Engenharia de Software

Modelo de Dinâmica Parte 1

Luís Morgado

Instituto Superior de Engenharia de Lisboa

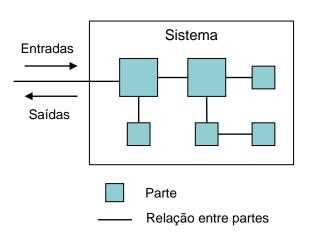
Departamento de Engenharia de Electrónica e Telecomunicações e de Computadores

Modelação de um Sistema Computacional

- O modelo de um sistema computacional é caracterizado por três perspectivas principais:
 estrutura, dinâmica e comportamento
- A estrutura de um sistema é o conjunto de partes e relações entre partes que o constituem
- A estrutura define a organização de um sistema num espaço, sendo um sistema lógico (software), o espaço é lógico, concretizado na memória do sistema, as partes residem nessa memória sob a forma de valores na memória (estruturas de dados), relacionadas através de relações de dados, por exemplo, dados que referenciam outras estruturas de dados
- A configuração das partes do sistema (os seus valores em memória) podem mudar e evoluir ao longo do tempo, determinando o comportamento do sistema. As configurações de valores que um sistema, ou parte de um sistema, pode assumir que determinam os comportamentos possíveis, constituem o seu **estado**

Estrutura

- Organização no espaço
- Denota as partes e as relações entre partes de um sistema
- Estado (memória)
 - Configuração do sistema relevante para caracterizar o seu comportamento



Modelação de um Sistema Computacional

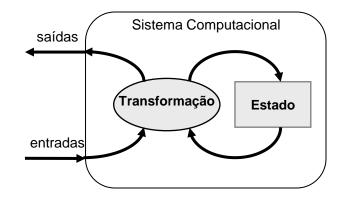
- Os vários estados que um sistema pode assumir e a forma como eles evoluem ao longo do tempo para produzir o comportamento do sistema, constitui a dinâmica do sistema
- O **comportamento** do sistema corresponde à forma como o sistema age, ou seja, activa as suas saídas (gera informação de saída), em função das suas entradas (informação de entrada, proveniente do exterior) e do seu estado interno
- Um sistema computacional geral, pode assim ser caracterizado de forma abstracta através de uma representação de estado, que define em cada momento a configuração interna do sistema, e de uma função de transformação que gera as saídas e o próximo estado em função das entradas e do estado actual do sistema

Dinâmica

- Evolução no tempo
- Descreve os estados que um sistema pode assumir e a forma como eles evoluem ao longo do tempo, determinando o comportamento do sistema

Comportamento

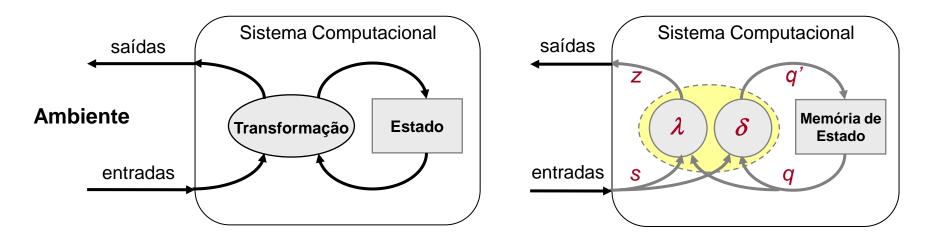
- Corresponde à forma como o sistema age (gera as saídas) perante a informação proveniente do exterior (entradas)
- Expressa a estrutura e a dinâmica do sistema



Modelo de Dinâmica de um Sistema

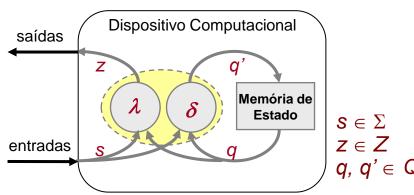
A dinâmica pode ser expressa como uma função de transformação que, perante o estado actual e as entradas actuais, produz o estado seguinte e as saídas seguintes

A função de transformação pode, por sua vez, ser descrita com base em duas funções distintas, uma função de transição de estado δ e uma função de saída λ



