

Master ESSV 2011-2012

Formal Design

ALTARICA Project

Alain Griffault
Bordeaux 1 University

This project is devoted to the modelisation of Stacks and FIFOs. The aim is to show that different implementations are bisimilar.

1 Project Requirements

The development shall be realized within the `arc` model-checker.

Project must be sent to `Alain.Griffault@labri.fr` by Friday, 9 december 2011. Send an archive of all your files.

2 Stacks models

In the `Stacks` directory, you will find:

- An `Alt` directory with ALTARICA codes describing two different implementations that are bisimilar.
- A `Spec` directory with `Acheck` and `Mec V` specifications files that define bisimulation relations between automata.
- Some latex files.
- A `GNUmakefile` script that you can use (try `make cleanAll` and `make all`) to generate results and pdf files.

The aim of this part is to modify the following files:

- `Stack1Array.alt`
- `Stack2Array.alt`
- `Stack3Array.alt`
- `Stack4Array.alt`
- `StackNArray.alt`

in order to obtain a bisimilar implementation of the node `Stack4Type`.

This implementation must reflect the classical implementation of a stack with an array and an integer pointing on the top of the stack.

3 FIFOs models

In the `FIFOs` directory, you will find:

- An `Alt` directory with `ALTARICA` codes describing two different implementations that are bisimilar.
- A `Spec` directory with `Acheck` and `Mec V` specifications files that define bisimulation relations between automata.
- Some latex files.
- A `GNUmakefile` script that you can use (try `make cleanAll` and `make all`) to generate results and pdf files.

The aim of this part is to create the following files:

- `Fifo1Array.alt`
- `Fifo2Array.alt`
- `Fifo3Array.alt`
- `Fifo4Array.alt`
- `FifoNArray.alt`

in order to obtain a bisimilar implementation of the node `Fifo4`.

This implementation must reflect one of the classical implementation of a FIFO with a circular array.