Assignment 2 – Synchronised models, (bi)simulation

Note: the FSMs in this exercise have their states numbered. This is only a convenience so that it is easier to define the (bi)simulation relations, but these labels are not proper part of the models themselves.

Exercise 1 (1 points) Consider the example shown in class of two mutually similar, but not bisimilar, FSMs. Prove that simulate each other, but are not bisimilar.

Exercise 2 (4 points)
(a) Consider the following two models. The one on the left represents a simple loop, while the one on the right has been unwound one time, and then the loop continues. Prove that the two models are bisimilar.
(b) Prove that any finite number of unwindings results in a bisimilar model.

(c) Consider the following two models, where one part of the FSM has been duplicated in the second one. Prove that the two models are bisimilar.
(d) Prove that any number of finite copies results in a bisimilar model.

Exercise 3 (7 points)
Go to [http://nusmv.fbk.eu/](http://nusmv.fbk.eu/) and download the NuSMV tool (not NuXMV), a well known model checker. In this exercise, we will use the tool to develop a model. You will be asked whether to download the tool with ZCHAFF built in. For this exercise, it does not matter which version you download.
(a) Familiarise yourself with the language syntax, and learn how it works. Take a look at the provided examples.
(b) Model the microwave oven from the previous exercise and check through simulation that it behaves as expected.
(c) Model the following scenario. Be sure to use different modules for behavior that should be isolated. Once your model is complete, write a temporal logic formula that says that printed circuits are manufactured indefinitely (i.e., it is always true at any point in time that a circuit will be manufactured in the future).
**HINT:** check the property patterns presented in class. Write the property in NuSMV and have the tool check it for you.

A factory specializes in the manufacture of printed circuits. To achieve this, it employs two robots (R1 and R2), which start by working separately on different pieces which are later assembled together. Each robot picks up the next piece to work on from its own conveyor belt (C1 and C2). The robots use two different tools (T1 and T2) to work on the pieces, but these tools are shared between them. None of the tools may be used simultaneously by the two robots.

R1 picks up pieces from C1 and decides (we don’t know how) whether it needs to apply both tools or only T1. It picks up the necessary tools one by one, performs its job, drops the tools one by one and waits for the other robot to be ready.

R2 works similarly, picking up pieces from C2, but decides whether it needs to use both tools or only T2. Once both have finished with their pieces, they assemble it together and place it in the output conveyor belt. A human operator periodically picks up completed circuits from this belt, which can carry at most two complete circuits at any time (the robots must wait to assemble the circuit if the belt is full). Assume that both robots have an infinite supply of pieces to work on.