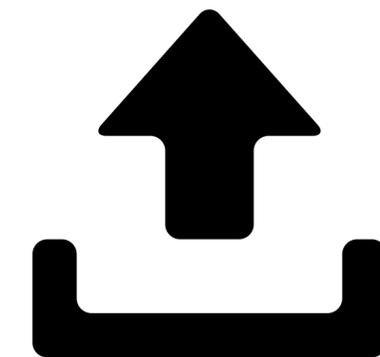
Estado



Transição



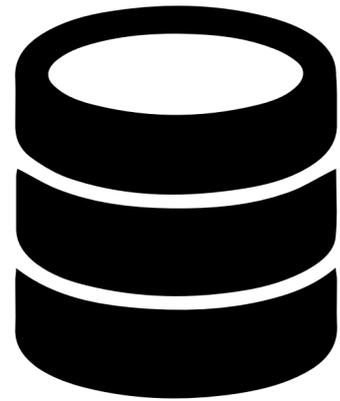
Saída



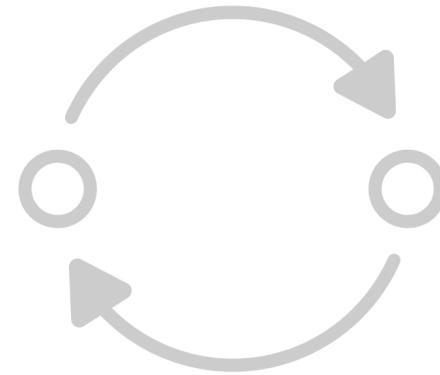
SFC

# Máquinas de estado finito

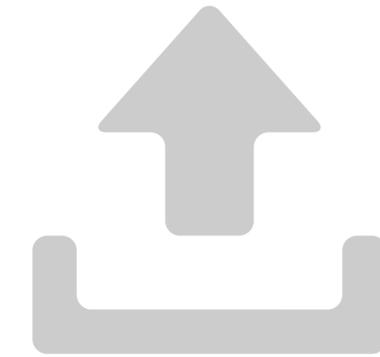
Estado



Transição



Saída



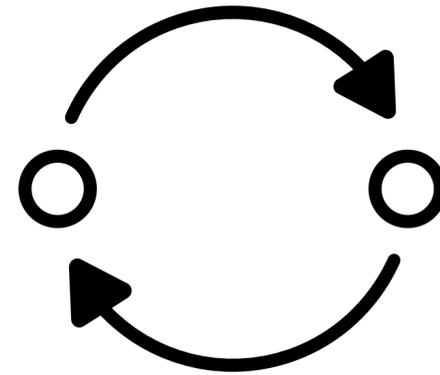
comporta-se como uma **memória**:  
armazena todas as informações sobre as  
saídas em um determinado momento

# Máquinas de estado finito

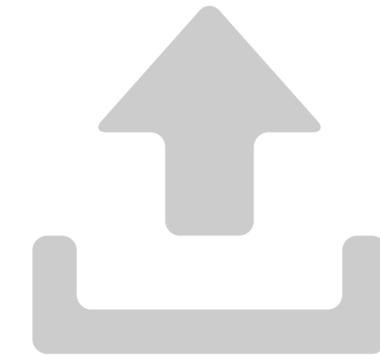
Estado



Transição



Saída



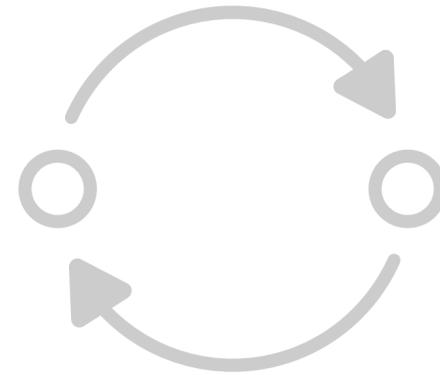
é a **condição** para que ocorra a **mudança** de um **estado** para outro

# Máquinas de estado finito

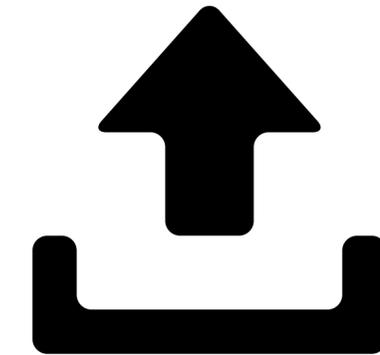
Estado



Transição



Saída



descreve a **atividade** que deve ser realizada num determinado estado

# Tipos de MEF

## Mealy Machine vs. Moore Machine

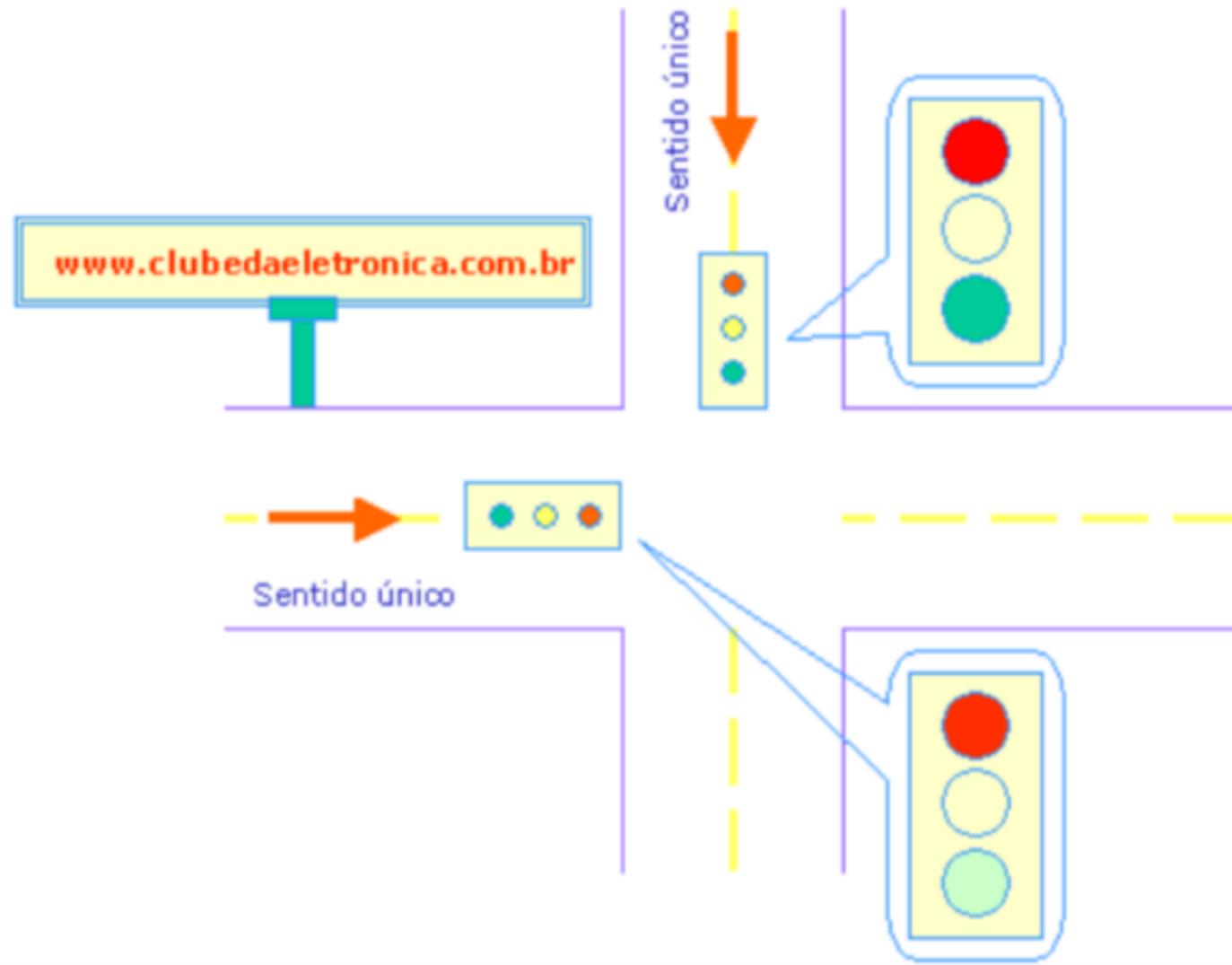
The following table highlights the points that differentiate a Mealy Machine from a Moore Machine.

Mealy Machine	Moore Machine
Output depends both upon the present state and the present input	Output depends only upon the present state.
Generally, it has fewer states than Moore Machine.	Generally, it has more states than Mealy Machine.
The value of the output function is a function of the transitions and the changes, when the input logic on the present state is done.	The value of the output function is a function of the current state and the changes at the clock edges, whenever state changes occur.
Mealy machines react faster to inputs. They generally react in the same clock cycle.	In Moore machines, more logic is required to decode the outputs resulting in more circuit delays. They generally react one clock cycle later.

# Exemplo — Controle de tráfego

Máquina de Estados Finitos

SFC



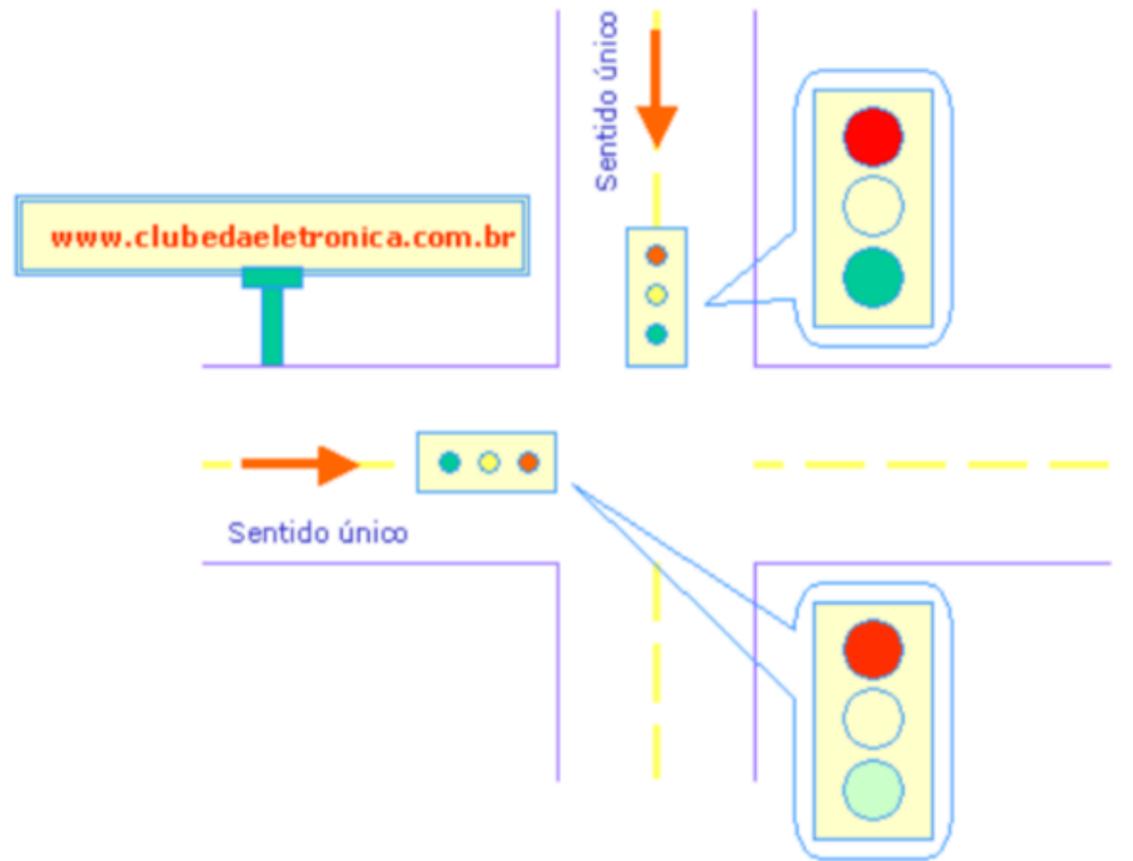
## Descrição do funcionamento:

O semáforo terá início quando um botão (BL) tipo “push button” for acionado, dar-se-à então início ao ciclo. O tempo (T) ficará a critério do operador e o sistema poderá ser desligado em qualquer momento através de um botão (BD), também push button.

