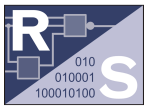
AMIDAR Simulator Quickguide

Lukas Jung (jung@rs.tu-darmstadt.de)
Christian Hochberger (hochberger@rs.tu-darmstadt.de)

TECHNISCHE
UNIVERSITÄT
DARMSTADT



Contents

1 Introduction

This document describes briefly how to use the AMIDAR simulator and the graphical debug tool within Eclipse IDE. Note that the graphical debug tool is limited to the synthesis of new Functional Units (FU) on a CGRA. The required software is:

- Java 8 JDK
- Apache Ant
- Eclipse Luna

For more detailed information on AMIDAR see `www.amidar.de`.

2 Installation

First install all software listed in Chapter 1. Then go through all following steps

2.1 Import the AMIDAR project into Eclipse

1. Download *Amidar.zip* from `www.rs.tu-darmstadt.de/fileadmin/Amidar.zip`
2. Unzip *Amidar.zip* in your desired location. This gets you a folder *<yourLocation>/Amidar*
3. Open Eclipse and right-click in the *Package Explorer* View and click *Import...*
4. Select *General→Existing Projects into Workspace*
5. Click *Browse...* and select the folder *Amidar* created in Step 2 as root directory
6. Select Amidar in the project list and click *Finish*

2.2 Install the Amidar-Debugger Plug-In [1]

1. Start Eclipse and go to *Help → Install new Software. . .*
2. Click on *Add. . . → Local...* and select the folder *<yourLocation>/Amidar/eclipse-debugger-updatesite*. Name the Repository
3. Make sure, that in the *Work with:* dropdown menu the created repository is selected.
4. In the main tree view, select the Amidar Debugger. If it does not show up, deselect *Group items by category*
5. Click *Next >* and complete the installation by acknowledging all dialogs.
6. Restart Eclipse.

2.3 Run the Simulator [1]

It is recommended, to use the simulator with two Eclipse windows on two screens (one for Java development, one for the Amidar debugger). To open a new window for an existing instance use *Window→New Window*. By using, *File → Exit* instead of the window decoration to quit the session, the window layout will be restored on the next launch.

1. Open the Amidar Perspective using *Window → Open Perspective*.
2. If not already happened, import the *Amidar* project into the workspace as described in Section 2.1
3. Create a Launch Configuration for the included Ant script:
   a) In the Package Explorer, right click on *build.xml → Run As... → Ant Build...*
   b) In the Targets tab, select only *compileall* as the executed target.
   c) Select *Apply* and *Run*.
   d) Verify that all required classes are compiled (build/api contains the Java api, build/apps contains the demos).
4. Select a global breakpoint in the Breakpoint view e.g. *Break after Step Zero* in the Amidar Perspective.
5. Launch an application from the *src/apps* folder by right click → *Debug As... → Amidar Simulator* (select the previously created Ant Launch Configuration, if asked). You can also select *Run As...* if no debugging is needed.

3 Using the Simulator

In this section a few hints are given how to use and configure the AMIDAR simulator.

- The simulator execution ist based on scripts. The standard script (*Amidar/scripts/simple.scr*) runs the desired application twice - first without synthesis and on the second run with synthesis. After each run statistics are reported via console output.
- Additionally you can easily activate or deactivate reports on different events in the script. For example if you want to know why a certain loop is not synthesized you can activate *simio* in order to obtain further information.
- In the script you can also define the CGRA which is used to synthesize new Functional Units. Several descriptions of CGRAs are located in *Amidar/cgra_instances*.
- Logfiles are created in the folder *Amidar/log*. The dot-files can be converted into graphics with the tool *Graphviz*. For Unix users the shell-script *Amidar/graph.sh* is provided. This script creates pdf-files from all dot-files and converts eps-files to pdf so that one can search the documents with *Ctrl-F*.
- You can run the simulator without Eclipse by calling *ant run* in a terminal in the *Amidar* directory. This will start the simulator using the application and the script defined in the file *Amidar/build.xml*. Note that you need Eclipse in order to use the graphic debug tool.

Remark: The AMIDAR Plugin is not yet adapted to the new Memory Interface of the CGRA. Thus, the tab *CGRA Memory* in the AMIDAR Eclipse View has no use.

4 Energy Consumption [2]

The energy model is embedded in the Amidar-simulator and estimates the energy consumption fully automatic during the simulation. This calculation works state-based and utilizes information about the state and operation of every module. Finally the overall consumption is totalized.

4.1 Properties

In the folder *energy_configuration* the following file can be edited in order to change the setup: *energyconfiguration_settings*

Entry	Meaning
General options:	
`default_zero_mode=true || false`	Activates the energymodel
`dynamic_consumption=true || false`	Calculation of dynamic consumption
`static_consumption=true || false`	Calculation of static consumption
Focus-options:	
`cgra_estimator=true || false`	NOT IMPLEMENTED
`module_focus=true || false`	Activates focus
`module_to_focus_on=javasim.components.IALU`	module to be profiled

4.1.1 Notes:

By activating the *module focus* the consumption of the module that should be profiled is calculated exclusively. Furthermore, *default_zero_mode* should only be active when simulations are running and one is not interested in powerconsumption. *default_zero_mode* deactivates the model and liberates processing power. This option might be important, if huge functionality tests are running.

Bibliography

[1] Sebastian Fahnenschreiber and Roman Ness. Amidar debugger, 2014.

[2] Dennis Wolf. Erweiterung des amidar simulators um ein energiemodell, 2015.