



*****Setting up New Software***

**Designing Embedded Processors with an Application Specific Instruction-Set**

This work uses ASIP Meister developed by PEAS Project, Osaka University.

5/8/2017

**Table of contents**

[1 Why to Update? 3](#_Toc482006701)

[1.1 Purchase of New ASIPmeister 3](#_Toc482006702)

[1.2 Goal of the Laboratory 3](#_Toc482006703)

[2 Installation 4](#_Toc482006704)

[2.1 Network Structure 4](#_Toc482006705)

[2.2 Text editors 4](#_Toc482006706)

[3 Integration of new ASIPmeister into the Lab 5](#_Toc482006707)

[3.1 The DLX architecture 5](#_Toc482006708)

# Why to Update?

The update is required based on the student feedback and complaints about the very old ASIPmeister tool.

## Problems

Following are the main problems that pushed to update the lab software:

1. ASIPmeister often crashes e.g. when Ctrl+C or Ctrl+V is pressed, and one has to repeat all the work again.
2. There is no copy-and-paste available. Students have to write almost similar text, which is tedious.
3. Scrolling is not available.
4. ASIPmeister required a separate retargetable compiler i.e. CoSy compiler that is installed on another machine.

## Purchase of New ASIPmeister

## Goal of the Laboratory

This laboratory will teach the creation of ASIPs from the design, over the high-level simulation to the final prototype on FPGA hardware. Benchmarks of speed, needed area and power/energy consumption will be performed and compared among different created ASIPs. For this purpose the usage of the different tools have to be practiced and the connection of these tools to form a tool chain has to be understood. The main goal is creating new ASIPs for special applications, to benchmark these ASIPs to find out their benefits and drawbacks and finally to interpret the benchmark results.

# Installation

This chapter explains the technical environment for the laboratory. This includes the usage of the computers and the directory structure for this laboratory. It is very important to understand completely the directory structure, as many scripts rely on this special structure and will not work at all or create an unexpected output if the directory structure is set up in a wrong way.

## Network Structure

## Text editors

Reference cards for vi and Emacs (or just Google for e.g. “Emacs reference filetype: pdf” if the links are no longer valid):

<http://www.digilife.be/quickreferences/QRC/Vi%20Reference%20Card.pdf>

<http://inst.eecs.berkeley.edu/~cs3/fa06/emacsreference.pdf>

# Integration of new ASIPmeister into the Lab

Dlxsim [DLX-Package] is an instruction accurate simulator for DLX assembly code. In this laboratory, we will use a modified version of dlxsim, which is changed in such a way, that it is behaving like the ASIP Meister specific implementation of the DLX Processor, which will be created and used in the later steps of the laboratory. In the first subchapter, some basic ideas about the DLX architecture and the DLX instruction set will be introduced. Afterwards the basic usage of dlxsim will be explained. In the last subchapter, it is shown how dlxsim can be extended to support new assembly instructions, which will be added to the DLX processor with ASIP Meister.

## The DLX architecture