# Exemplo — Entradas/Saídas

Máquina de Estados Finitos

SFC

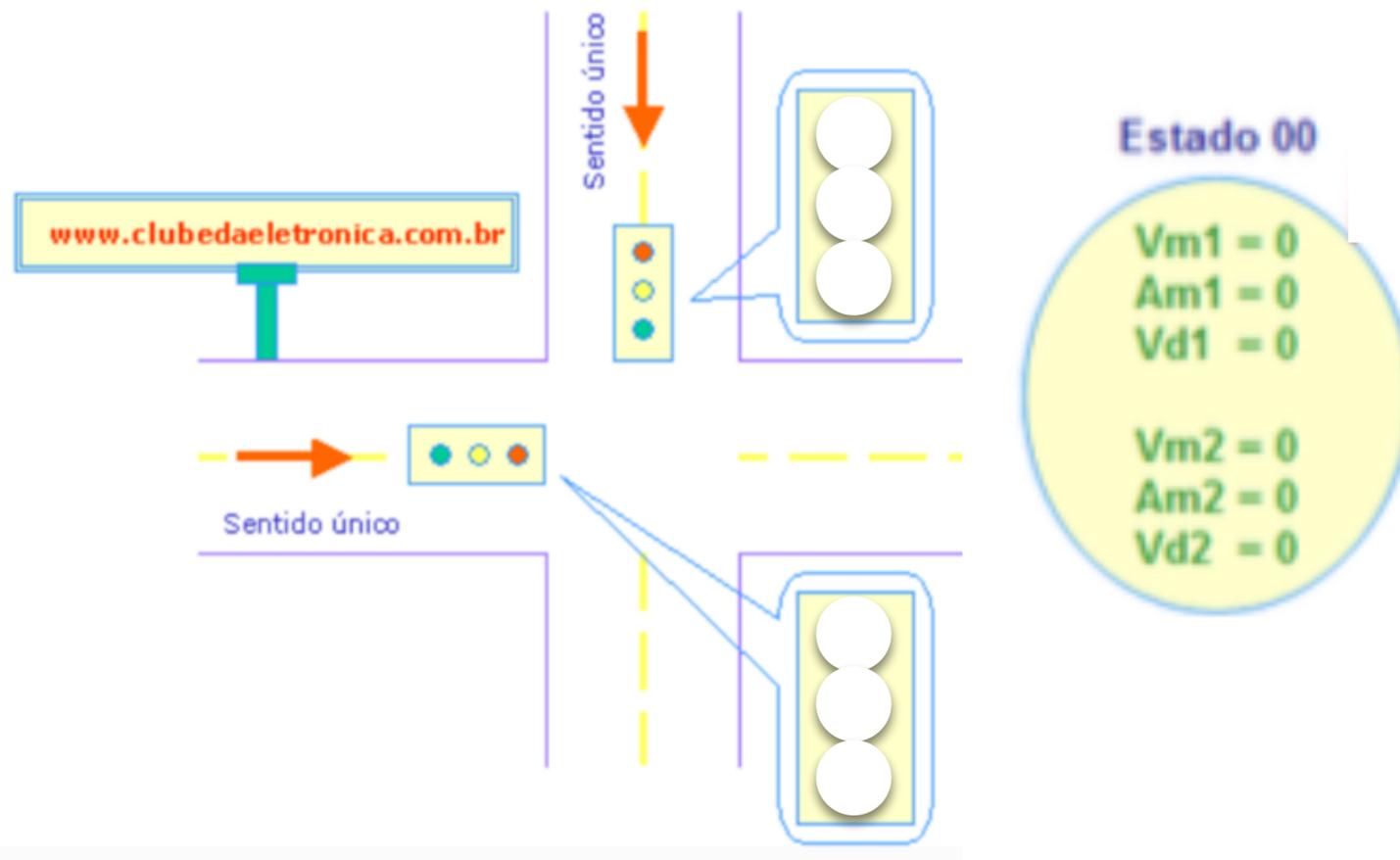


Entradas	Saídas Semáforo 1	Saídas Semáforo 2
Botão liga = I0	Vermelho 1 = O0	Vermelho 2 = O3
Botão desliga = I1	Amarelo 1 = O1	Amarelo 2 = O4
	Verde 1 = O2	Verde 2 = O5