Esta representação de um sistema computacional é independente da forma concreta como este possa ser implementado em termos físicos, por exemplo, o suporte físico pode ser mecânico, electrónico ou outro

- Entradas e saídas abstraídas em termos dos conjuntos de símbolos que nelas podem ocorrer:
 - Esses conjuntos de símbolos são designados alfabetos
 - Consideremos um alfabeto de entrada Σ e um alfabeto de saída Z
- Estado interno do sistema descrito em termos de um conjunto de estados possíveis:
 - -Q
- Função de transformação do sistema descrita com base em duas funções distintas δ e λ:
 - Função de transição de estado:
 - $\delta: Q \times \Sigma \to Q$
 - Função de saída:
 - $\lambda: Q \times \Sigma \to Z$



Um modelo formal de computação pode ser descrito como um quíntuplo $(Q, \Sigma, Z, \delta, \lambda)$, onde:

- -Q é o **conjunto de estados** que caracterizam o sistema
- $-\Sigma$ é o conjunto de símbolos de entrada (o **alfabeto de entrada**)
- -Z é o conjunto de símbolos de saída (o *alfabeto de saída*)
- $-\delta: Q \times \Sigma \to Q$ é a função de transição de estado
- $-\lambda: Q \times \Sigma \rightarrow Z$ é a função de saída

Este tipo de modelo descreve um mecanismo computacional designado *Máquina de Estados*

- A sua implementação física implica que o número de estados possíveis seja finito
- Máquinas de Estados Finitos

Duas formulações distintas da função de saída λ

 Máquinas de Mealy, nas quais a função de saída depende das entradas, ou seja:

$$\lambda: Q \times \Sigma \to Z$$

 Máquinas de Moore, nas quais a função de saída não depende das entradas, ou seja:

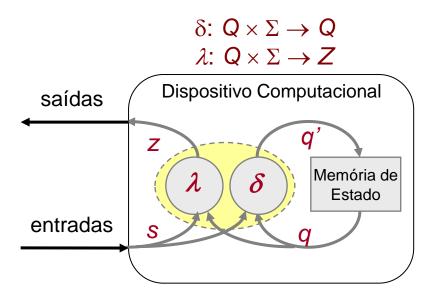
$$\lambda: Q \to Z$$

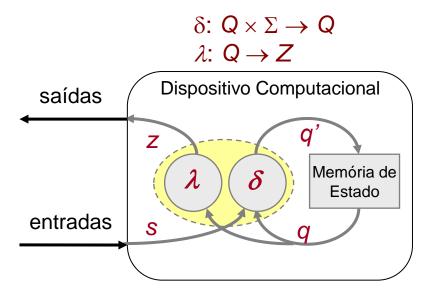
Máquinas de Mealy

A função de saída depende das entradas

Máquinas de Moore

A função de saída não depende das entradas



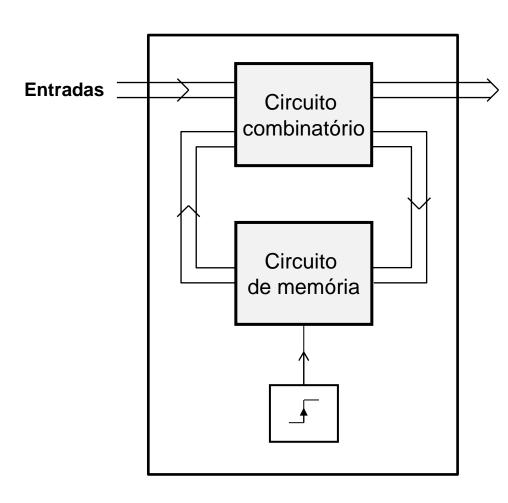


$$s \in \Sigma$$
 $z \in Z$
 $q, q' \in Q$

Exemplo: Sistema Digital

Modelo geral de um Sistema Digital

[Peatman, 1981]



Um modelo geral de sistema digital é um exemplo de concretização do conceito de máquina de estados, onde o estado é mantido através de um circuito de memória (um circuito sequencial, com um sinal temporal de referência que determina o ritmo de evolução), e a função de transformação é implementada através de um circuito combinatório que recebe as entradas e a informação de estado, produzindo as saídas e o próximo estado.

