# UNIVERSIDADE NOVA DE LISBOA

FCT – Departamento de Engenharia Electrotécnica

#### Secção de Robótica e Manufactura Integrada

|  |  |
| --- | --- |
| Lab work nº 3 | |
| Discipline | Real-time Systems |
| Year | 2015/2016 |
| Goals | Development of real-time applications using graphical languages |
| Sessions | 3 Lessons x 3 hours each + 6 Hours outside lab. |
| Due date | 2015/12/13 |
| Concrete Objectives:   1. Utilization of graphical languages to model real-time systems    1. Modeling of concurrent/simultaneous behaviors    2. Modeling with Petri nets and LADDER diagrams    3. Solving of a Petrinet problem    4. Solving a problem with LADDER diagrams 2. Fill in the lab. report (annex 2), during the work, to be delivered at the specified due date. | |

# Annex 1 – Work lab description

The purpose of this work is to solve problems with real-time constraints using the modeling approaches based on graphical languages, namely, Petri Nets (PN) and LADDER diagrams.

This work is composed of three parts, namely:

1. Model and simulate PNs of several types, learn how they behave, and perceive fundamental properties (as provided in theoretical lessons).
2. Apply the PN modeling framework in the solution of a real problem.
3. Transformation of a PN in a corresponding LADDER diagram and execute it in a PLC.

**1st Part - Use the Hpsim tool for modeling PN and study their properties**

In the first part of this work, we have the opportunity to simulate the behavior of Petri-Nets during their life-cycle, using HPsim (fig. 1). After starting a simulation, we can visualize the enabling and firing of a PN’ transitions and the tokens in each place. The work comprises modeling distinct PN types, which also exhibit fundamental PN properties (conservative, secure, etc).