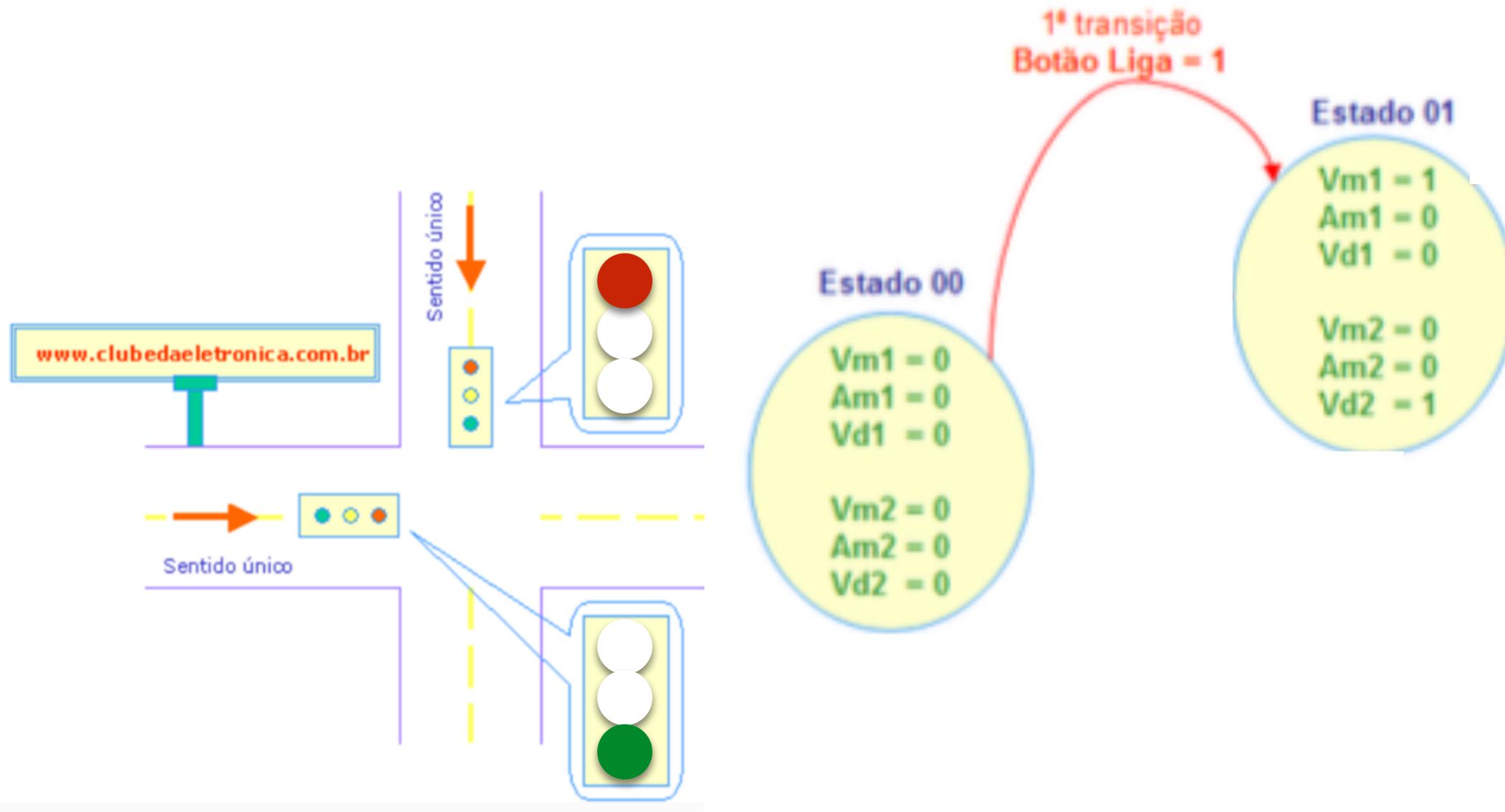
# Exemplo — Diagrama de estados

Máquina de Estados Finitos

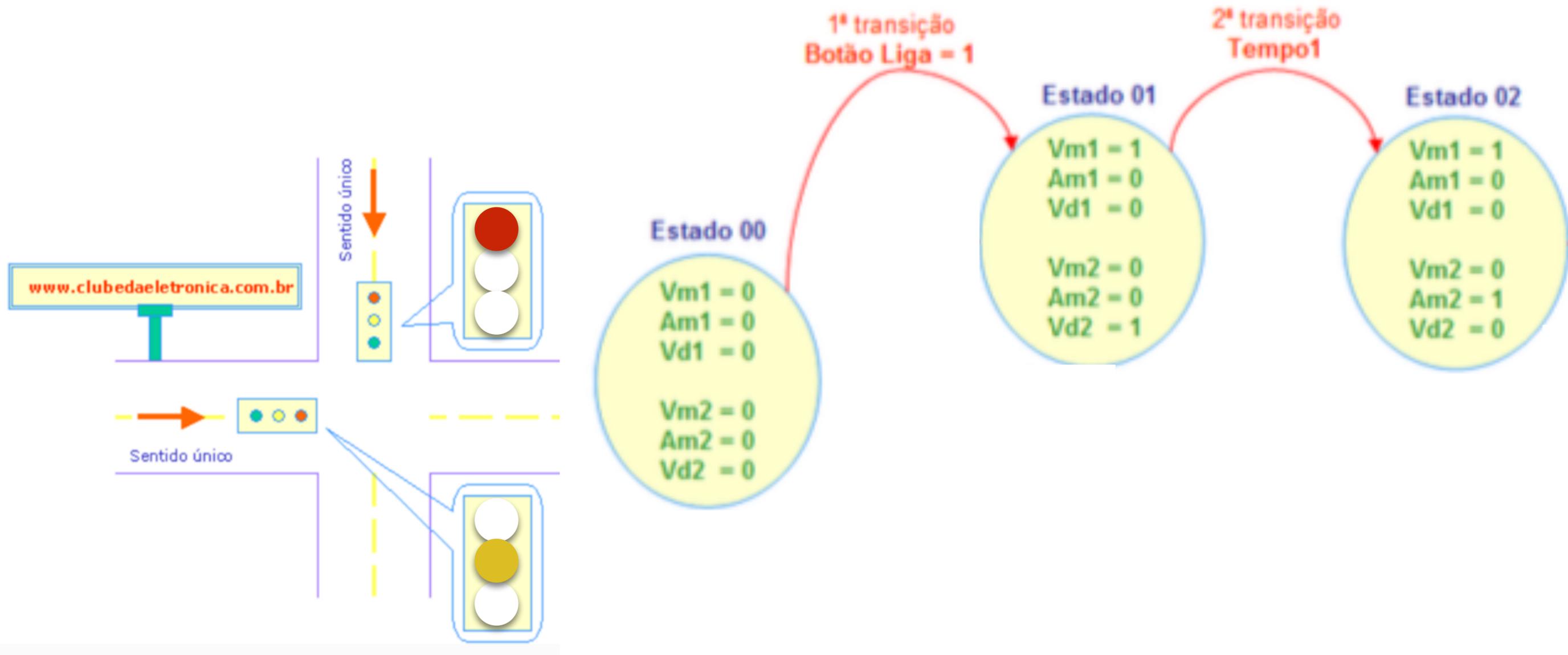
SFC



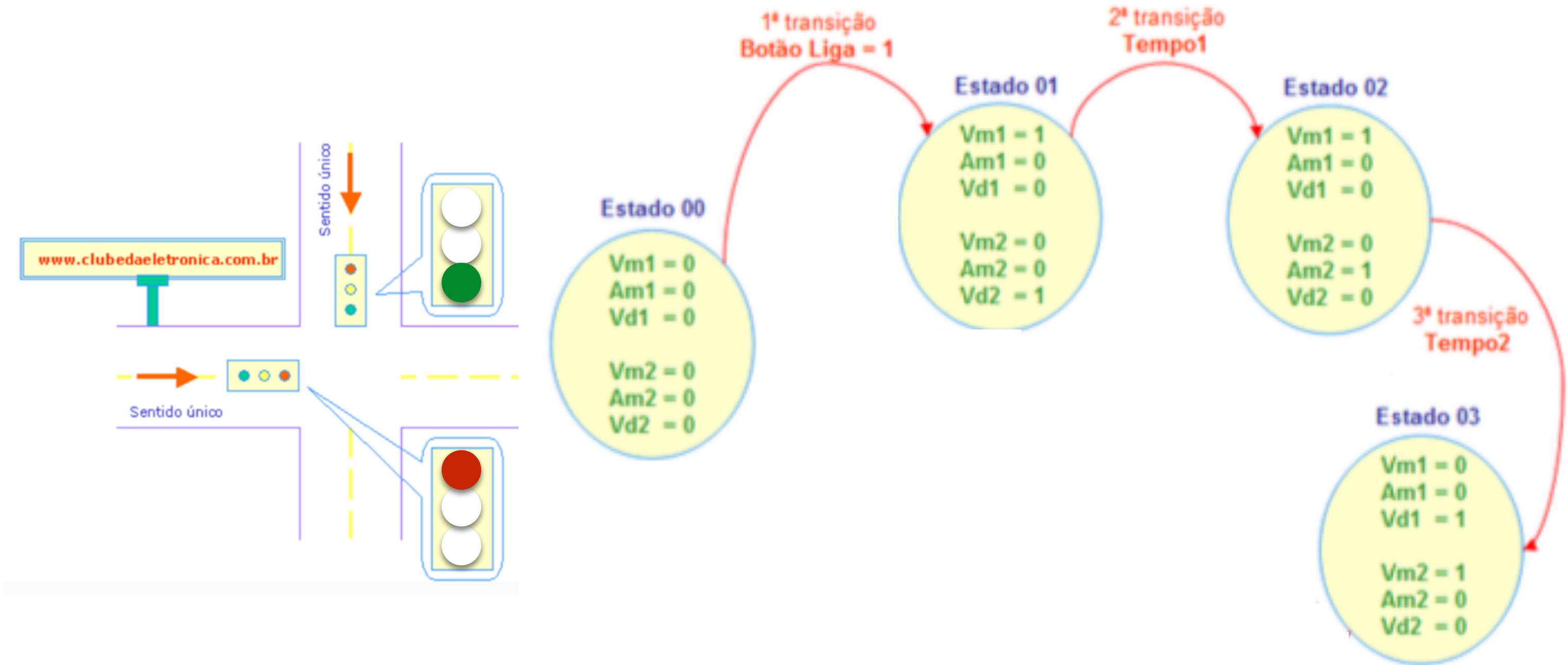
# Exemplo — Diagrama de estados



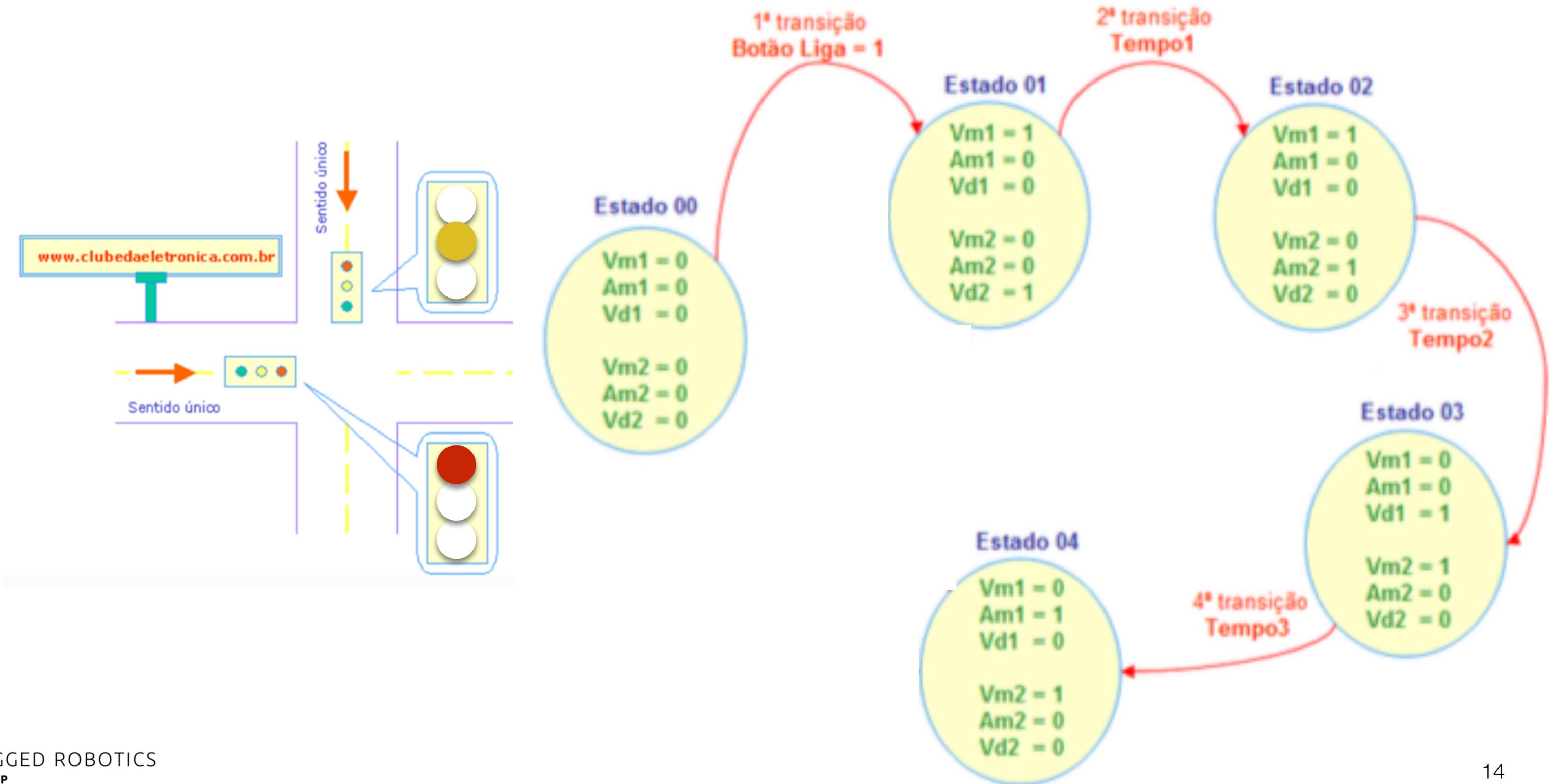
# Exemplo — Diagrama de estados



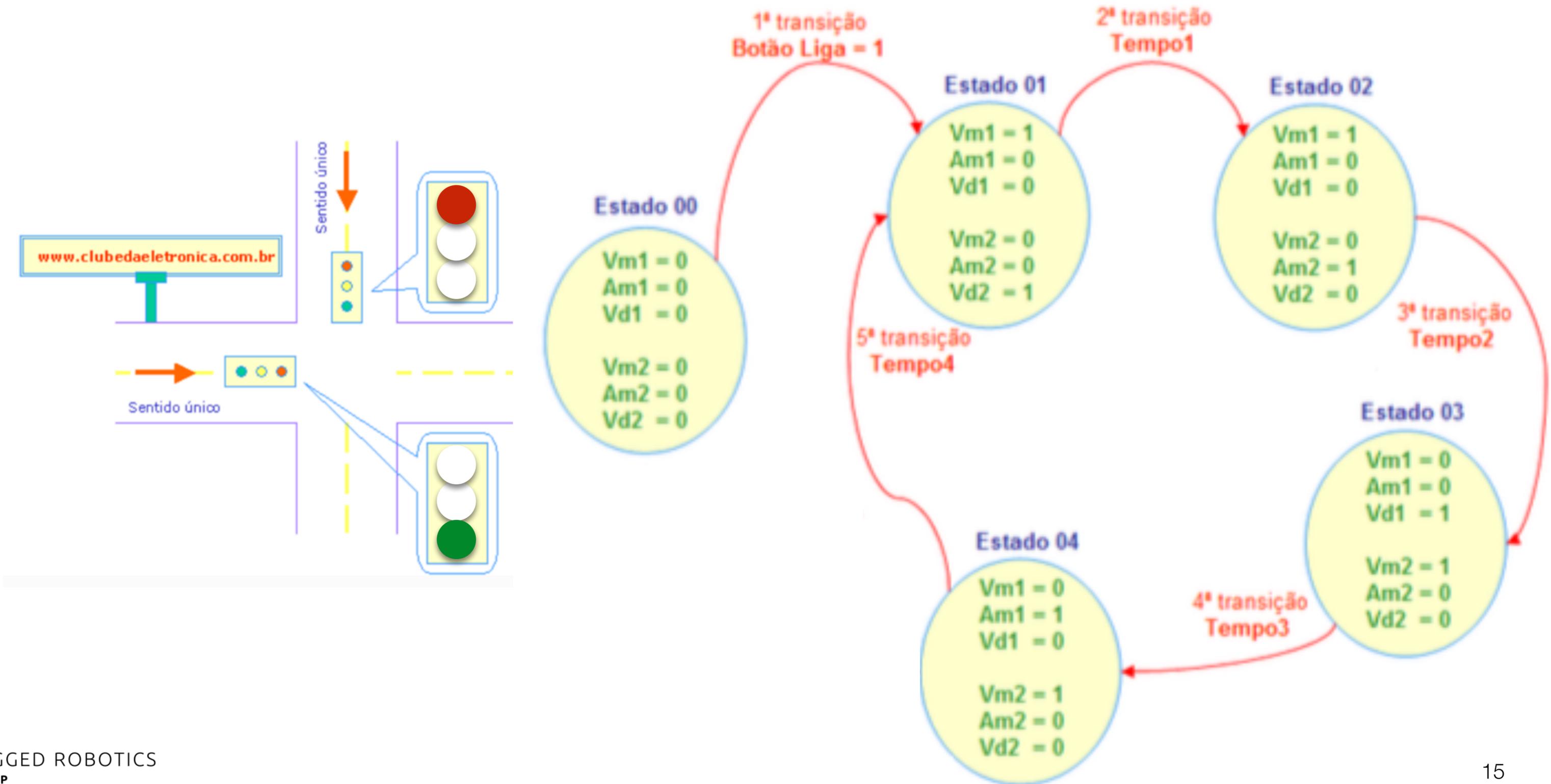
# Exemplo — Diagrama de estados



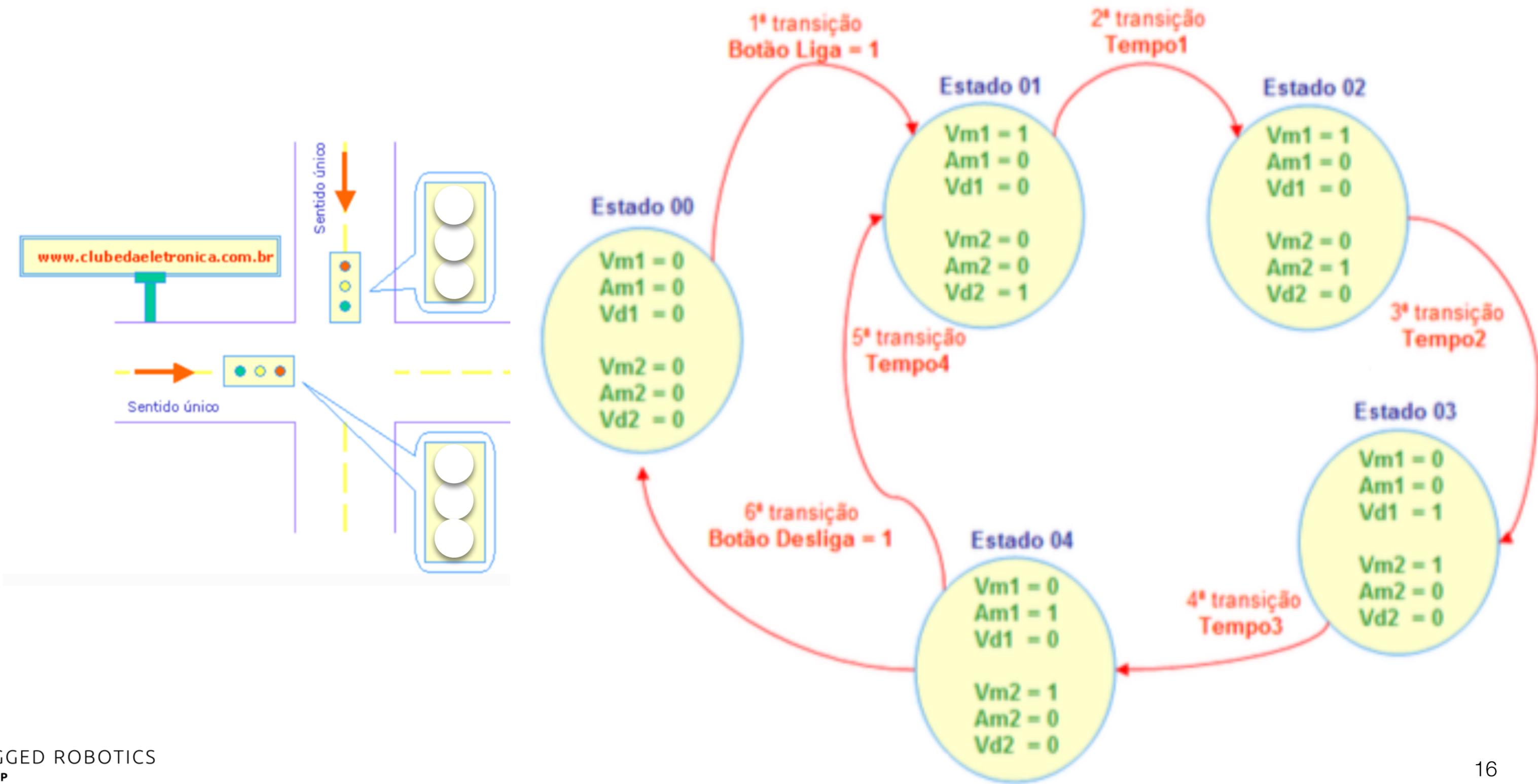
# Exemplo — Diagrama de estados



# Exemplo — Diagrama de estados



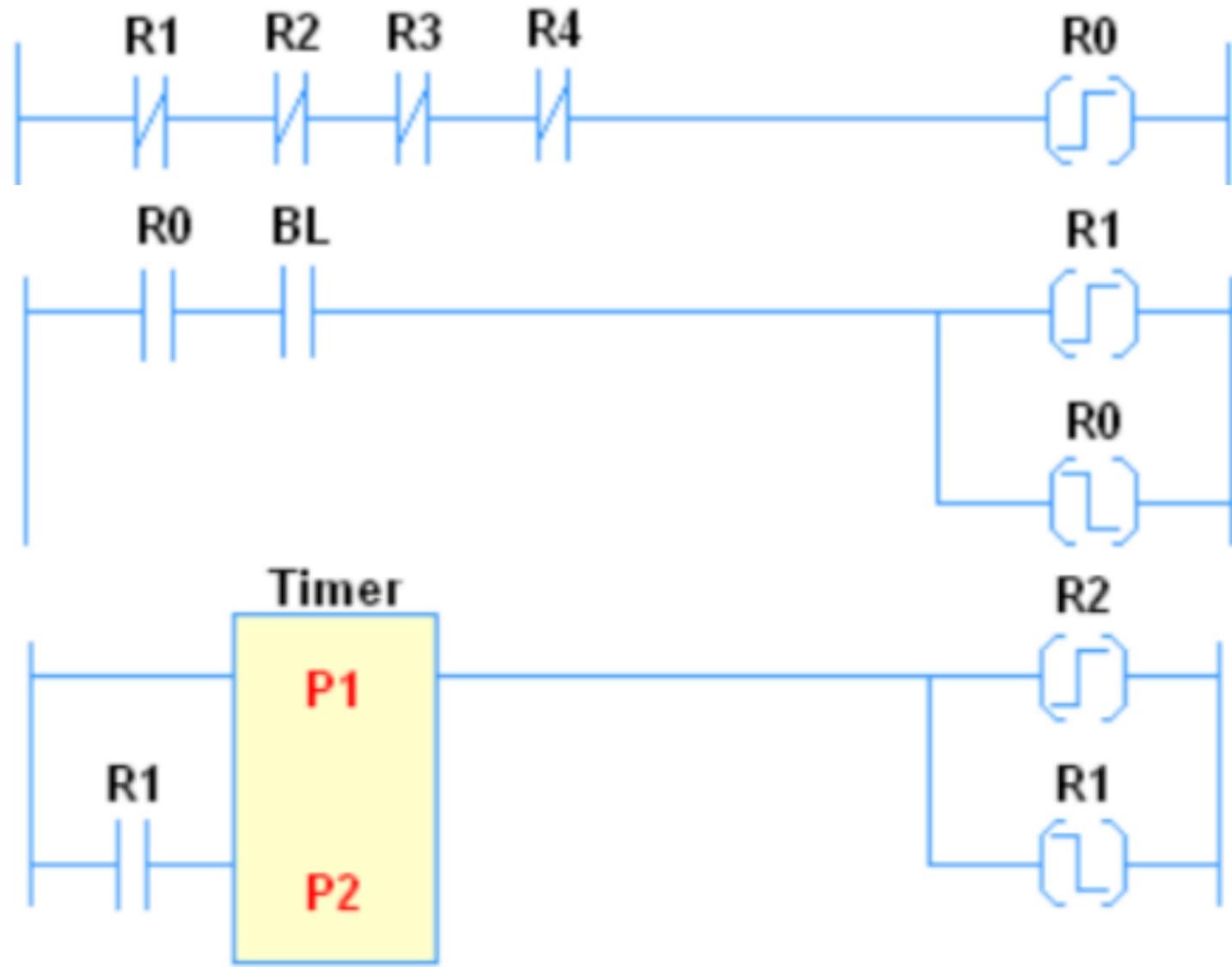
# Exemplo — Diagrama de estados



# Exemplo — Programa Ladder

Máquina de Estados Finitos