Saídas

Este modelo de circuito digital é a base de organização dos microprocessadores e dos computadores em geral

Exemplo

Sistema de Regulação Automática de Temperatura

O sistema de controlo recebe do exterior uma entrada que pode assumir valores com as seguintes representações simbólicas:

• T_REG : indica que a temperatura está dentro dos limites definidos

• T_BAIXA : indica que a temperatura está abaixo do limite mínimo

• T_ALTA : indica que a temperatura está acima do limite máximo

Por sua vez, o sistema produz uma saída para controlo dos mecanismos de aquecimento e de arrefecimento, a qual pode assumir valores com as seguintes representações simbólicas:

• AQ : sistema de aquecimento é activado

AR : sistema de arrefecimento é activado

Na ausência dos valores AQ ou AR à saída do sistema de controlo, os mecanismos de aquecimento e de arrefecimento respectivos mantêm-se inactivos.

O sistema de controlo é caracterizado por três estados:

• q_{inactivo} : os sistemas de aquecimento e de arrefecimento estão inactivos

ullet $q_{
m aquecimento}$: apenas o sistema de aquecimento está activo

• $q_{\text{arrefecimento}}$: apenas o sistema de arrefecimento está activo

Exemplo

Da descrição do problema podemos identificar:

- Um conjunto de símbolos de entrada (o alfabeto de entrada):
 - $\Sigma = \{ T_REG, T_BAIXA, T_ALTA \}$
- Um conjunto de símbolos de saída (o alfabeto de saída):
 - **Z** = { AQ, AR }
- Um conjunto de estados que caracterizam o sistema de controlo:
 - $Q = \{ q_{\text{inactivo}}, q_{\text{aquecimento}}, q_{\text{arrefecimento}} \}$

Função de transição de estado:

$$\delta: \mathbf{Q} \times \Sigma \to \mathbf{Q}$$

Q Σ	T_REG	T_BAIXA	T_ALTA
<i>q</i> _{inactivo}	$q_{inactivo}$	$q_{\sf aquecimento}$	$q_{ m arrefecimento}$
$q_{aquecimento}$	$q_{inactivo}$	$q_{\sf aquecimento}$	$q_{ m arrefecimento}$
q _{arrefecimento}	$q_{inactivo}$	$q_{ m aquecimento}$	$q_{ m arrefecimento}$

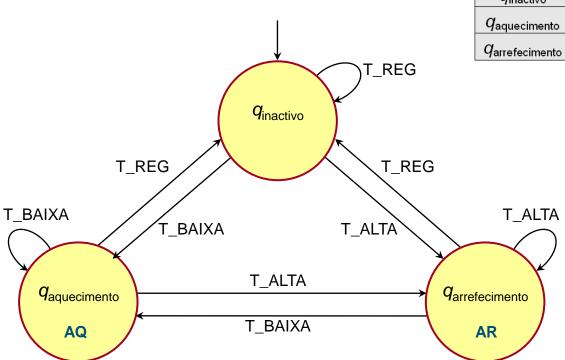
Função de saída:

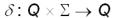
$$\lambda: \mathbf{Q} \to \mathbf{Z}$$

Q	Z
<i>q</i> _{inacti∨o}	
$q_{ m aquecimento}$	AQ
<i>G</i> arrefecimento	AR

Exemplo

Representação gráfica da dinâmica do sistema





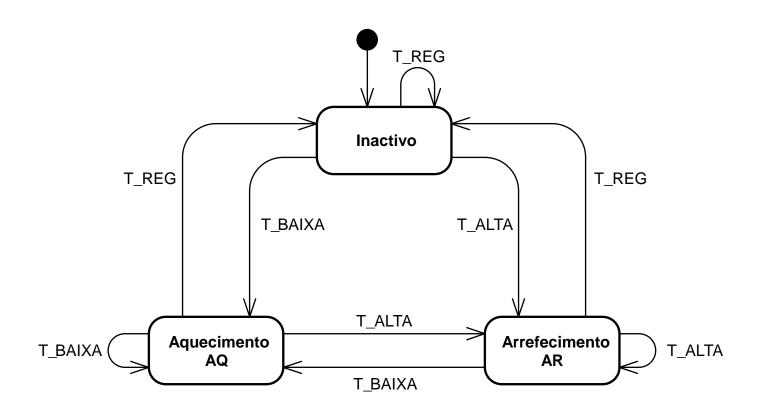
Q Σ	T_REG	T_BAIXA	T_ALTA
<i>q</i> _{inactivo}	$q_{inactivo}$	<i>Q</i> aquecimento	<i>Q</i> arrefecimento
<i>Q</i> aquecimento	$q_{inactivo}$	$q_{ m aquecimento}$	<i>G</i> arrefecimento
<i>Q</i> arrefecimento	$q_{inactivo}$	$q_{ m aquecimento}$	<i>q</i> _{arrefecimento}



Q	Z
$q_{inactivo}$	
$q_{ m aquecimento}$	AQ
<i>q</i> _{arrefecimento}	AR

Diagrama de Transição de Estado

Representação gráfica da dinâmica do sistema em linguagem UML Diagrama de transição de estado



Dinâmica de um Sistema

- Arquitectura de um sistema
 - Organização no espaço (estática)
 - Estrutura
 - Organização no tempo
 - Dinâmica

Estrutura

 Denota as partes e as relações entre partes de um sistema

Dinâmica:

 Denota a forma como as partes e as relações entre partes de um sistema evoluem no tempo

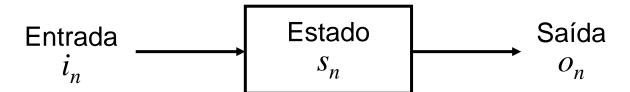
Comportamento

 Denota a forma como o sistema age ou reage perante os estímulos do ambiente envolvente

Modelação da Dinâmica de um Sistema

Estado

- Configuração da estrutura de um sistema discriminadora do comportamento do sistema
 - Representa situações relevantes da evolução de um sistema
 - Exemplo: quando se prime uma tecla de um teclado o código gerado depende do estado do CAPS-LOCK (ON / OFF)
 - Comportamento depende apenas de alguns aspectos da história de evolução, neste caso, não depende de quantas teclas foram pressionadas anteriormente
- A saída de um sistema computacional, em qualquer momento, é completamente determinada pelas suas entradas e pelo seu estado



Estado e Variáveis de Estado

Estado no sentido lato (micro)

- Cada estado representa um conjunto distinto de configurações das partes do sistema (e.g. os valores da memória)
- Problema
 - Mesmo para programas pequenos o número de estados é impraticável
 - Exemplo: programa com uma variável do tipo inteiro 32 bits
 - $^{\circ}$ 2³² = 4.294.967.296 estados

Estado como generalização (macro)

- Abstração
- Discriminação de padrões de comportamentos

Variáveis de estado

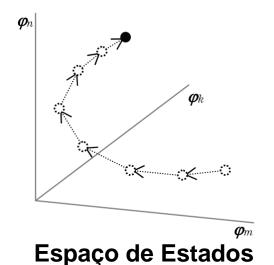
- Aspecto quantitativo
- Exemplo:
 - Parar ao fim de 1000 teclas pressionadas
 - Não é útil definir 1000 estados distintos
 - Variável auxiliar (e.g. contador de teclas)
- Podem no entanto tornar a modelação menos compreensível e flexível

O conjunto de estados de um sistema constitui um *espaço de estados*, com dimensões φ_n que correspondem aos domínios de valores do estado, cada estado corresponde a uma posição nesse espaço

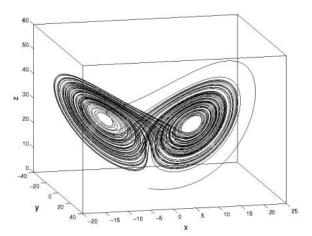
A evolução do estado de um sistema, pode ser representada geometricamente num espaço de estados sob a forma de trajetórias de evolução de estado, que descrevem padrões de comportamento do sistema independentes do tempo, constituindo uma forma gráfica de descrição do comportamento de um sistema