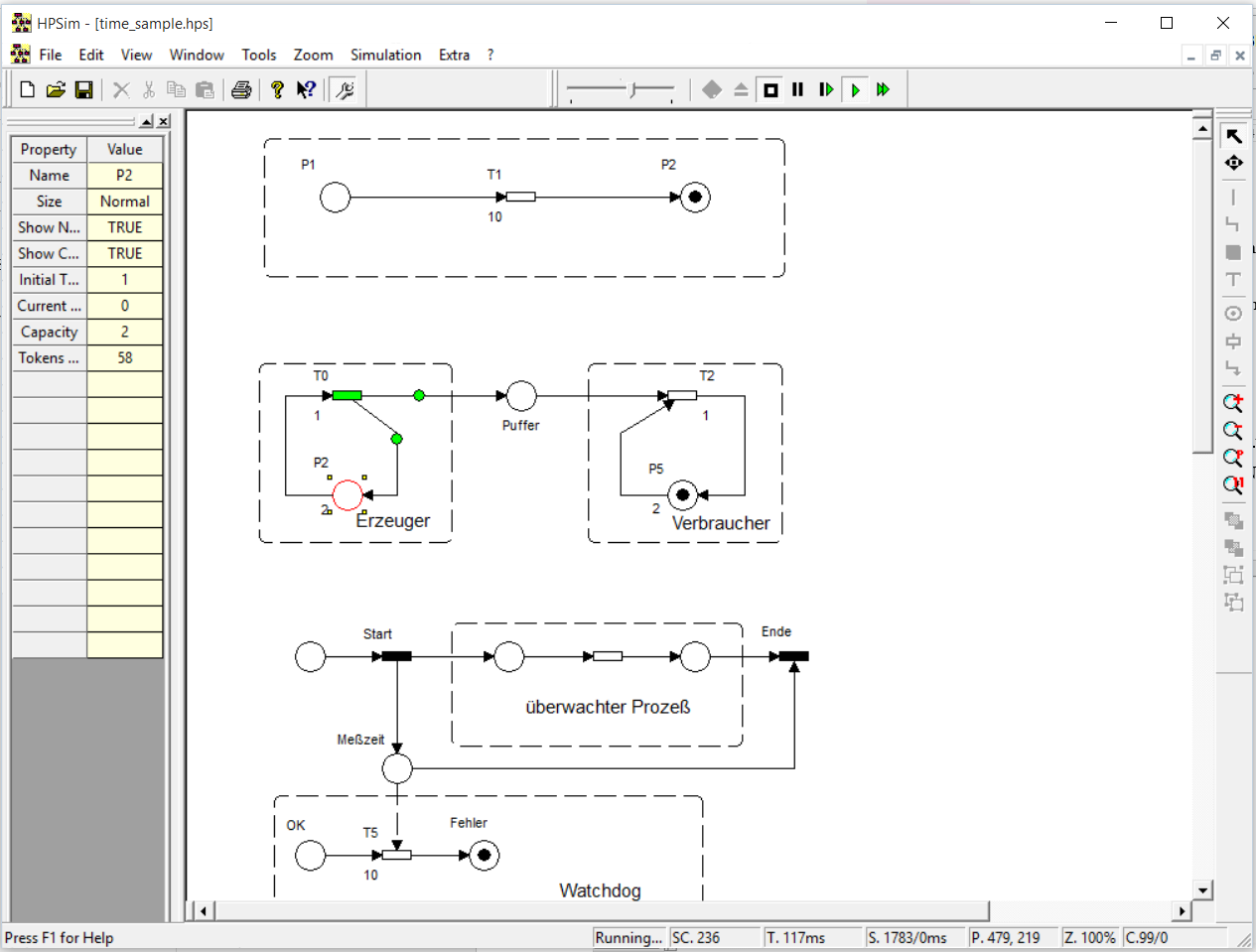


Figure 1 – The Hpsim simulation tool

**2nd Part – Control a carwash station with Petr-Nets**

The problem we are going to model consists of the development of a program to control the car wash station illustrated in fig. 2.

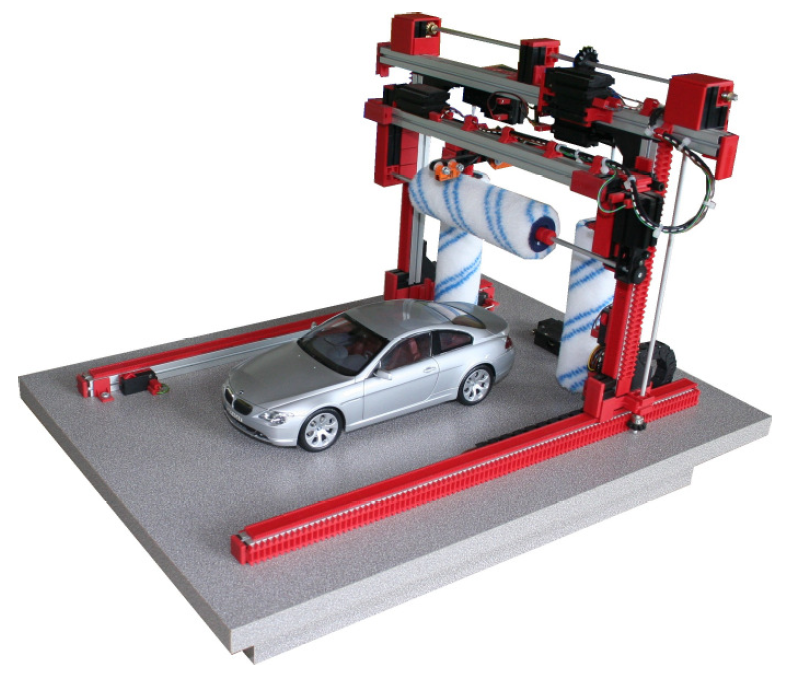


Figure 2 – The car wash station kit.

The execution of this work should be subject to the following steps:

* Study the requirements of the problem (partially defined below)
* Model the problem into a Petri net diagram.

The carwash management program must provide the following services to the costumers:

* Simple wash program (3 euros)
* VIP wash program (20 euros)
* Mechanism for emergency stop and resume

Use the PetriNetSim tool (fig. 3) for modeling a PN diagram that is able to control the car wash station. The software was modified to include a button (at the upper right side), which allows switching between using simulated or real car wash station.