SFC

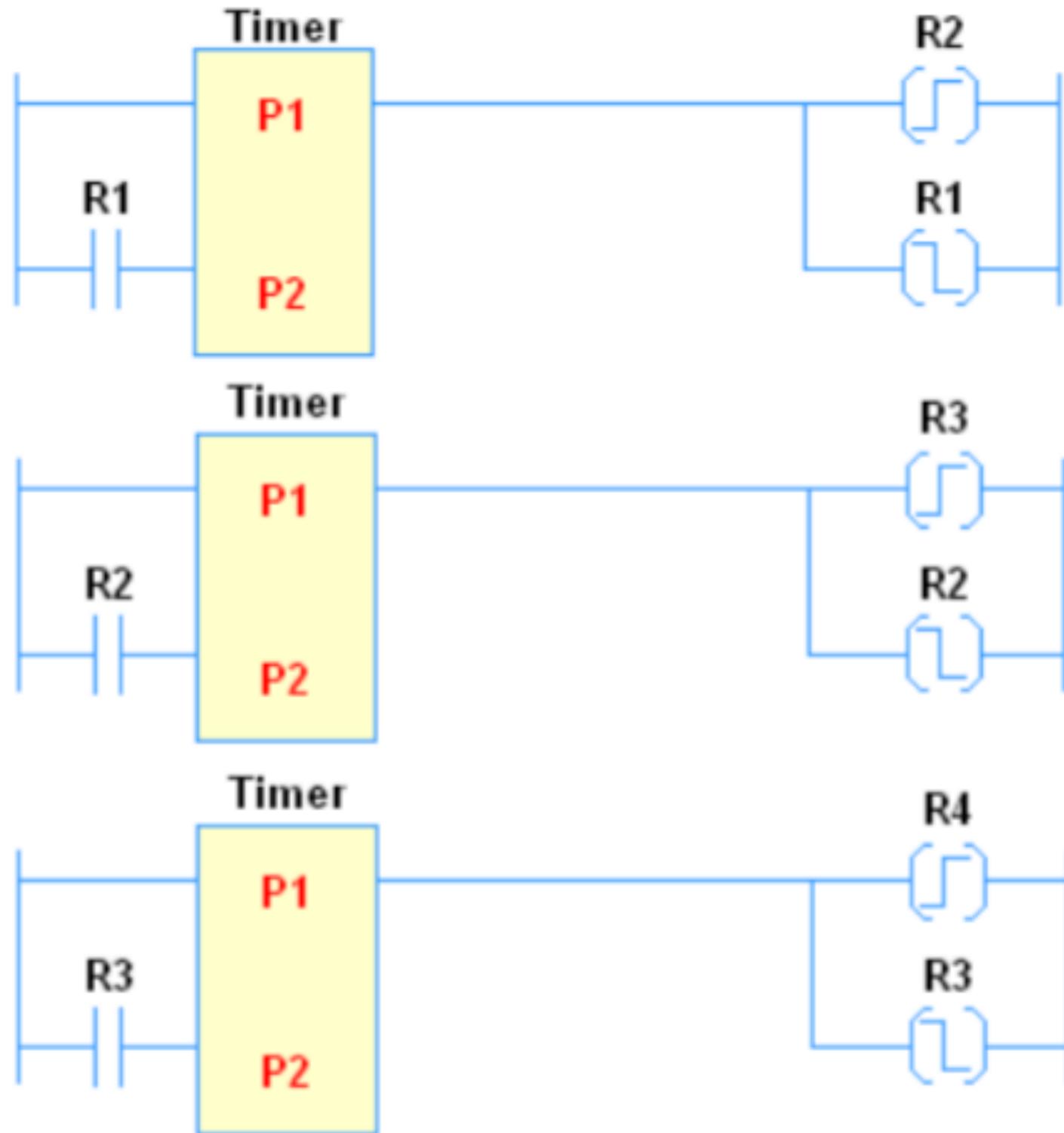


Cada estado de seu sistema deve ser associado a um contato auxiliar (R).

Se o estado inicial (R0) estiver acionado e se o Botão (push button) liga (BL) for pressionado é setado o estado (R1) e resetado o estado (R0).

Se o estado (R1) estiver setado dar-se-á início a contagem de tempo (T1) e ao final o estado (R2) será setado e (R1) ressetado.

# Exemplo — Programa Ladder

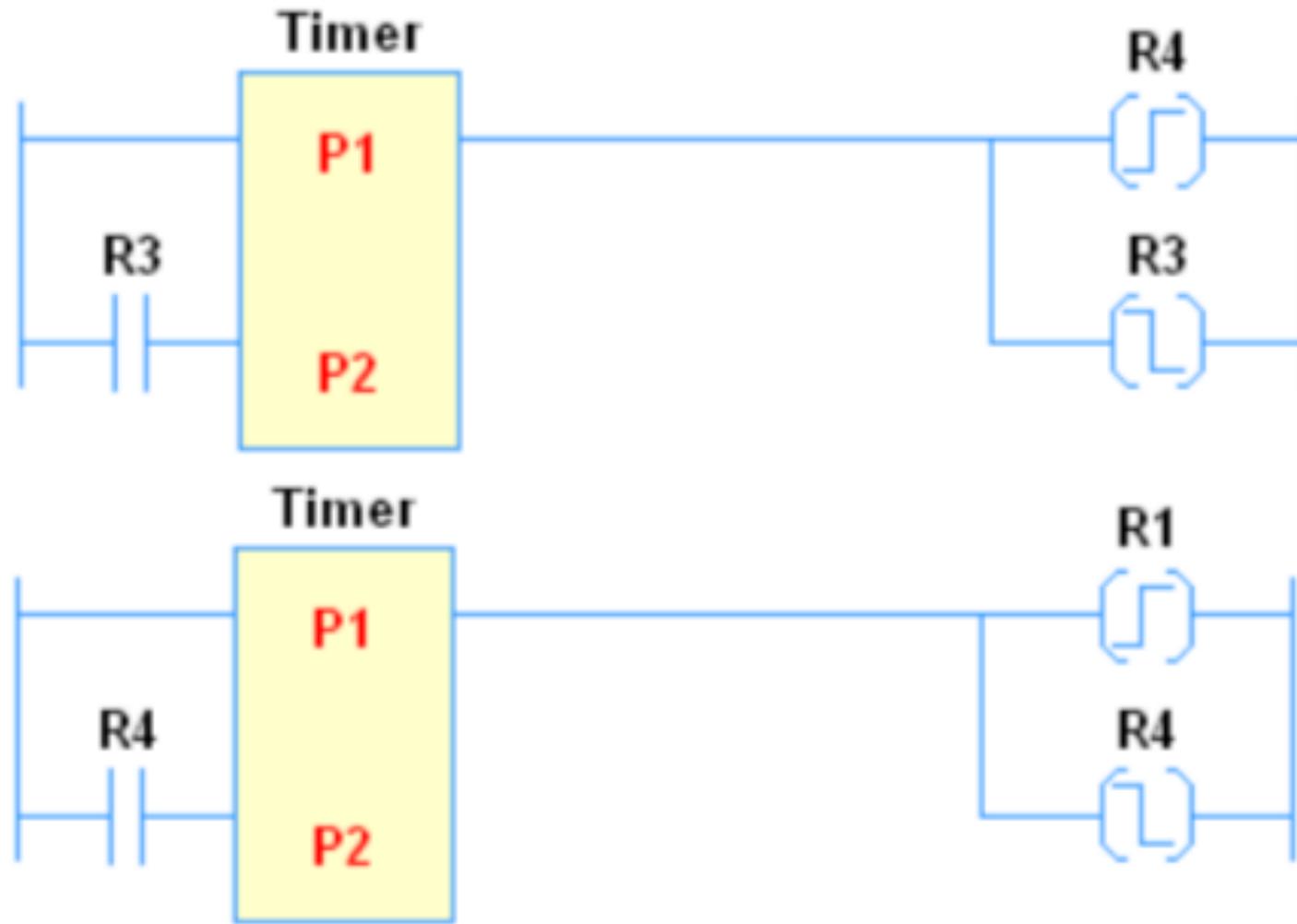


Se o estado (R1) estiver setado dar-se-á início a contagem de tempo (T1) e ao final o estado (R2) será setado e (R1) ressetado.

