

2009 - 2010



# Inginerie Software pentru Comunicatii (ISC / RST)

Titular curs: **Eduard-Cristian Popovici**

Suport curs: <http://discipline.elcom.pub.ro/isc/>

Moodle: <http://electronica07.curs.ncit.pub.ro/course/category.php?id=4>

## Continut curs

### 1. Introducere in ingineria software

- 1.1. Necesitatea unei abordari sistematice a dezvoltarii software
- 1.2. Abordari si metodologii larg utilizate in ingineria software

### 2. Introducere in limbajul UML

- 2.1. Definirea, rolul si istoricul limbajului de modelare unificat (UML)
- 2.2. Tipuri de diagrame UML. Organizarea ierarhica a diagramelor

### 3. Diagrame UML statice

- 3.1. Diagrame UML de clase
- 3.2. Diagrame UML de obiecte
- 3.3. Diagrame UML de pachete
- 3.4. Diagrame UML de componente
- 3.5. Diagrame UML de structuri compozite
- 3.6. Diagrame UML de *deployment* (amplasare)

## Continut curs

### 4. Diagrame UML dinamice

- 4.1. Diagramele UML de caz de utilizare
- 4.2. Diagrame UML de comunicare si de robustete
- 4.3. Diagrame UML de secventa si de sumar al interactiunilor
- 4.4. Diagrame UML de masini de stari
- 4.5. Diagrame UML de activitati
- 4.6. Diagrame UML de timp

### 5. Introducere in procesul de dezvoltare Rational unificat (RUP)

- 5.1. Organizarea iterativa a proiectelor
- 5.2. Fazele si activitatile procesului RUP

### 6. Introducere in managementul si organizarea proceselor de dezvoltare

### 7. Elemente de reutilizabilitate a software-ului. Pattern-uri de proiectare

*A picture is worth more than 1024 lines of code*

## 4. Diagrame UML dinamice

### 4.6. Diagrame UML de timp (*timing*, sincronizare)

## 4.6. Diagrame UML de timp (timing, sincronizare)



### Diagramele de timp (*timing*, sincronizare)

Diagramele de timing (sincronizare) sunt

- o cale **alternativa** de reprezentare a **diagramelor de secventa**
- care **arata explicit schimbarile de stare** de-a lungul liniei vietii
  - **reprezentata** insa **pe orizontala** in loc de verticala
- precum si **intervalele de timp dintre evenimente si mesaje**
  - in unitati de timp
- utile in aplicatiile cu **constrangeri de timp**

In diagrama urmatoare, se folosesc:

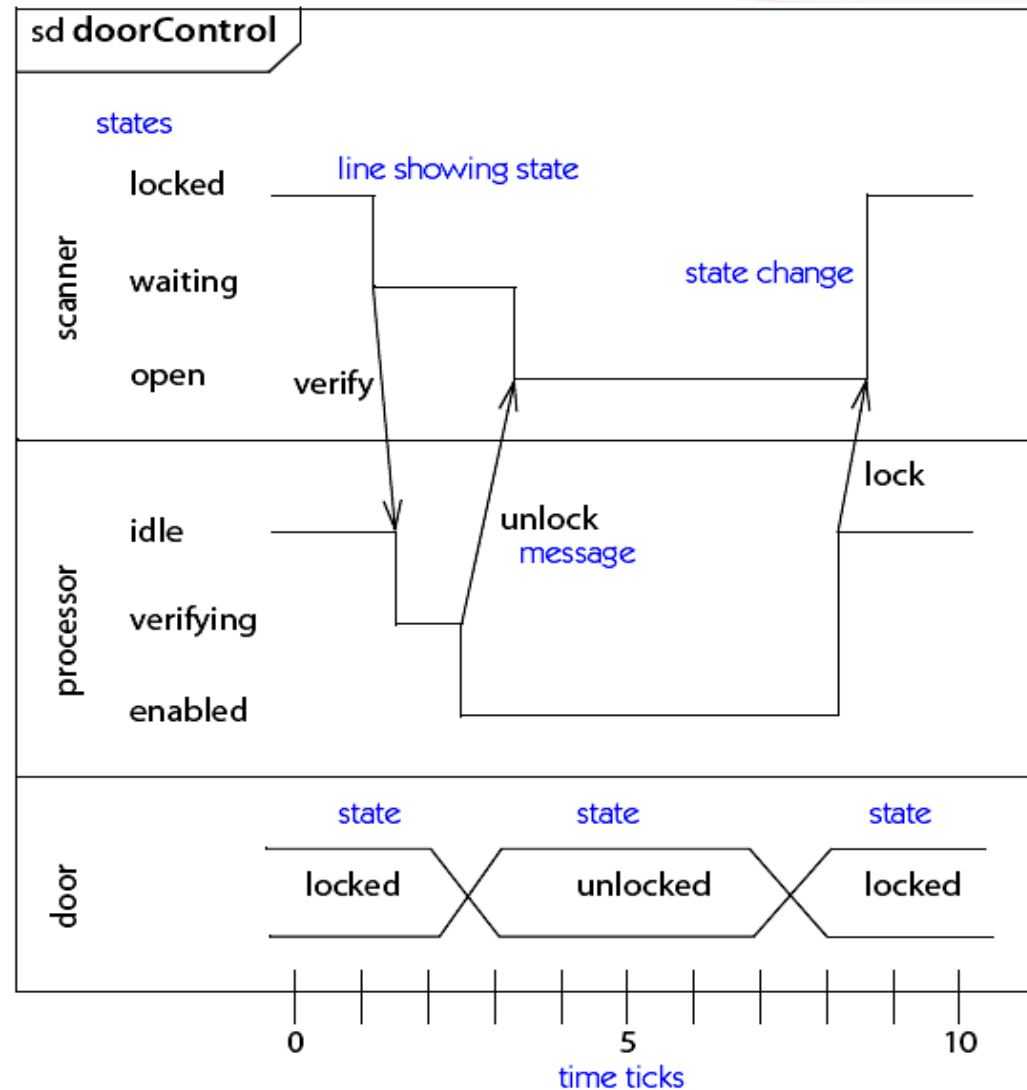
- **notatia robusta** pentru **scanner** si **processor**, si
- **notatia concisa** pentru **door** (usa).

