Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institut für Integrierte Systeme
Integrated Systems Laboratory

Department of Information Technology and Electrical Engineering

## VLSI III: Test and Fabrication of VLSI Circuits

227-0148-00L

Exercise 4

# Tester Configuration

F. K. Gürkaynak
Prof. Dr. H. Kaeslin

SVN Rev.: 1880
Last Changed: 2017-03-02

**Reminder:**
With the execution of this training you declare that you understand and accept the regulations about using CAE/CAD software installations at the ETH Zurich. These regulations can be read anytime at `http://eda.ee.ethz.ch/index.php/Regulations`.

# 1 Introduction

In the last training you got an overview of the IC tester Advantest SoCV93000, and by now you should already have an idea how a complete test flow looks like. In this exercise you are going to start from scratch and have a look at the very first step which has to be performed for every test flow: the tester setup.

In particular, you will learn how to

- generate ATPG vectors using Tetramax and dump functional vectors from Modelsim,
- configure the tester channels (I/O setup),
- calibrate fixture delays,
- load testvectors into the tester and perform simple functional and ATPG tests.

# 2 Preparations

You can find detailed manuals for the tester on our eda-wiki[1]. To prepare for this exercise, you should read the following pages: *Functional Vectors*, *Tetramax*, *Tester I/O Setup*, *Simple Tests*. If you feel unsure about the basic handling you might want to brush up your memory by having a look at *Tester: Basic Handling* and *Tester Software*.

# 3 During the Training

**Student Task 1:** Preparations:

- Take ESD protection measures.
- Login to the hava workstation with your vlsi3 student account.
- Prepare the tester for testing with the `Incorruptus` load board and the `Standard QFN56 93000 (Feburary 2013)` test board.
- Run the install script to copy the data required for this training to your account. It will create a directory `ex_04` with all required files. You can also find in the folder the documentation of chip we are testing in this training.

```
sh> /home/vlsi3/ex_04/install_ex04
```

- Have a look at the `ex_04` folder. Also have a look at the datasheet of the chip we are going to test.

[1] `http://eda.ee.ethz.ch/index.php/Tester`

Before we work with the tester, we have to prepare the functional testvectors and ATPG patterns.

**Student Task 2:**

- Step into the `umcL180` cockpit folder and inspect the the modelsim and tetramax folders.
- Prepare the functional test vectors. The functional simulation has already been set up (see `compile.sh` and `sim.sh` scripts), but the vectors still need to be dumped and converted into the `.avc` format, such that they can be loaded into the tester. Follow the description in the *Functional Vectors* article on the DZ wiki.
- Prepare the ATPG vectors. An ATPG pattern generation script has already been prepared, but you still have to dump and convert the vectors. Follow the description in the *Tetramax* article on the DZ wiki.

You should always start with a very simple test flow (low frequency, simple vectors, no special operating modes, safe interface timings, safe levels, etc.). It doesn't make sense to directly start with more complicated tests like max. speed, power and parametric tests, since you should first check whether your chip gives any signs of life. Also, it is much easier to develop more elaborate tests if you have a working setup as a basis.

In this exercise, we are only going to set up the I/O configuration of the test flow, calibrate the fixture delays and instantiate the ATPG and functional tests. The voltage and timing configuration of the tester has already been performed.

**Student Task 3:**

- Start the tester software, and open the device `eva_device`.
- Open the I/O file `eva_io.csv` in the `dz_import` folder located in the device root, and complete it using the chip datasheet and testvecor file headers.
- Configure the I/Os and calibrate the fixture delays as described in the *Tester I/O Setup* article on the DZ wiki.
- After calibration, load the level and timing setups (`eva_levels`, `eva_timing`). If you adhered to the standard naming convention of the I/O groups, you should not get any errors in the console.
- Following the article *Simple Tests*, create a simple initial test flow. In case any test fails, go back to your settings and double check them[a]

[a] Note: if all your settings are correct, it could be that the currently loaded chip sample is defective. You may also run the basic tests with a different sample - but in this exercise, we know that the provided samples work. So you should look for issues in the tester setup.

**Student Task 4:** Cleaning up:

- Close all programs and properly turn off the tester.
- Log off from the workstation and clean up the workspace.
- Do not forget to remove the ESD strip from your feet!

## 4 Discussion

Discuss the following points with your assistant:

- Did you experience any pad/signal naming problems? What can you say about the choice and length of the signal names?
- Why is it important to first create a simple test flow with short functional vectors?
- In which step did you experience problems? What was the problem and how did you solve it?