# Exemplo — Programa Ladder

Máquina de Estados Finitos

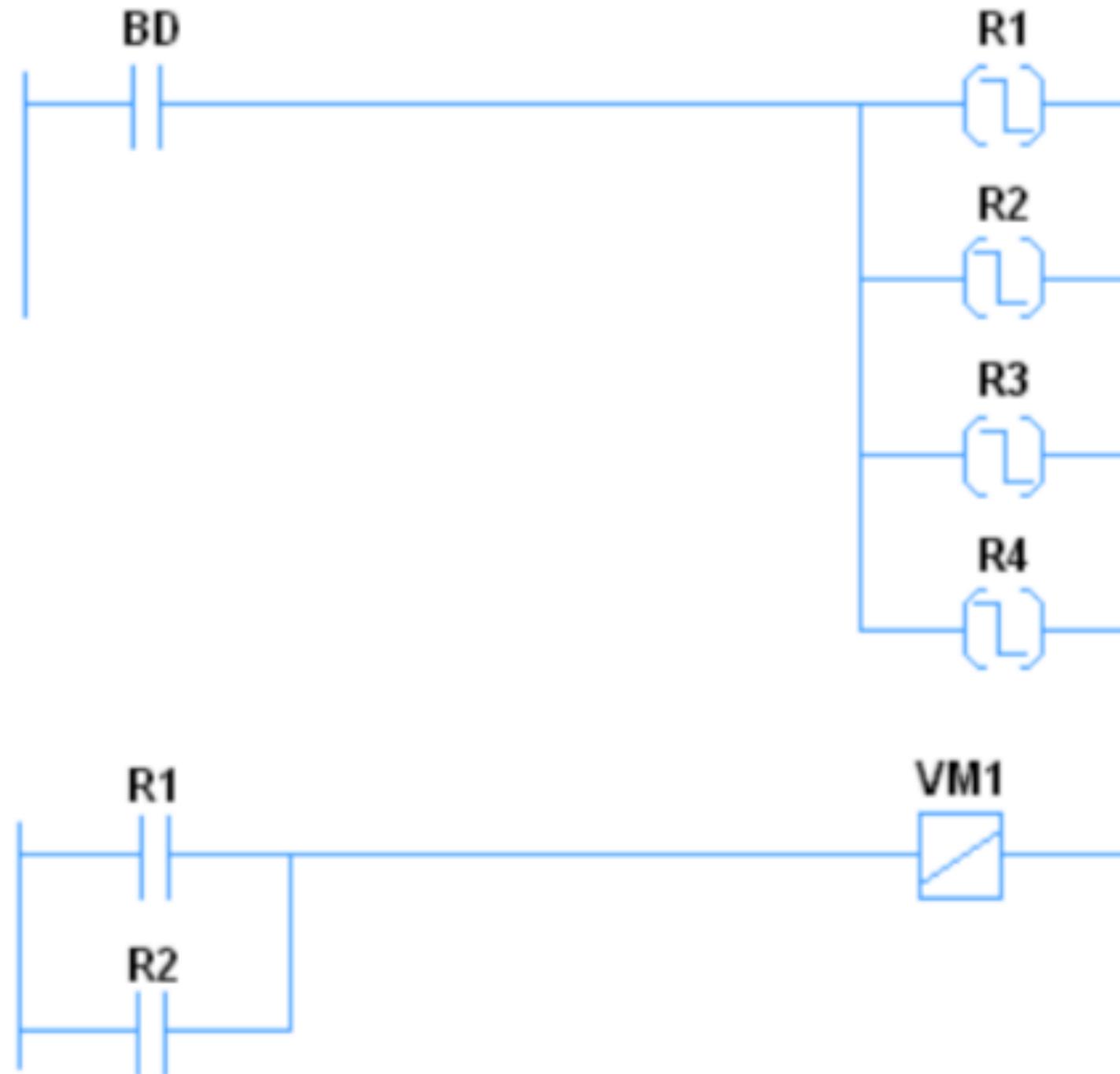
SFC



# Exemplo — Programa Ladder

Máquina de Estados Finitos

SFC



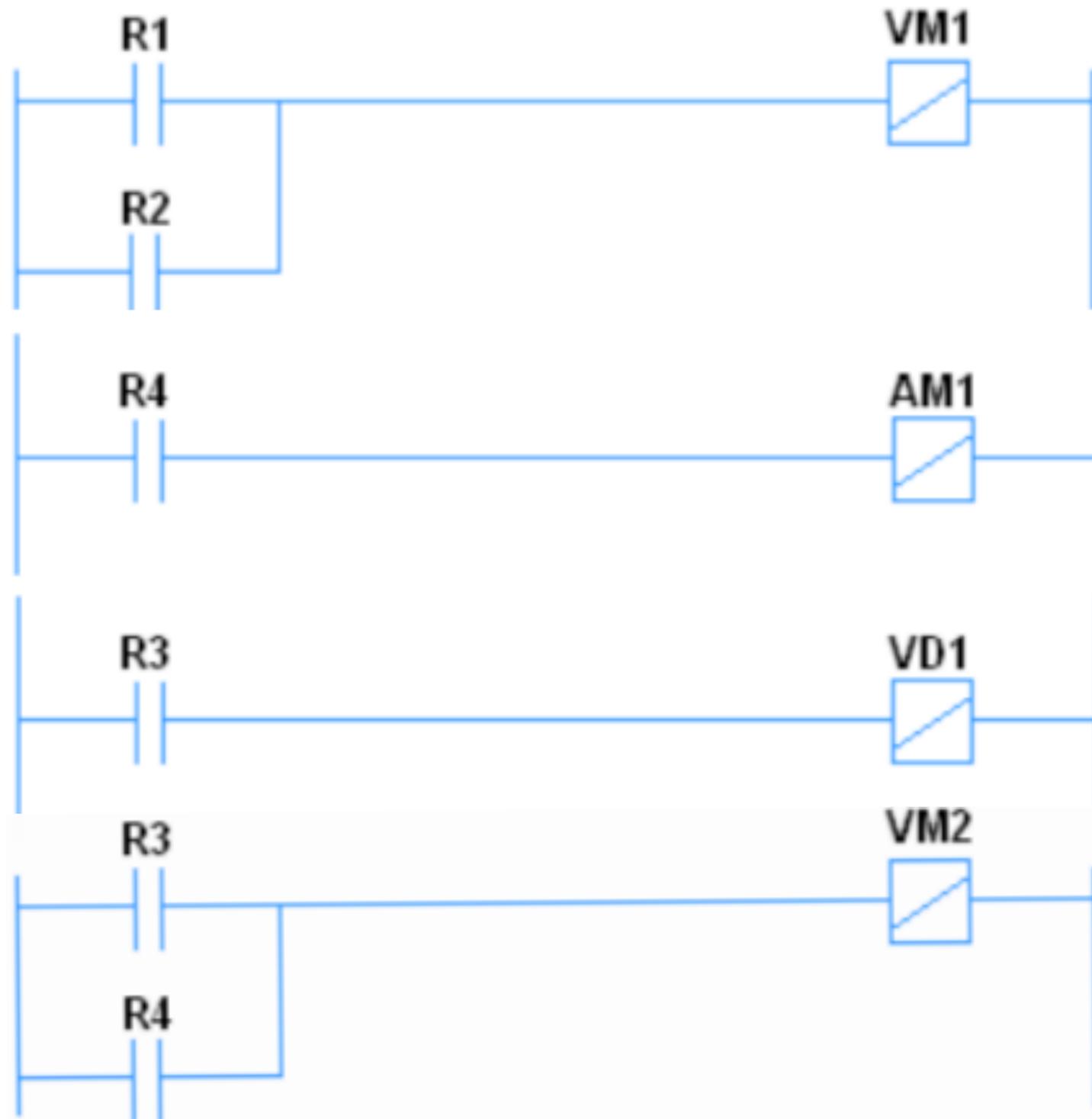
Se o botão de desligar (push button) for pressionado todos os estados (R1), (R2), (R3) e (R4) serão ressetados.

Acionando a saída vermelha do semáforo 01 (VM1)

# Exemplo — Programa Ladder

Máquina de Estados Finitos

SFC

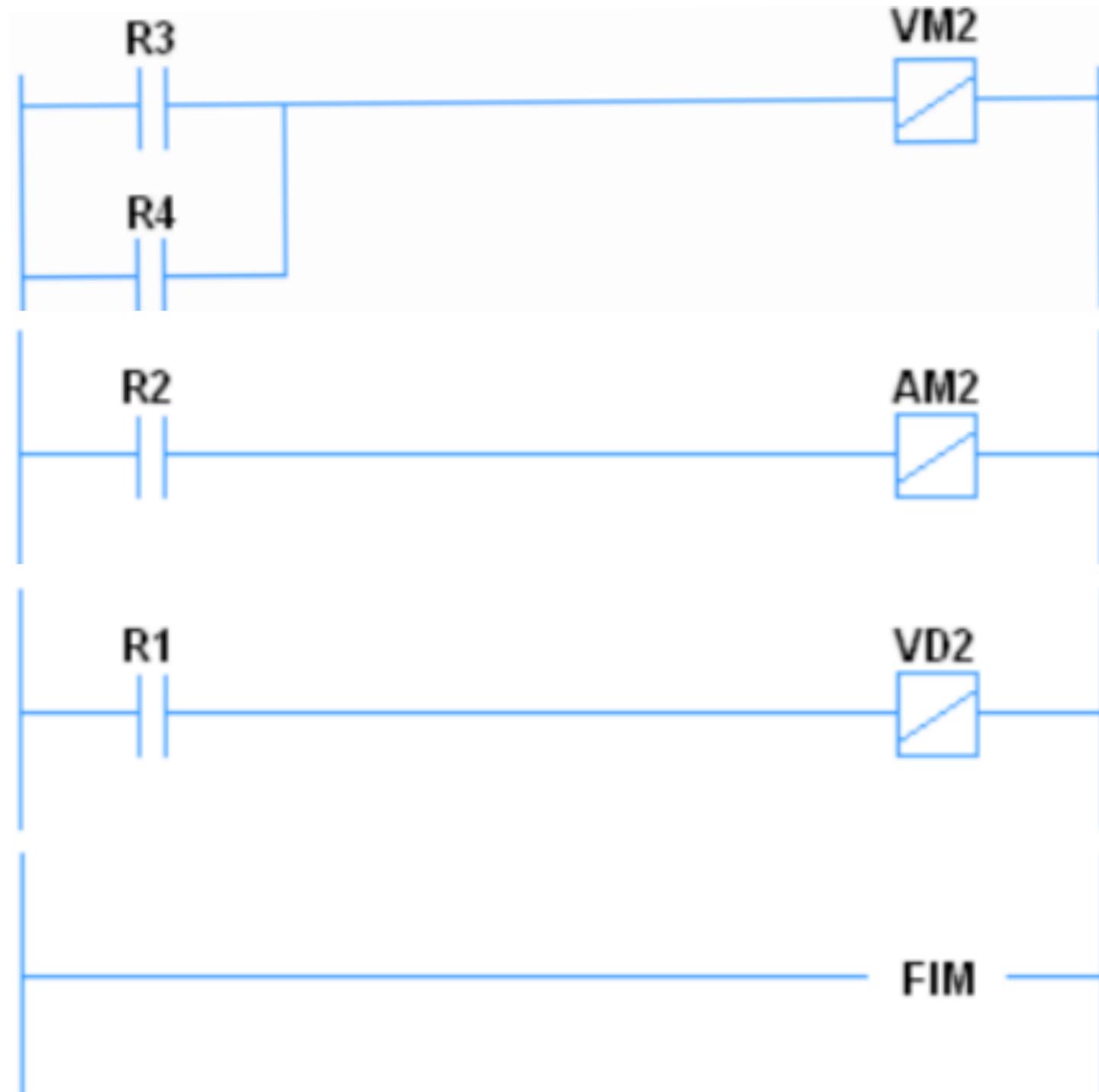


Acionando a saída vermelha do semáforo 01 (VM1)

# Exemplo — Programa Ladder

Máquina de Estados Finitos

SFC



# Conteúdo

---

Máquina de  
Estados Finitos



- Elementos principais
- Exemplo

SFC

# IEC 61131-3 — Linguagens de programação

Máquina de Estados Finitos



Textuais

Lista de instruções (IL)  
Texto Estruturado (ST)

SFC

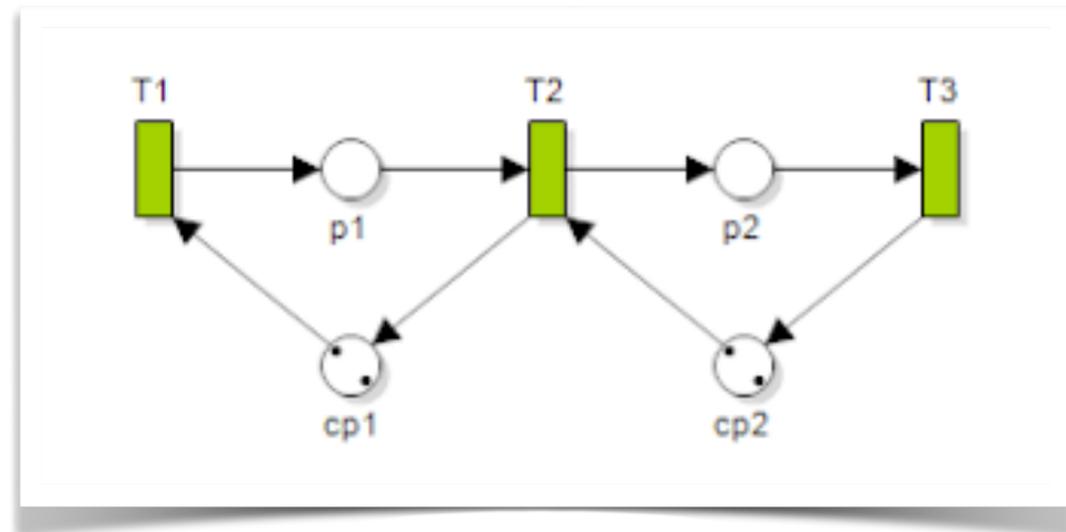


Gráficas

Diagrama de relés (LD)  
Diagrama de Blocos Funcionais(FBD)  
Funções Gráficas de Sequenciamento (SFC)

# GRAFFCET

Graphe Fonctionnel de Commande, Étapes Transitions



Baseada em Redes de Petri



1988 — Publicação da norma  
**IEC 848**

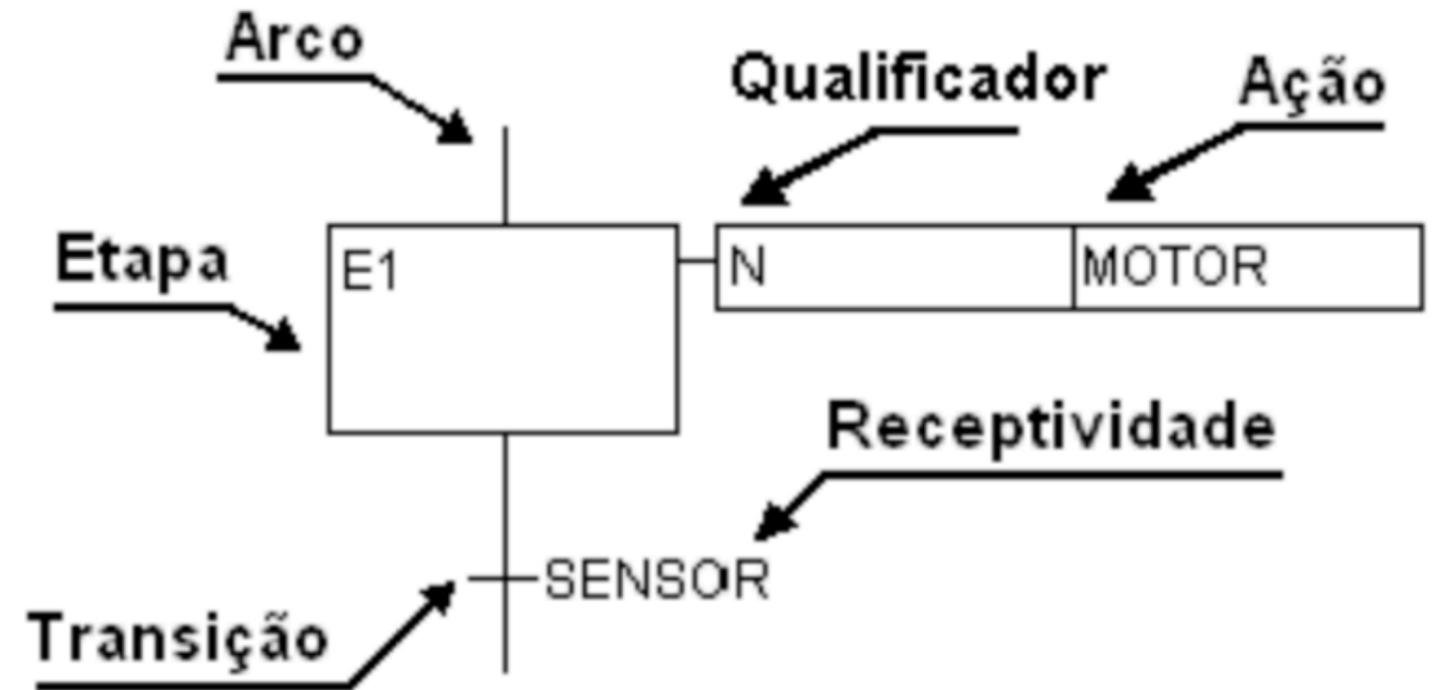
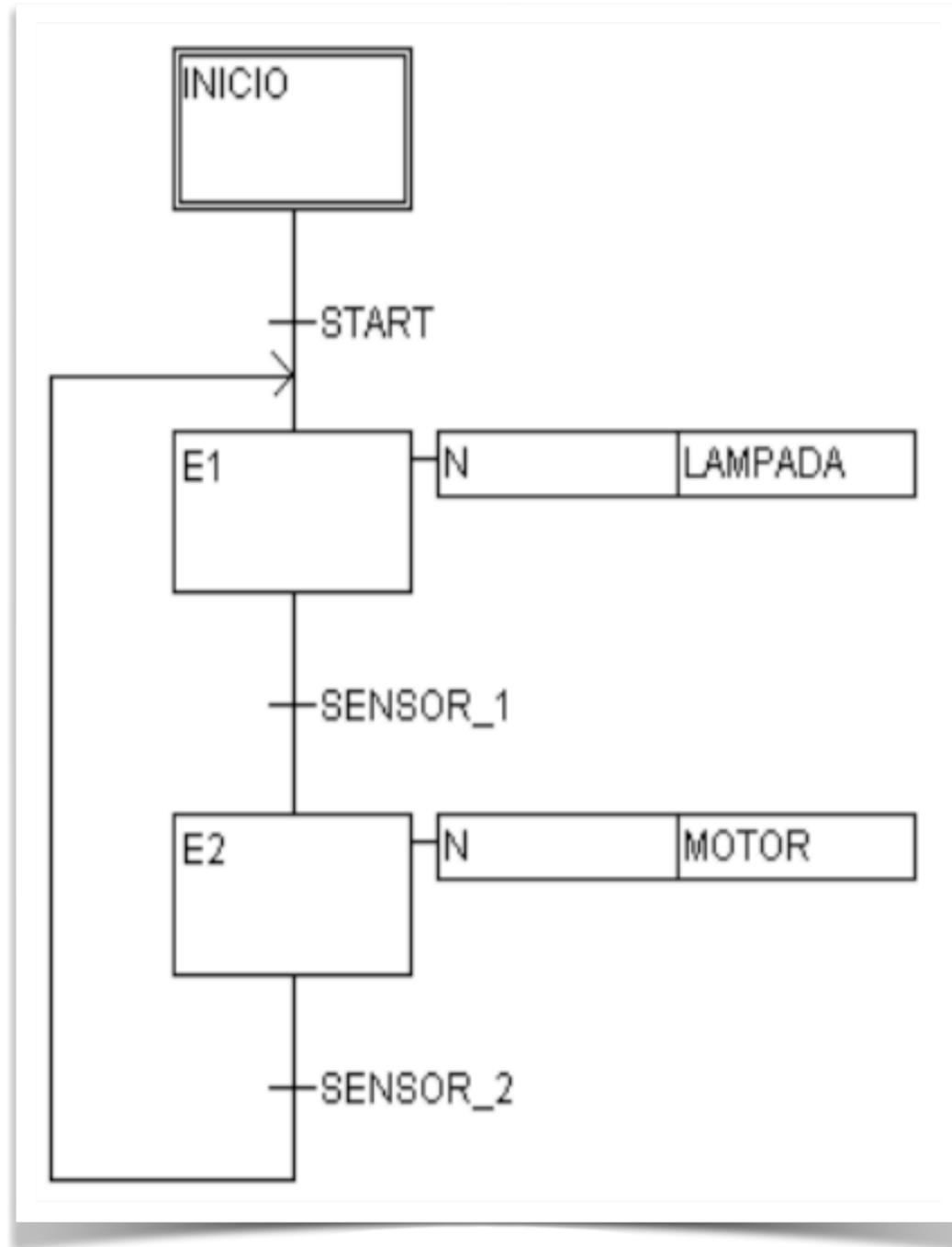
# SFC