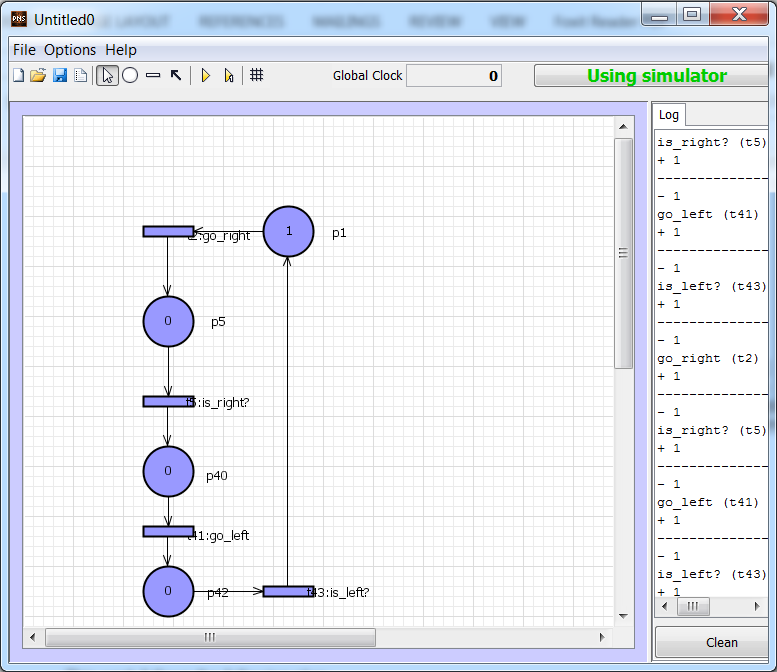


Figure 3 – PetriNetSim simulation tool

**3rd Part - Transforming PNs in corresponding LADDER diagrams**

In this part, we will learn how to transform PN in corresponding LADDER diagrams. A LADDER diagram can then be uploaded into a Programmable Logic Controller (PLC), which is prepared to execute LADDER.

A PLC is a device widely used in industrial processes. Typical examples of PLC utilization are in the automotive industry, chemistry, refineries, textiles, etc. By its nature, PLCs are suit well in situations characterized by concurrent real-time behaviors, which need to be coordinated.

The PLC can be programmed with LADDER diagram. However, some manufacturers provide PLCs that can be directly programmed with GRAFCET (graphe Fonctionnel of Commande, Etapes Transitions). GRAFCET is a graphical language, very similar the Petri nets, which allows modeling problems / systems with real-time characteristics.

The architecture for PLC-based operation is illustrated in fig. 4. In this figure, the (a) connection is used to download the LADDER diagram in the PLC, therefore, it is not permanent. It also allows monitoring (debug) the program downloaded in the PLC. Connection (b) wires the storage to the PLC and is permanent.



Figure 4 – Schematic of PLC connections: (a) Connection used to program the PLC; (b) Wires the storage to the PLC

Fig. 5 illustrates a LADDER diagram. In its basic form, each "Network" in the figure represents a logical expression which, depending on its evaluation, can be used to drive output loads, change the state of internal memory, activate timers and counters, call sub-routines, and so on.

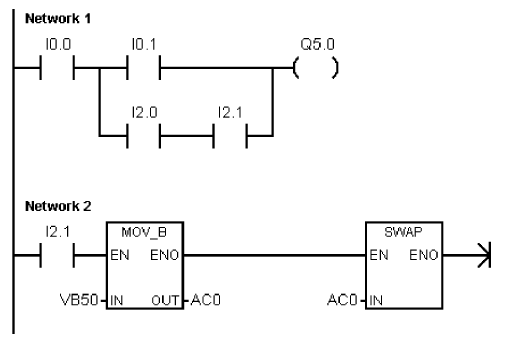


Figure 5 – LADDER diagram example

The development and subsequent download of the LADDER diagrams is done through the software Siemens S7-200 Micro/win, which is illustrated in fig. 6.