Se os valores de estado forem do domínio real, o espaço de estados é contínuo, sendo nesse caso designado um *espaço de fase*, pois o que é possível discriminar são as fases de evolução do estado do sistema

Domínios de valores discretos



Domínios de valores contínuos

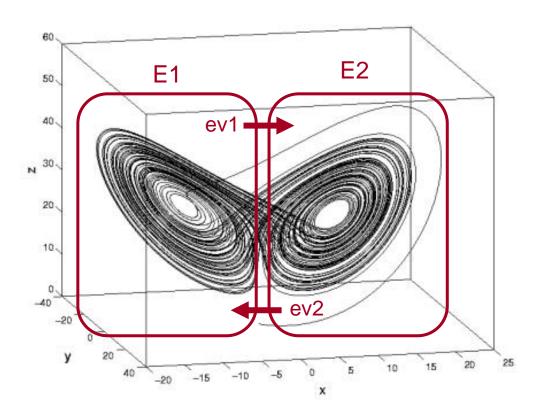


Espaço de Fase

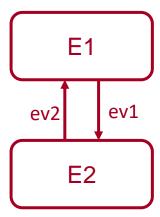
Suporte de descrição do **comportamento** de um sistema

No caso de um sistema caracterizado por um *espaço de fase* (contínuo), é possível modelar os padrões gerais de evolução de estrado, abstraindo conjuntos de trajetórias que correspondem ao mesmo tipo de comportamento, como **estados** abstractos, por exemplo, num espaço de fase de um modelo atmosférico podem ser identificados estados abstractos como *céu limpo* ou *céu nublado*

As transições entre estados podem ser abstraídas por **eventos**, os quais representam ocorrências que determinam a transição entre estados, por exemplo, a ocorrência de um determinado *nível de pressão atmosférica*

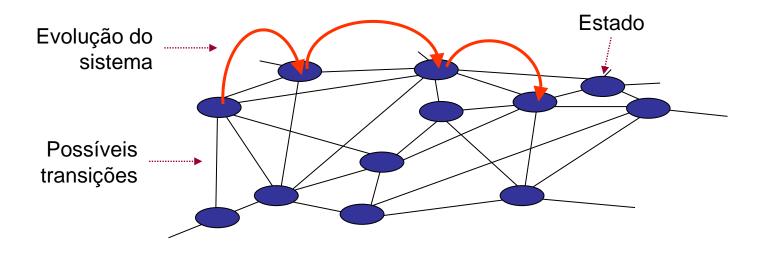


Representação abstracta



Um espaço de estados é uma forma concisa e prática de descrever a dinâmica de um sistema

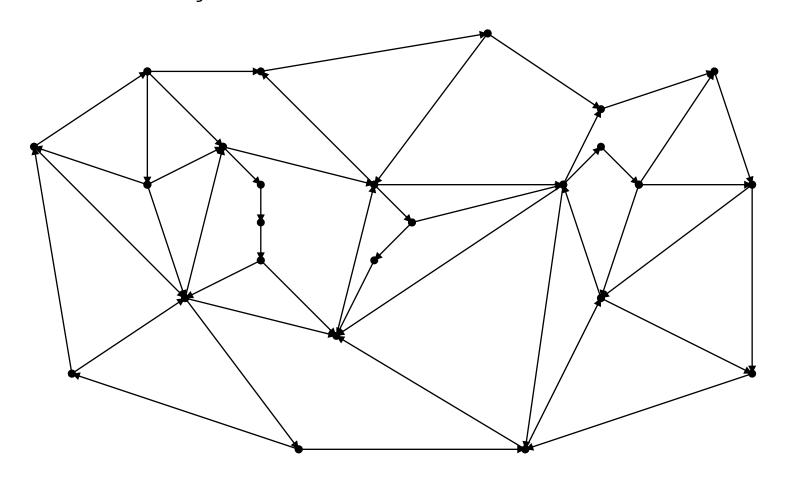
Define as evoluções válidas do sistema em termos de estados, transições entre estado e percursos de evolução de estado



