**Tom Scherlis**

**Verilog stuff**

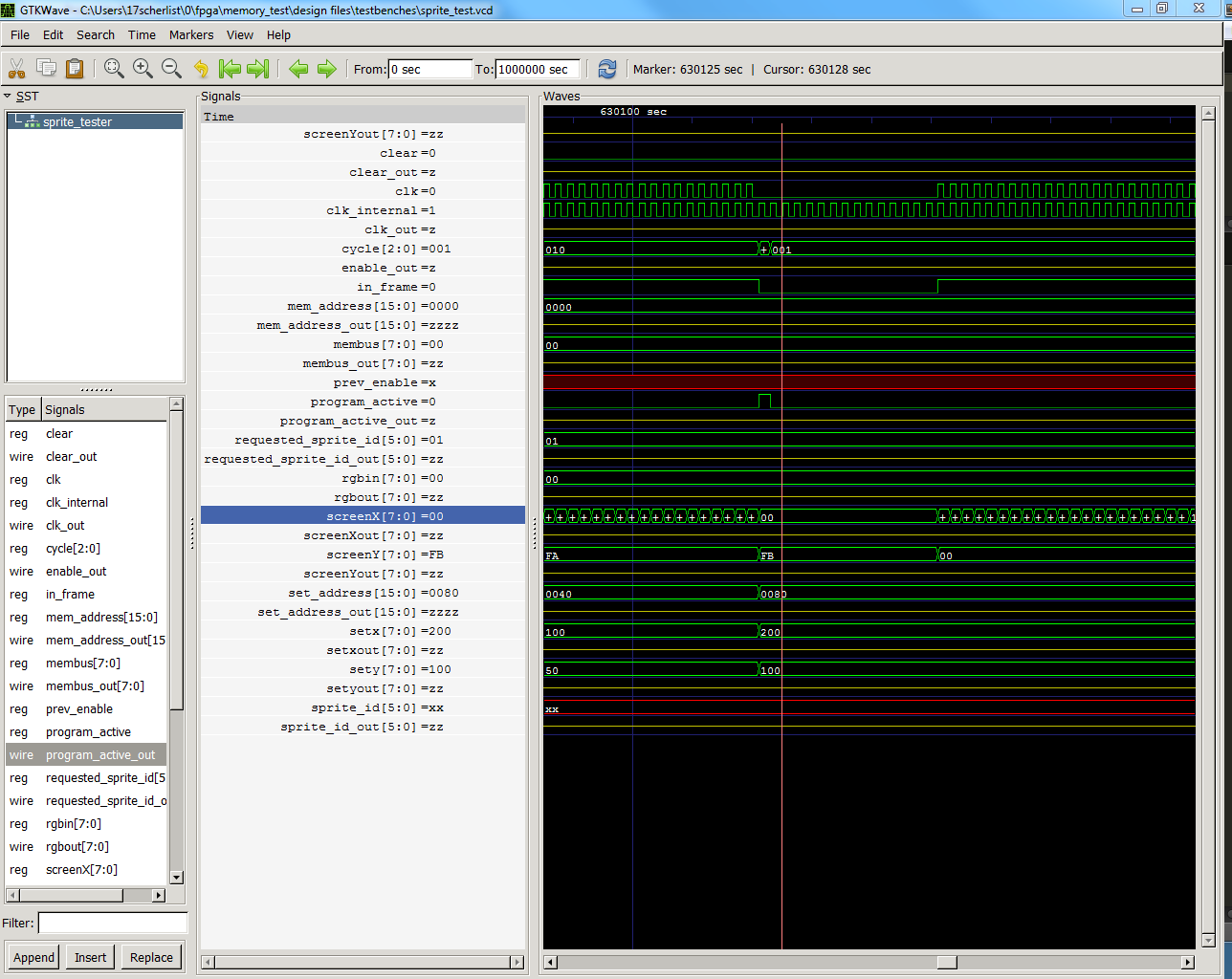
**Learning Testbenches:**

This week I started learning test benches after struggling to get the setup to compile without it. The advantage is that you can setup very specific test conditions and view the waveform output very quickly.