## 4.6. Diagrame UML de timp (timing, sincronizare)



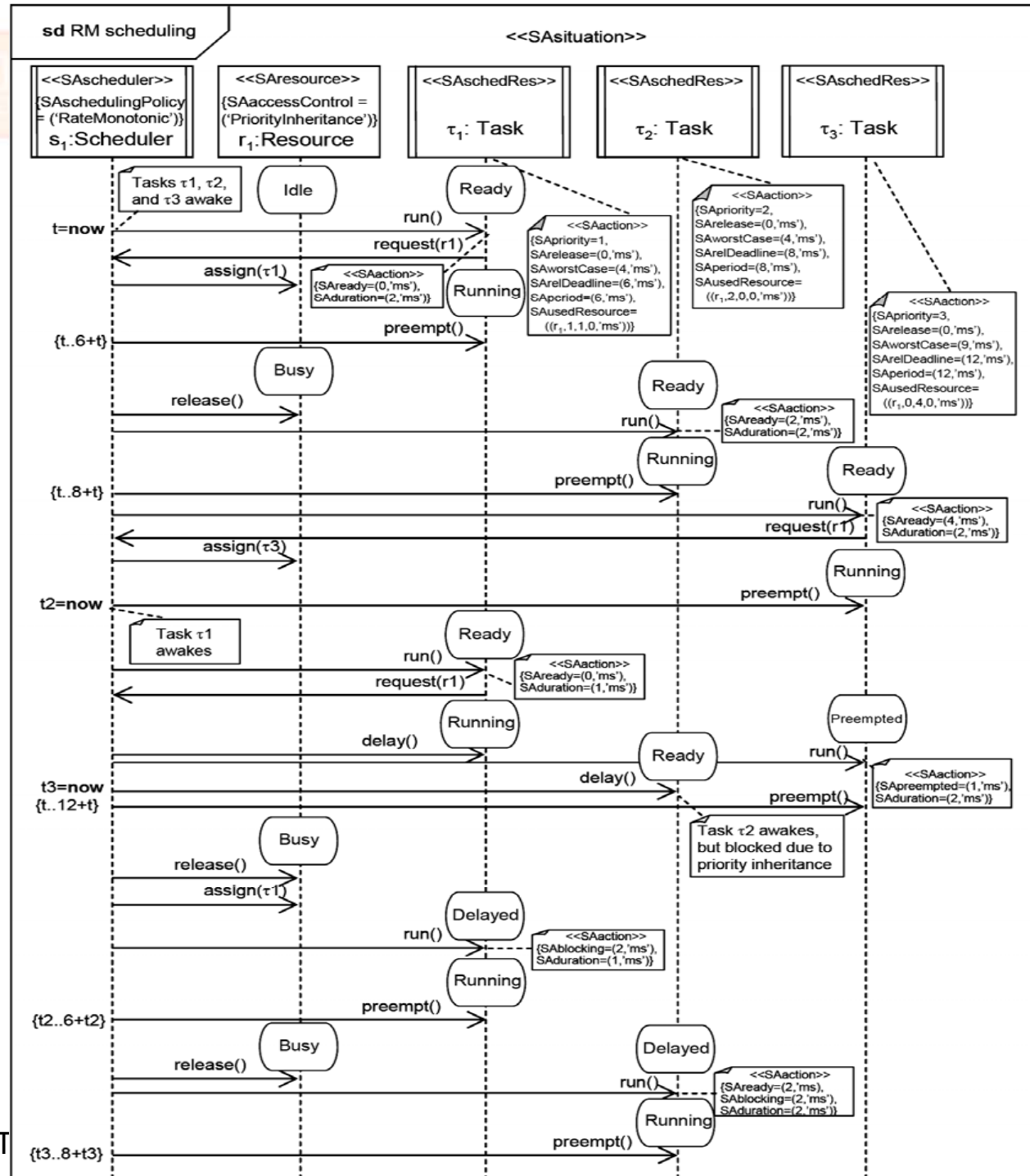
### Diagramele de timp (*timing*, sincronizare)

timing diagram



# Diagramele masinilor de stari

Starile pot fi reprezentate si in diagramele de secventa de secventa (MSC)



# 4.6. Diagrame UML de timp (timing, sincronizare)



## Legatura dintre diagramele de secventa si cele de timing