Espaço de estados

Representação de um espaço de estados sob a forma de um grafo

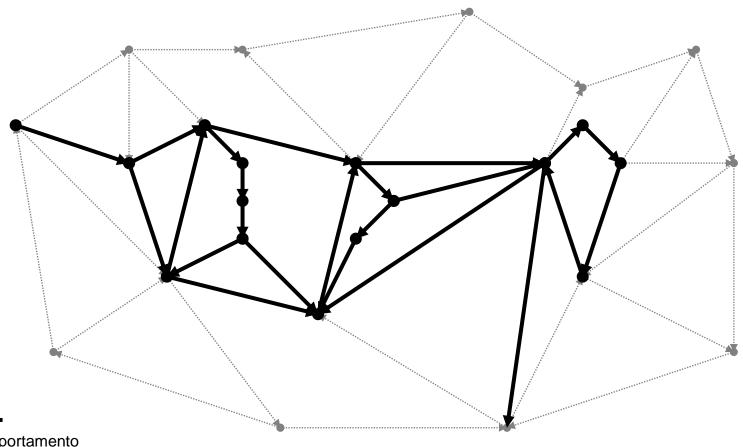
- Vértices Estados
- Arcos Transições de estado



Espaço de estados (grafo)

Definição de **comportamento válido** com base no espaço de estados Apenas o comportamento definido é admissível

- Abstrai complexidade procedimental
- Definição robusta de comportamento

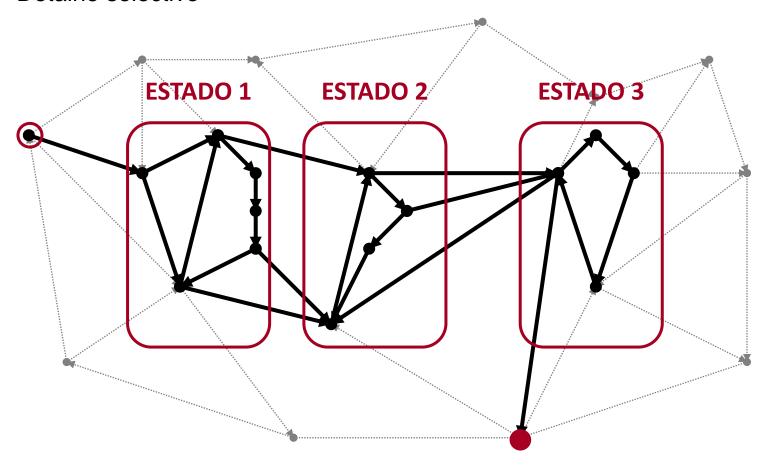


Comportamento válido

Abstracção

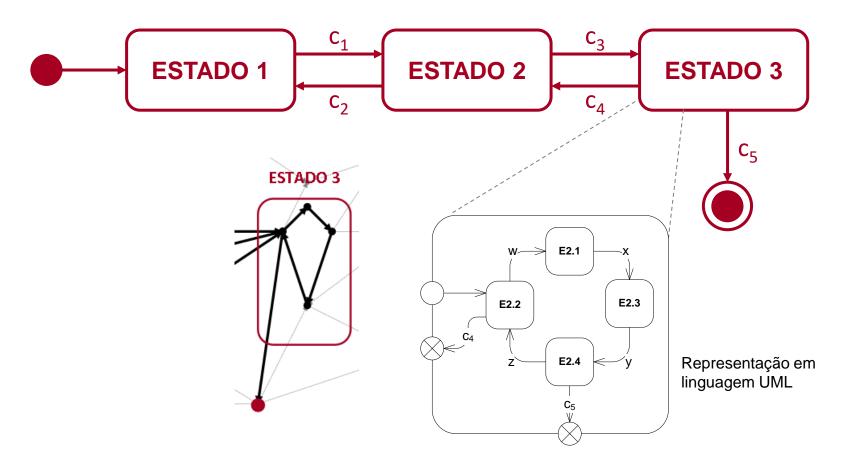
Permite gerir a complexidade da dinâmica

- Modularização
- Detalhe selectivo



Diagramas de Transição de Estado

Um diagrama de transição de estado é uma representação gráfica dos estados e transições entre estados de um sistema, a qual pode ser definida a diferentes níveis de abstracção