Sequential Function Chart

# Componentes

Máquina de Estados Finitos

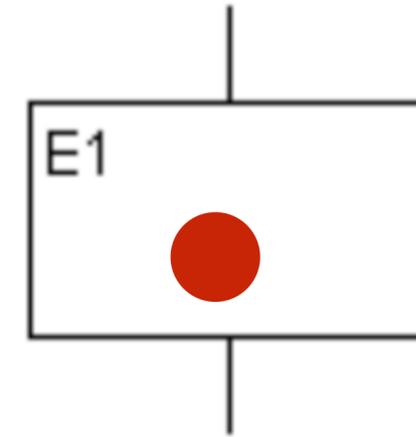
SFC



# Etapa

Condição invariável e bem definida do sistema

**pode estar *ativa* ou não**

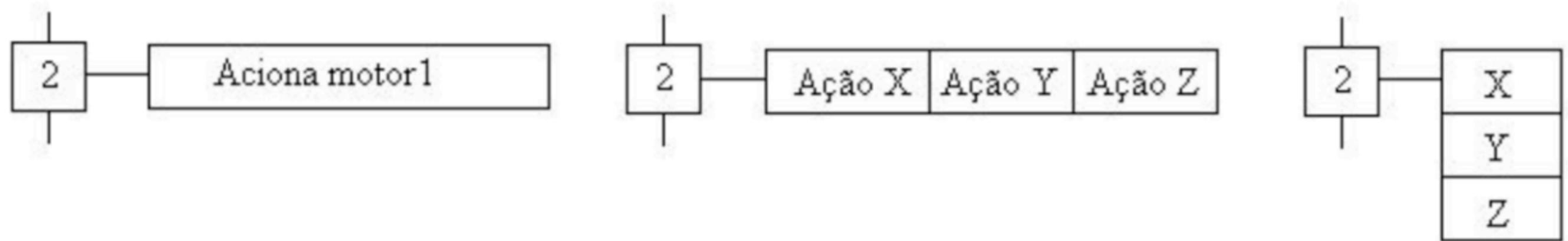


**Etapa inicial**



# Ação associada à etapa

Executadas quando a etapa associada estiver **ativa**

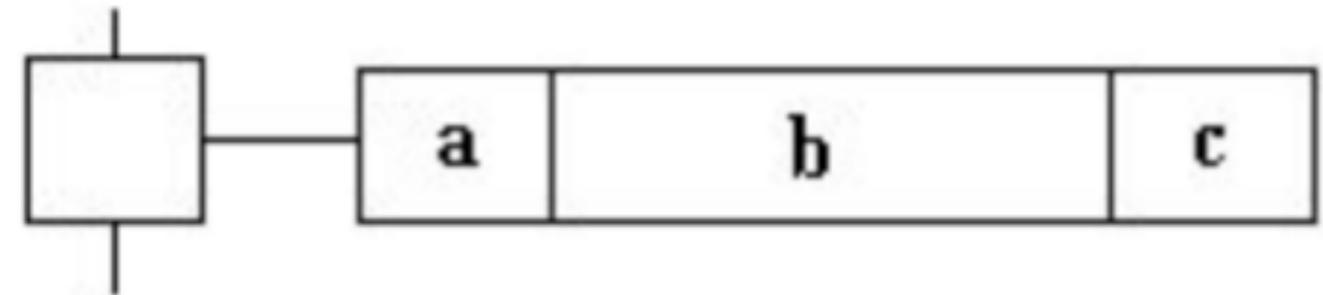


Pode-se definir se ação será mantida ou finalizada após **desativação** da etapa



# Qualificadores das ações

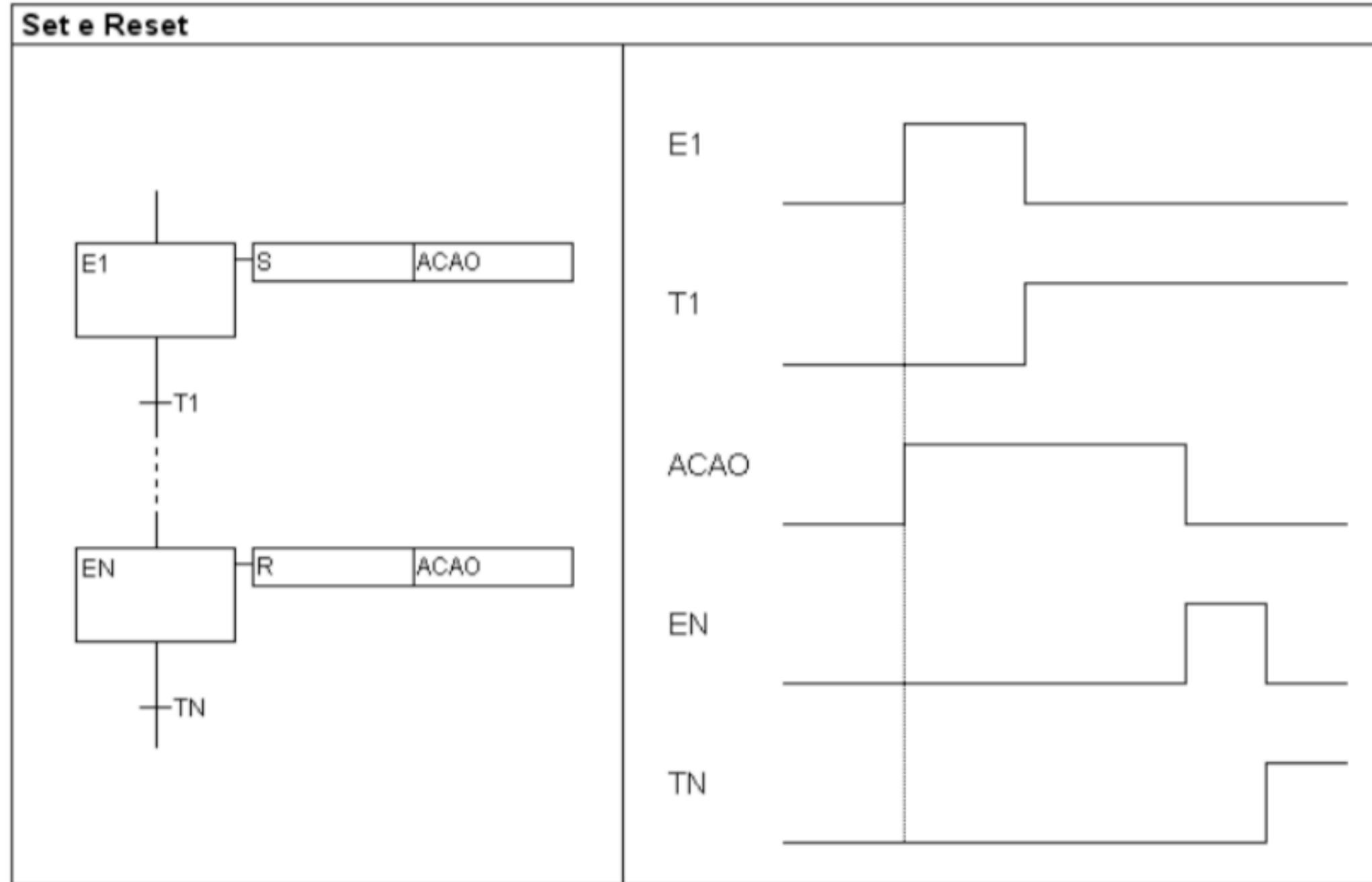
Condicionam e/ou limitam uma determinada ação



- 1) **Campo 'a'** - Qualificador que define como a ação associada à etapa será executada;
- 2) **Campo 'b'** - Declaração textual ou simbólica da ação;
- 3) **Campo 'c'** - Referência do sinal de retorno que será verificado pela transição seguinte.

Qualificadores Definidos	Função
S	Stored
D	Delayed
L	Time limited
P	Pulse shaped
C	Condition

