Assignment 2

[Working directory: **atm**]

> **Note:** When reading-in the SVA file (in MV mode), make sure to select SVA standard 2009 or later. Otherwise, the SVA keyword **implies** will not be recognized by OneSpin.

In this assignment you will become more familiar with the use of the language SVA in formal property checking.

Given: the SystemVerilog description of a **part** of an ATM controller (ATM = *Asynchronous Transfer Mode*, a telecommunication protocol). Its behavior depends not only on the current input but also on previous inputs. The controller's task is to classify incoming ATM messages (called *cells*) based on the results of a separate CRC checker (CRC = Cyclic Redundancy Check). ATM cells consist of a header, a CRC block and a data block. The error detection using CRC allows to detect and correct single-bit errors and also to detect multiple-bit errors.

The controller determines for each incoming ATM cell whether the cell needs to be corrected or whether it is to be dismissed. The controller receives information about the outcome of error detection from the CRC checker. (Note that the CRC checker is specified in a separate module which is not considered in this assignment.)

Design description (atm.sv):

In every clock cycle a new ATM cell is processed and examined by the error checker.

Input signals (binary, "**bit**"):

error_i:	set if the ATM cell is erroneous (single-bit or multiple-bit error)
multiple_i:	set if and only if the error is a multiple-bit error

Note: In case of a multiple-bit error both input signals are set.

Output signals (binary, "**bit**"):

correct_o:	ATM cell is to be corrected
dismiss_o:	ATM cell is to be dismissed

Fig. 1 shows the state transition diagram of the ATM controller. The labels at the edges of the graph denote sets of input and output value pairs in the following format:

error_i multiple_i / correct_o dismiss_o

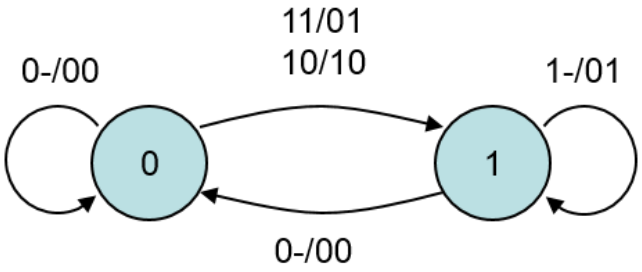

Figure 1: State transition graph of the ATM packet monitor

After reset, the design is in state 0.

Specification of the design

1. A cell is never corrected and dismissed at the same time.
2. An error-free cell is neither corrected nor dismissed.
3. All cells with multiple-bit errors are dismissed.
4. A first erroneous cell coming in is corrected if the error is a single-bit error and not a multiple-bit error.
5. A second erroneous cell is always dismissed.

Task 1

For each item of the specification above, formulate an assertion and prove it using **onespin**. For the last two specifications (4th and 5th), describe the properties using the SVA sequence logical operator **implies**. (How does **implies** differ from the temporal implications **|->**, **|=>**?) Analyze any counterexample found by the tool, and determine whether it is caused by an erroneous design or whether your property is not (yet) correct. Fix any errors.