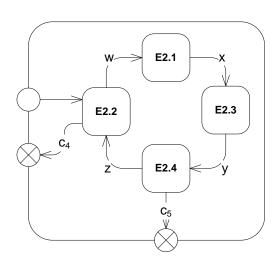


Abstracção → Controlo de complexidade

Máquinas de Estados

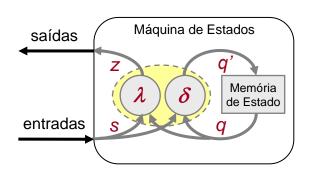
Um diagrama de transição de estado define a dinâmica de um sistema ou parte de um sistema, a qual pode ser concretizada sob a forma de uma máquina de estados

Diagrama de Transição de Estado



Especifica a dinâmica de um sistema ou parte de um sistema

Máquina de Estados



Concretiza a dinâmica de um sistema ou parte de um sistema

Caso Prático

Torniquete de controlo de acessos

Pretende-se implementar o sistema de controlo de um torniquete de controlo de acessos.

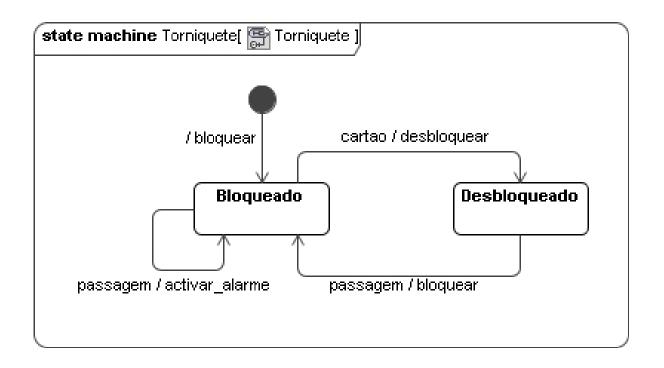
O torniquete é composto por um detector que indica a presença de cartão válido e a ocorrência de passagem e por um actuador composto por um trinco que permite bloquear e desbloquear o acesso e por uma sirene de alarme que pode ser activada ou desactivada.

Por omissão o acesso está bloqueado. Quando é detectado um cartão válido o acesso é desbloqueado, voltando a ser bloqueado após a passagem. No caso da passagem ser forçada deve ser activada a sirene de alarme, que se manterá activa até o sistema ser reiniciado.



Caso Prático

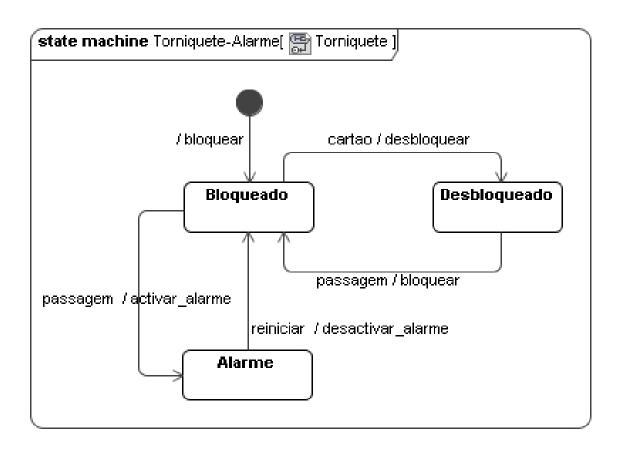
Torniquete de controlo de acessos (sem estado de alarme)



Limitação: não possibilita a desactivação do alarme em caso de alarme activo

Caso Prático

Torniquete de controlo de acessos com estado de alarme



Bibliografia

[Booch et al., 1998]

G. Booch, J. Rumbaugh, I. Jacobson, *The Unified Modeling Language User Guide*, Addison Wesley, 1998.

[Eriksson et al., 2004]

H. Eriksson, M. Penker, B. Lyons, D. Fado, UML 2 Toolkit, Wiley, 2004.

[Peatman, 1981]

J. Peatman, The Design of Digital Systems, McGraw-Hill, 1981.

[OMG, 2020]

Unified Modeling Language (Specification), OMG, 2020.