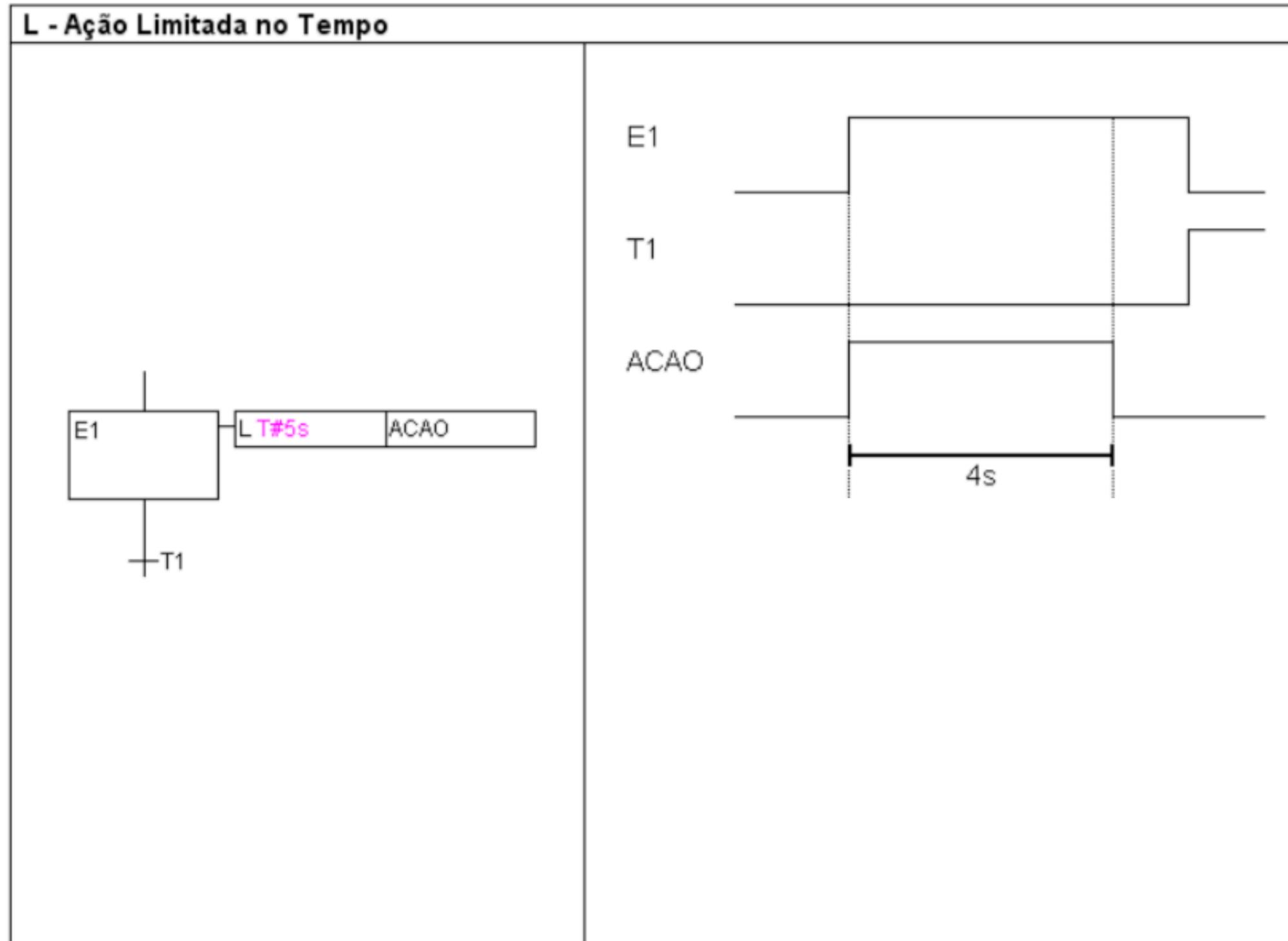
# Qualificadores S e R



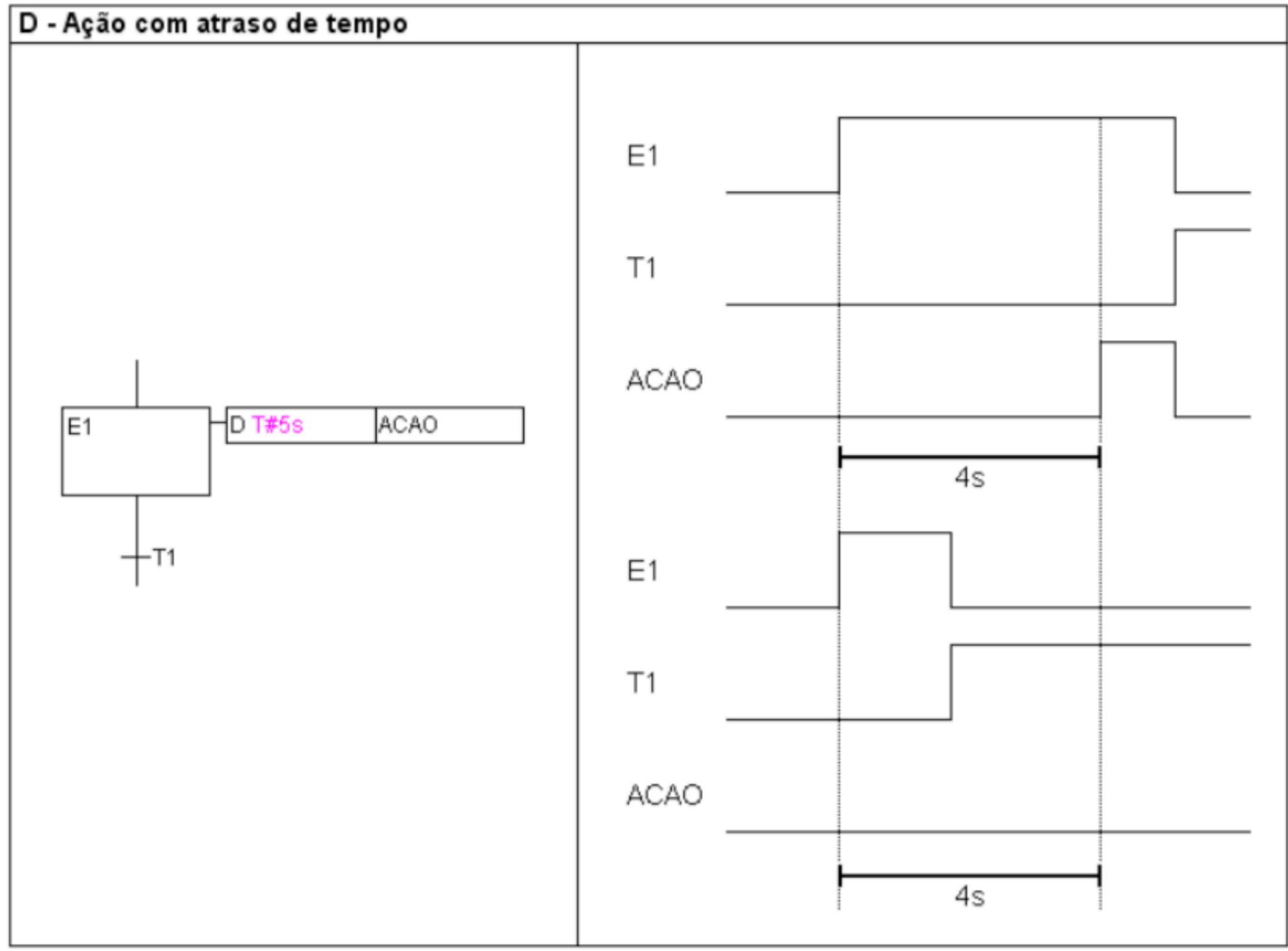
# Qualificador L

Máquina de Estados Finitos

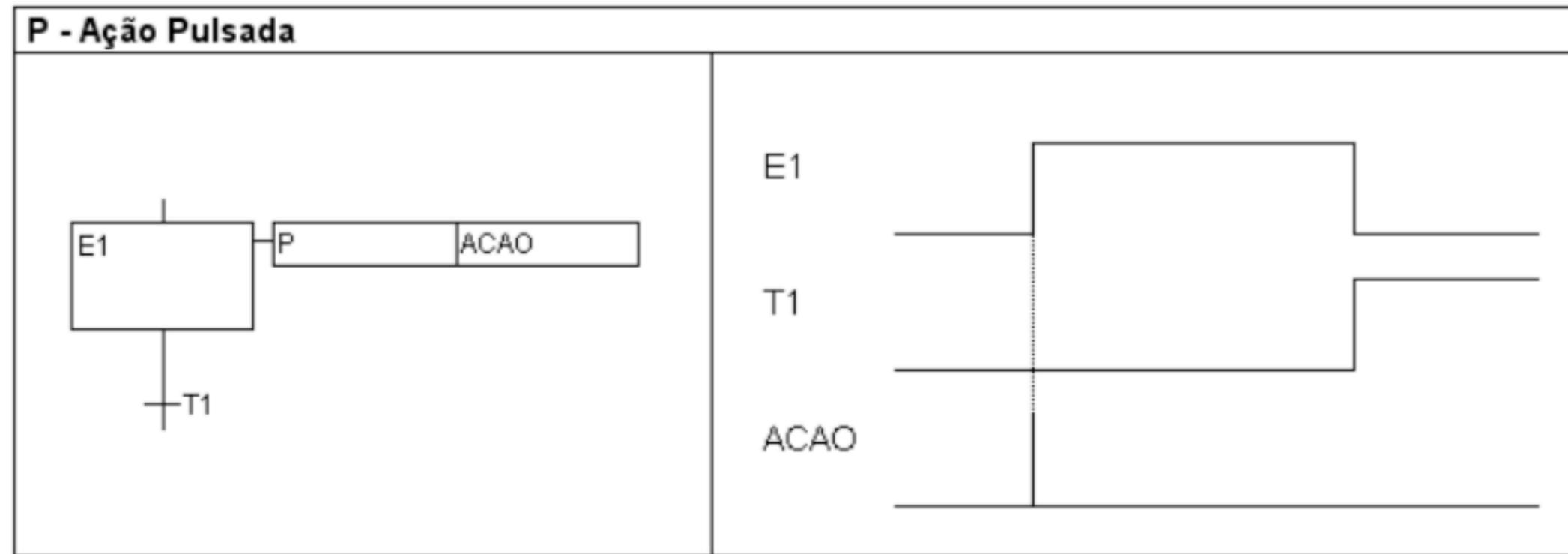
SFC



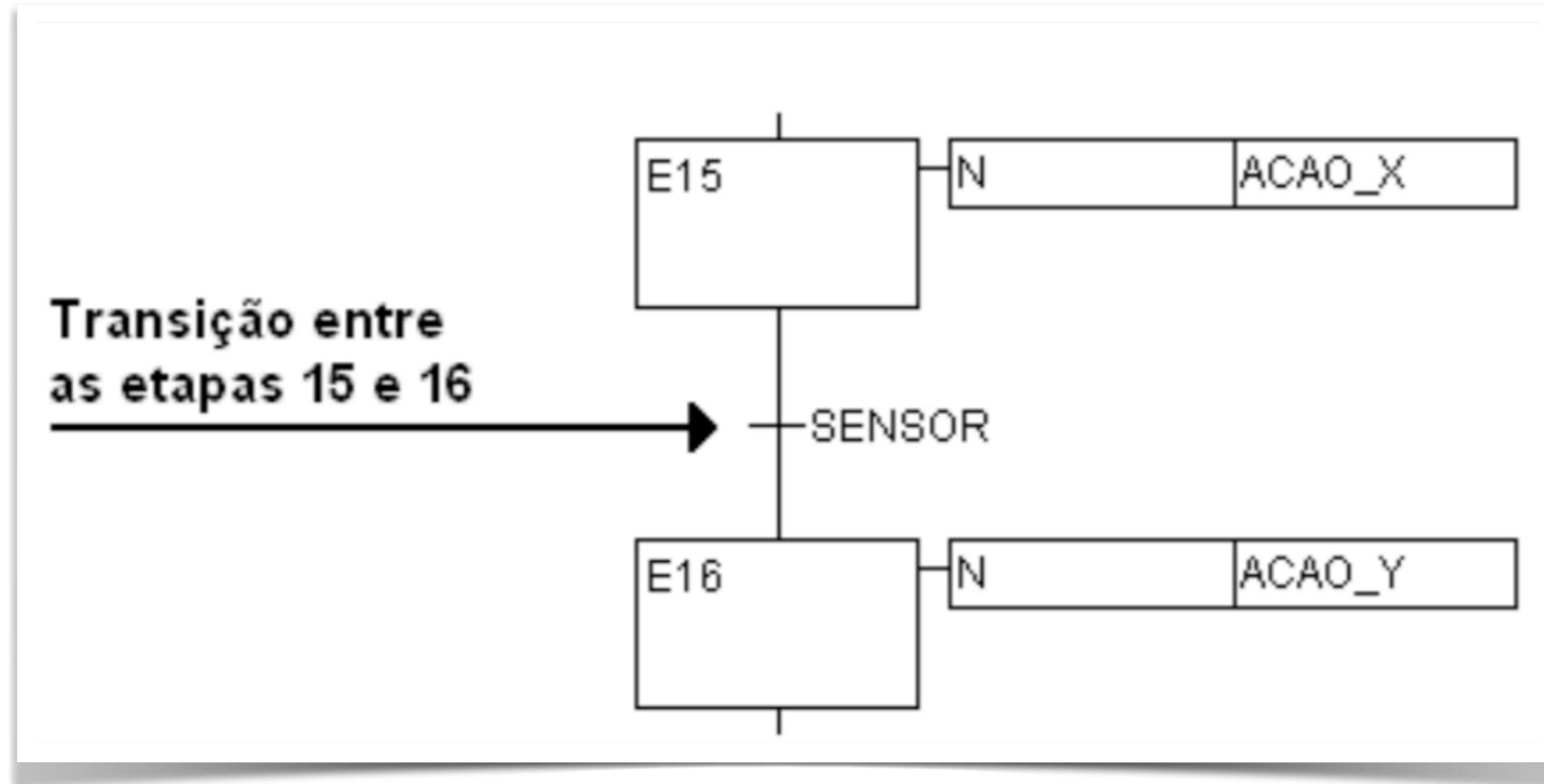
# Qualificador D



# Qualificador P



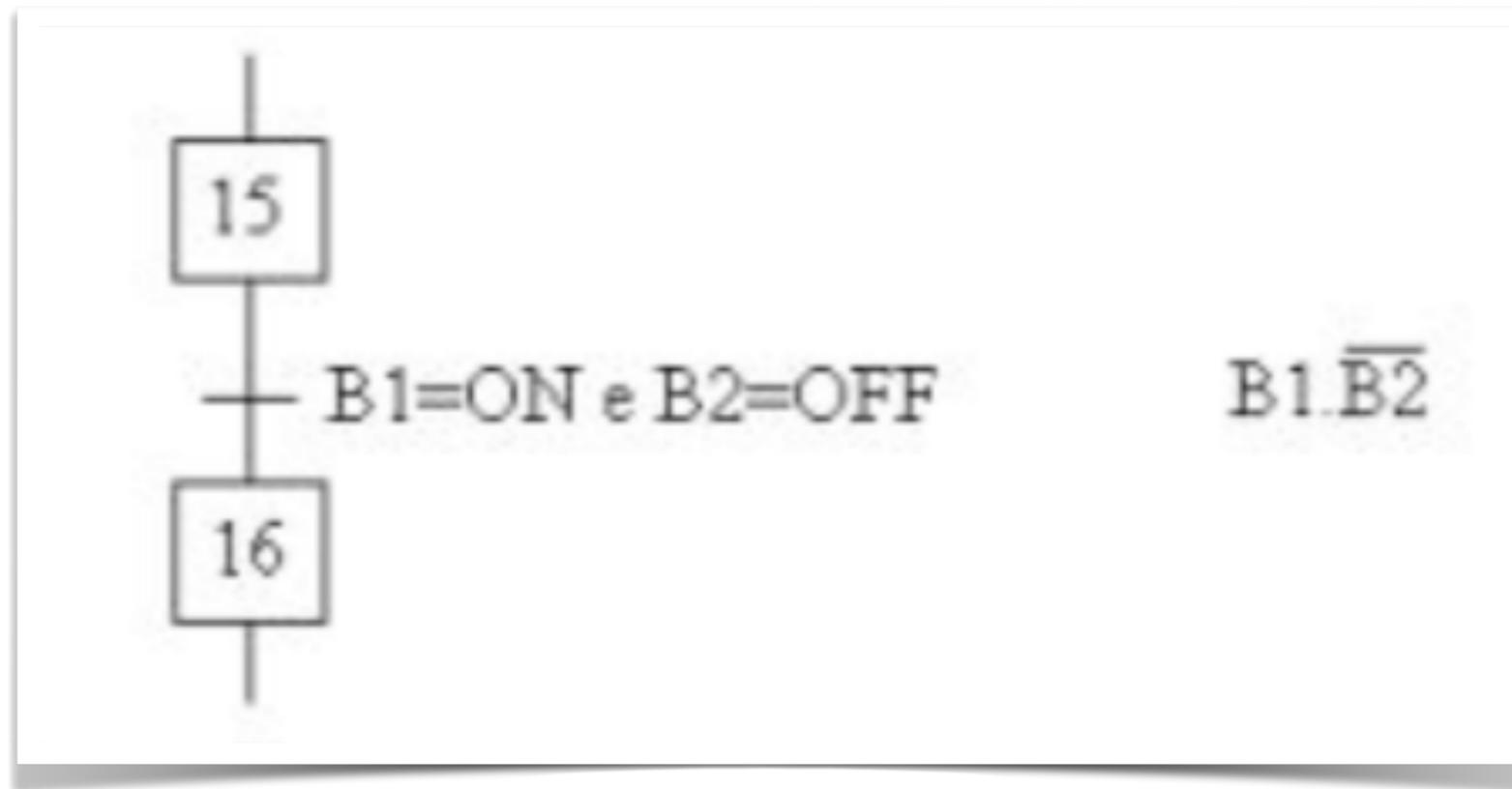
# Transições



Transição encontra-se **habilitada** se etapa(s) precedente estiver(em) **ativa**(s)