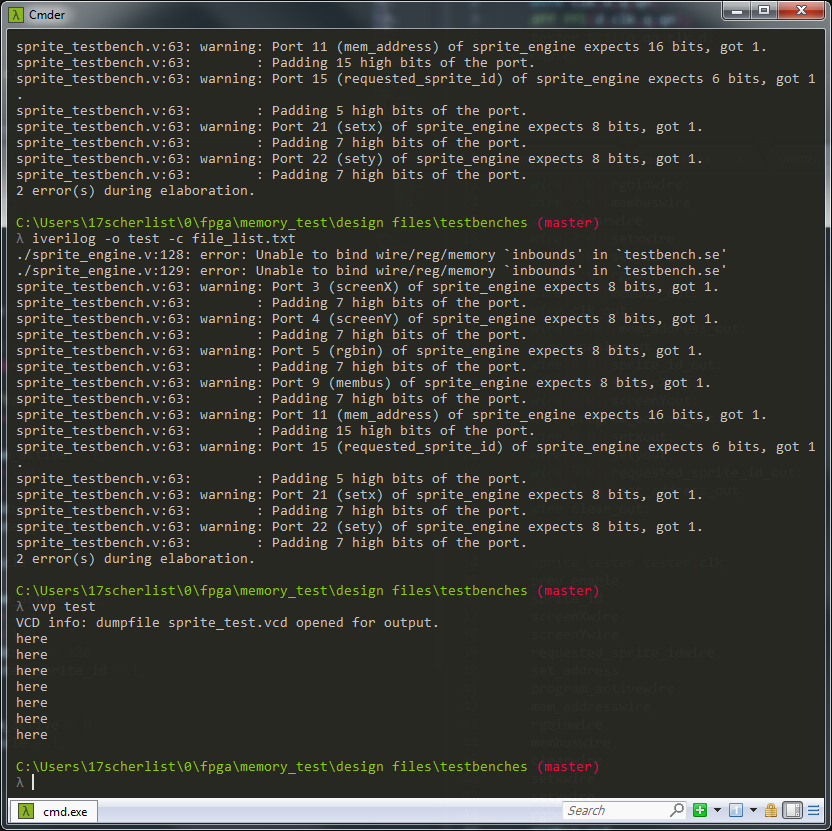
The testbench is composed of 3+ parts:

1. The module(s) you want to test
2. A tester module. This module gets extended Verilog syntax for time control, such as #*n* which is a *n* unit delay, *initial* which is a code block that runs on simulation start, as well as $ functions, including $display which prints a value, $dumpfile and $dumpvars which define the waveforms to dump, and $finish to stop the simulation.
3. A testbench module, which instantiates and connects the other two modules together.
4. A file list, which is a .txt file that tells the compiler which files to include in the build path. (optional)

I used Icarus Verilog – or iVerilog – to compile my code, and it has a command “vvp” to run a simulation and dump the waveform data to the file defined in $dumpfile. To view the dumped waveform, I installed GTKWave, and open source waveform viewer that can display the .vcd files.

Here are some screenshots: 1) The GTKWave output

2) The compiler output (On an error)



The current setup I have running is to test the sprite engine. My tester file, “sprite\_tester.v” simulates a working vga controller as well as another module that would program the sprites. As shown in the GTKWave output, it is generating all the waves properly. It is connected through “sprite\_testbench.v” to the sprite\_engine *se* but unfortunately there is a function in the sprite engine that refuses to compile.

In the next weeks I want to get the simulations to check out, so that I can maybe get some sprites to actually display. My ultimate goal for the end of the class right now is to build up a toolkit for Verilog, and do something with it after. Therefore I decided I should learn testbenches rather than just jumping through the program until it works.

I’ve attached the 3 relevant files: sprite\_engine.v, sprite\_tester.v, and sprite\_testbench.v.