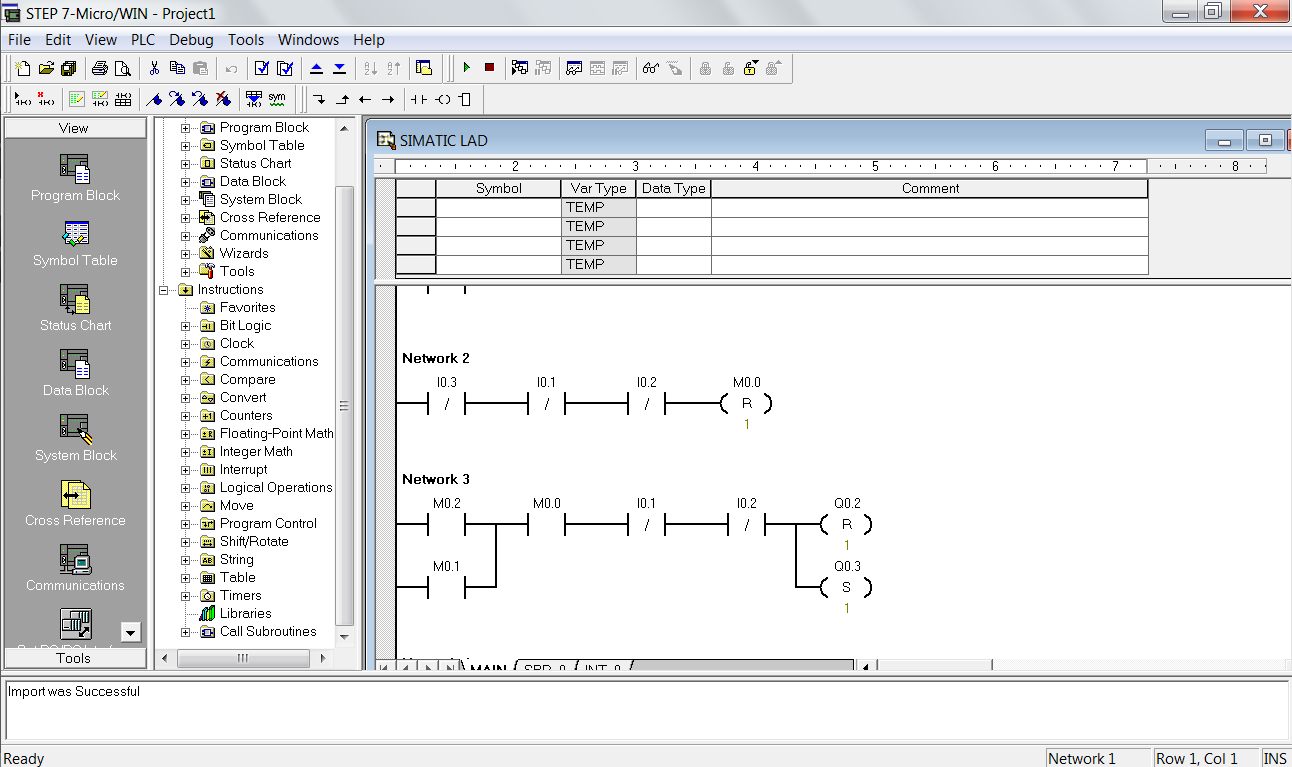


Figure 6 – The S7-200 Micro/Win software

As there is only one car wash station kit in the lab, a car wash simulator is provided, as illustrated in fig. 7. This simulator allows developing the proposed problem without the necessity to frequently test the solution with the real system.

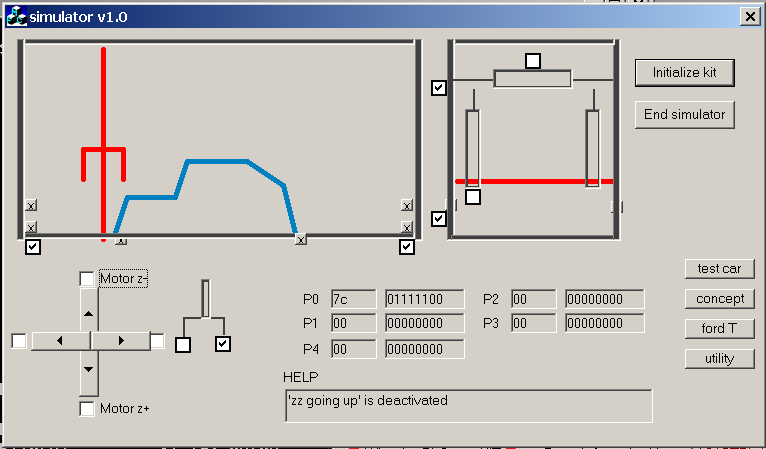


Figura 7 – carwash simulator

Additionally, a simulator of the PLC – the Siemens S7-200 – is also provided. This simulator has the ability to interact with the storage simulator (see button in Fig. 7). This way, the LADDER diagrams can be developed and simulated with the mentioned simulators, allowing integral simulation/testing prior to download in the real PLC. It is mandatory, that all the diagrams are firstly tested in the simulators. The developed diagrams can be downloaded to the real PLC only when the tests with the simulators are successful.