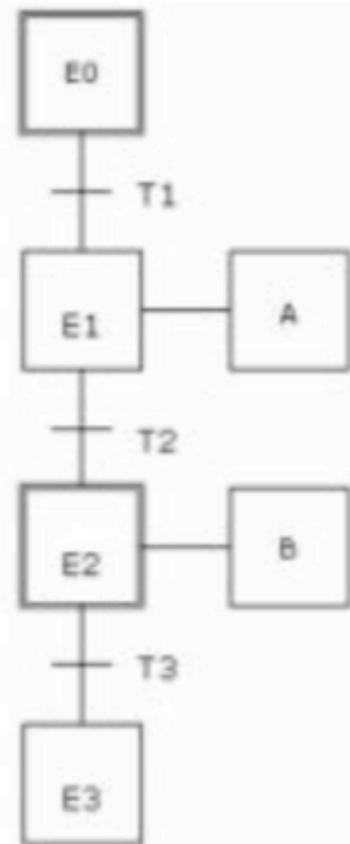
# Receptividade

Função **lógica (binária)** associada a cada transição

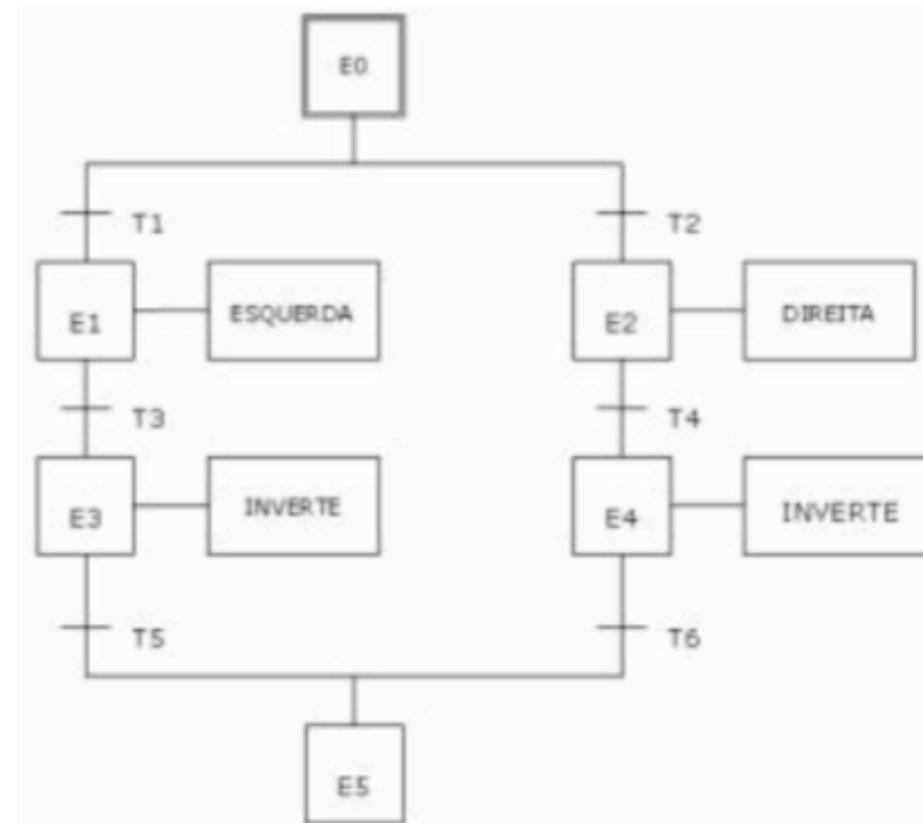


# Estruturas sequenciais

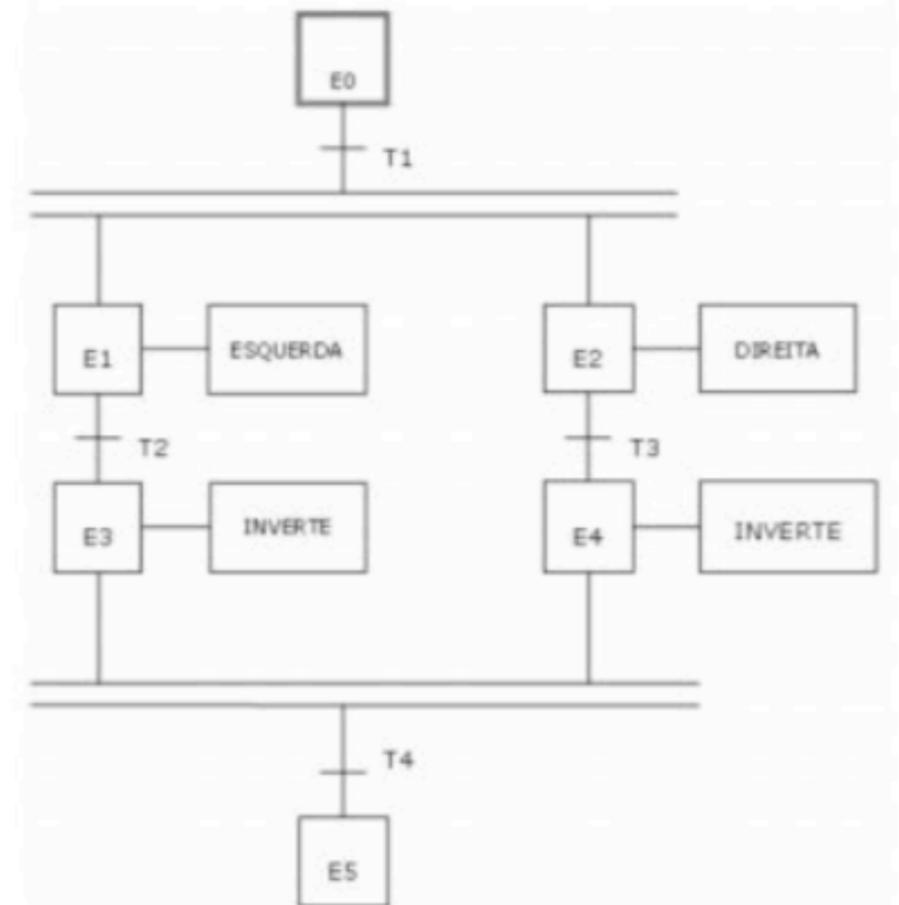
## Sequência Única



## Seleção entre Sequências

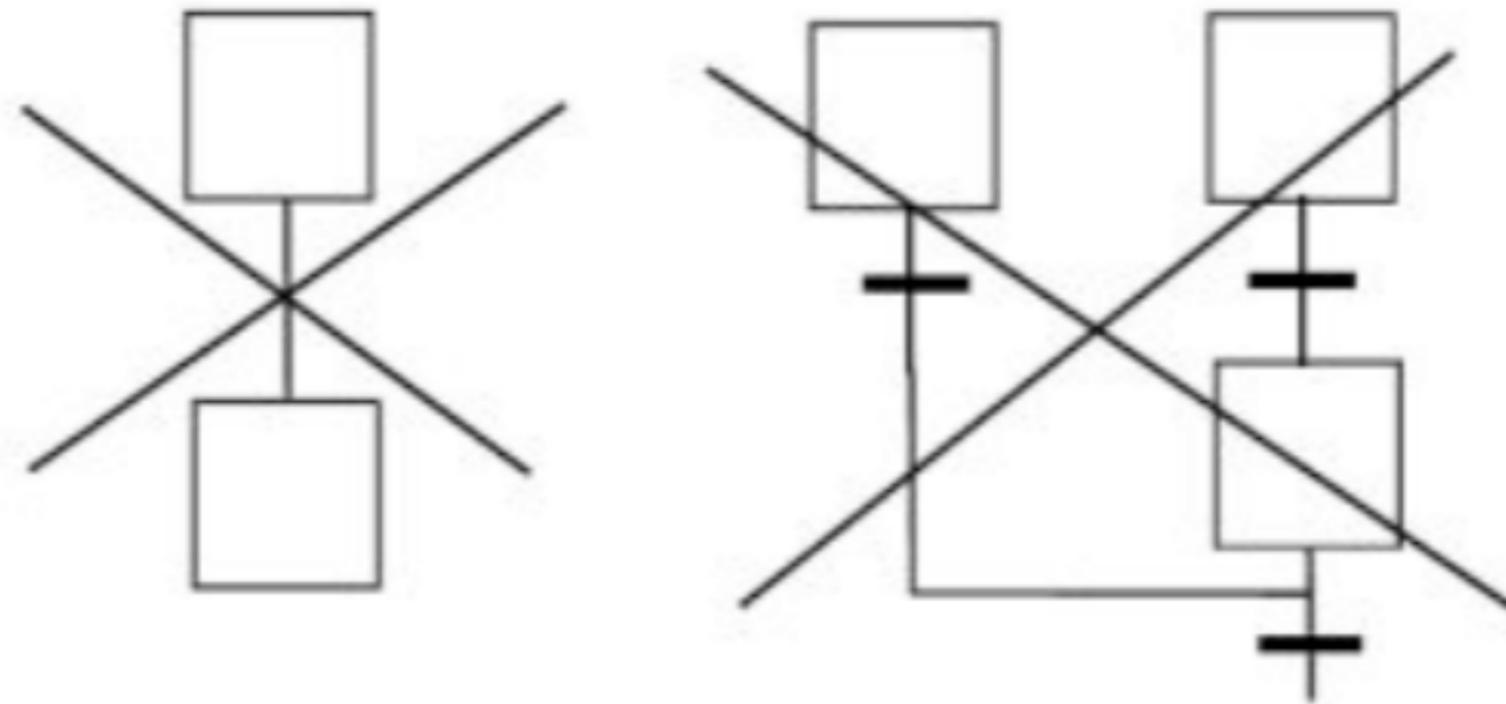


## Sequências Paralelas



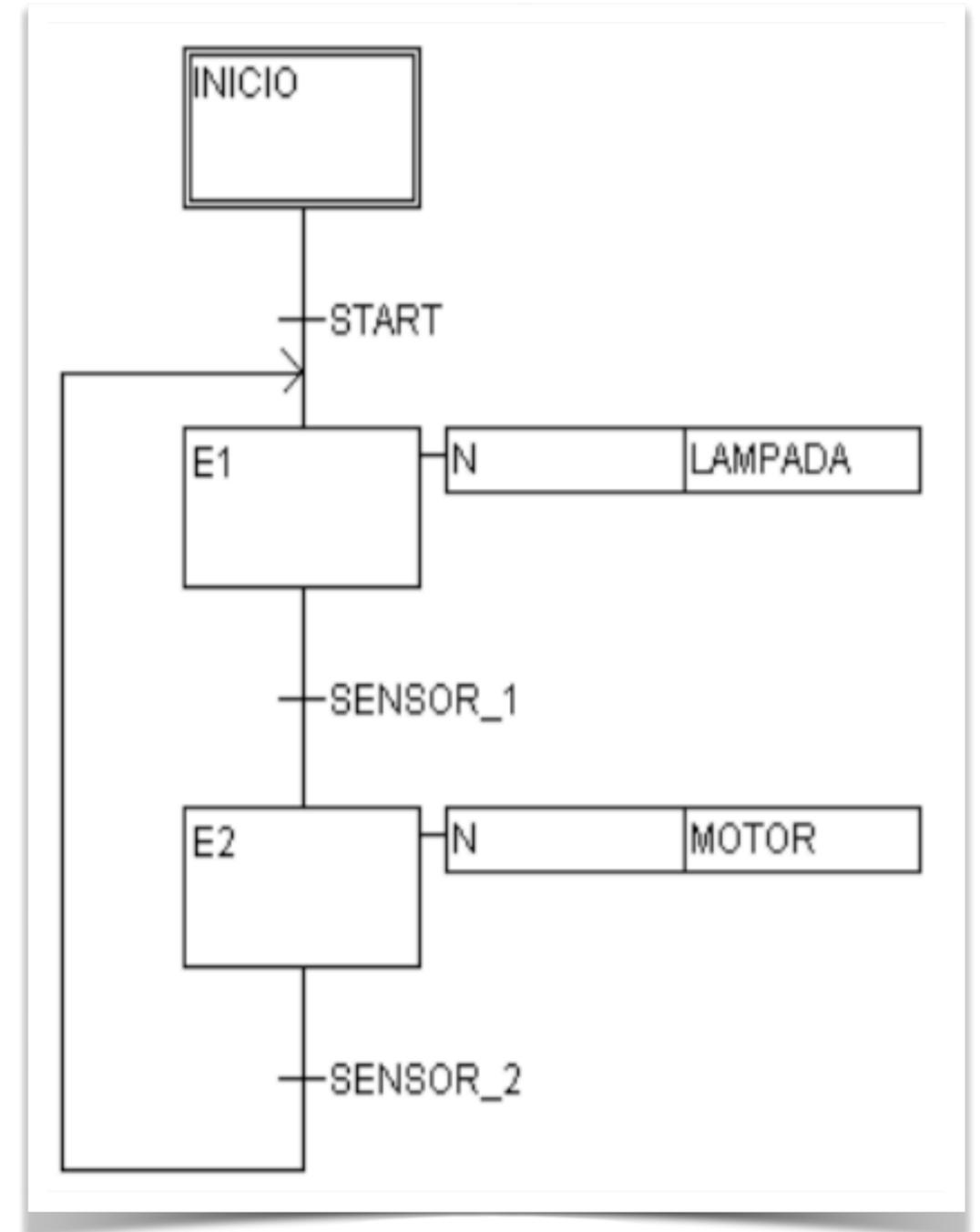
# Regras de evolução

- Duas etapas nunca podem estar ligadas diretamente;
- Duas transições nunca podem estar ligadas diretamente.



# Regras de evolução

- Deve existir pelo menos uma etapa inicial;
- Uma transição só é transposta se estiver habilitada e a condição associada for verdadeira;
- A transposição de uma transição ocasiona a ativação das etapas imediatamente seguintes e a desativação das etapas anteriores;



Dúvidas





*That's all Folks!*