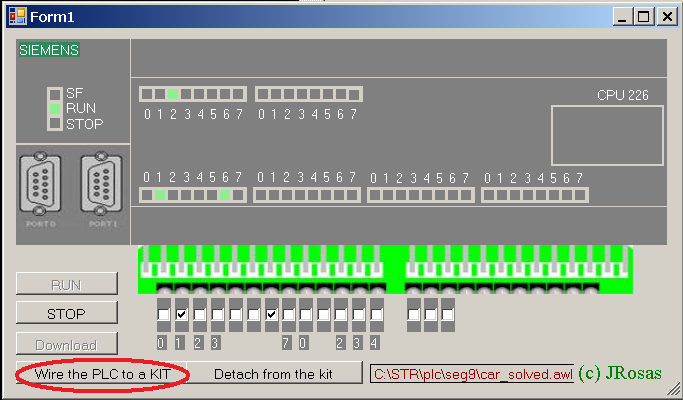


Figure 7 – The simulator of the Siemens PLC S7-200.

The requirements that need to be considered during these parts are in the table below.

|  |  |
| --- | --- |
| **Req.** | **Description** |
|  | **Practice on PetriNet models, using HPSIM. Take an image of each network and add it to the last sheet (the diagrams sheet), in this annex.** |
| R1 | Model and simulate a secure PN; a conservative PN; and a PN with capacity places and weighted arcs; and model a PN which starts working and enters in a deadlock state after a few iterations. PNs from theoretical classes can be used in this part. |
|  | **Model a PN to control the carwash station:** |
| R2 | Simple wash program |
| R3 | VIP wash program. |
| R4 | Emergency stop and resume button (the same button). |
|  | **Petri Net to LADDER transformation** |
| R5 | Transformation of a PN into a LADDER diagram |
| R5.1 | Simple wash operation |
| R5.2 | VIP wash operation |
| R5.3 | Emergency stop and resume button (the same button). |

This work follows the following plan:

1st class: Introduction to HPsim; Implementation of 1st part.

2nd class: Introduction to PetriNetSim. Implementation of the 2nd part.

3nd class: PLC programming. Implementation of the 3nd part.

# Annex 2 – Lab. Report

|  |  |
| --- | --- |
| **Report for Lab work nº 3** | |
| Disciplina | Real-time Systems |
| Year | 2015/2016 |
| Student Nº 40798  Student Nº 39656  Student Nº 37624 | Name: Lino Miguel Santos Estêvão  Name: Filipe Miguel Aleixo Perestrelo  Name: Ana Maria Aires Carreira |
| Due date: | 23 de Dezembro de 2015 |

Add or remove lines as you need during the filling of the following tables.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Functional requirements’ answers (check with “X”)** | | | | |
| **Req.** | **Success (100%)** | **Almost done** | **Unable to finish**  **but with some work done** | **Unable to do it** | **Professor’s review**  **(leave it blank)** |
| R1 | X |  |  |  |  |
| R2 | X |  |  |  |  |
| R3 | X |  |  |  |  |
| R4 | X |  |  |  |  |
| R5 |  |  | X |  |  |
| R6 | - |  |  |  |  |
| R7 | - |  |  |  |  |
| R8 | - |  |  |  |  |
| R9 | - |  |  |  |  |
| R10 | - |  |  |  |  |
|  | - |  |  |  |  |

**Diagrams sheet (relate each diagram to the corresponding functional requirement(s))**

|  |
| --- |
| Rq.2 (Com a activação das escovas em go\_right e posterior desactivação em go\_up)    Rq.3 (Vai para a direita, volta para a esquerda e repete o processo. As escovas são activadas em go\_right1 e posteriormente desactivadas em stop\_complete\_wash)    Rq.4 (POWER DOWN: Desactiva as escovas em go\_left\_stop e vai para a esquerda, terminando o programa; STOP: Pára tudo o que está a fazer guardando o estado em que se encontra em stop\_everything; RESUME: Na mesma rede de STOP, recupera o estado anterior e prossegue o que estava a fazer em resume\_everything)    Rq.1  Rede Segura    Rede Conservativa    Rede com pesos nos arcos e capacidades    Rede que entra em dead-lock    Rede que entra em conflicto (extra)    Rq.5 (Não foi possível por a funcionar)  Symbol table    Networks (Com comentários) |