FRONTGRADE

APPLICATION NOTE

Creating Certus-NX-RT Debug with ModelSim

3/18/2022

Version #: 1.0.0

Table 1: Cross Reference of Applicable Products

Product Name	Manufacturer Part Number	Device Type
Lattice Certus-NX-RT FPGA	UT24C407	RL01

1.0 Overview

This document details the process of debugging a **Certus-NX-RT FPGA** project using the **Lattice ModelSim.** For the purposes of this document, create a project named **counter_sim_debug** and configure **ModelSim** to include all the source modules required for a successful build. Using this template, projects are created using (a) the preferred application source directory structures and (b) the directory structure for the **Lattice Radiant-supplied files. Figure 1** shows the counter simulation.

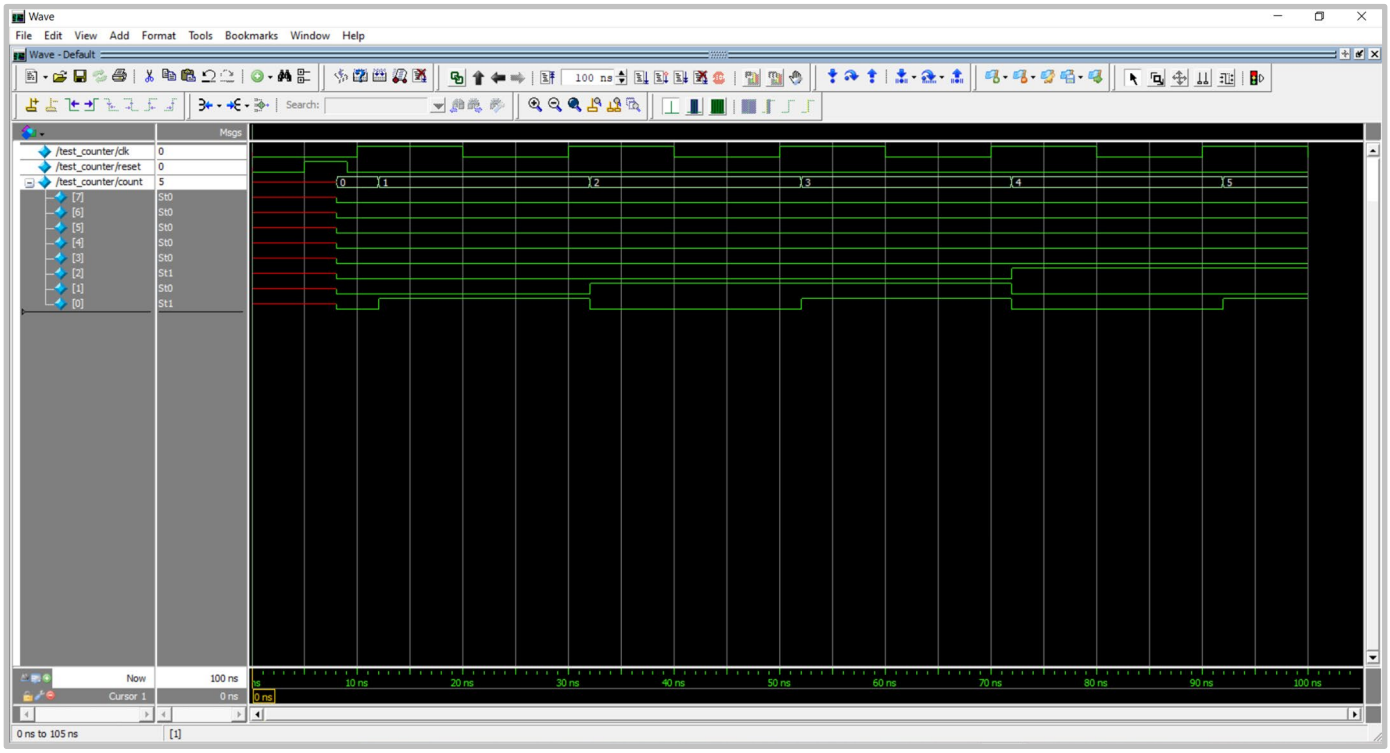

Figure 1: An Example ModelSim Wave

2.0 Creating a Project

1. Copy **Mentor Graphics** example files from **Lattice Radiant** installation directory:
   <installation dir>\modeltech\examples\tutorials\verilog\basicSimulation

   To:

   <project dir>\Lattice\applications

2. Launch **Lattice ModelSim**
3. Click **Jumpstart** to create a project, see **Figure 2**.

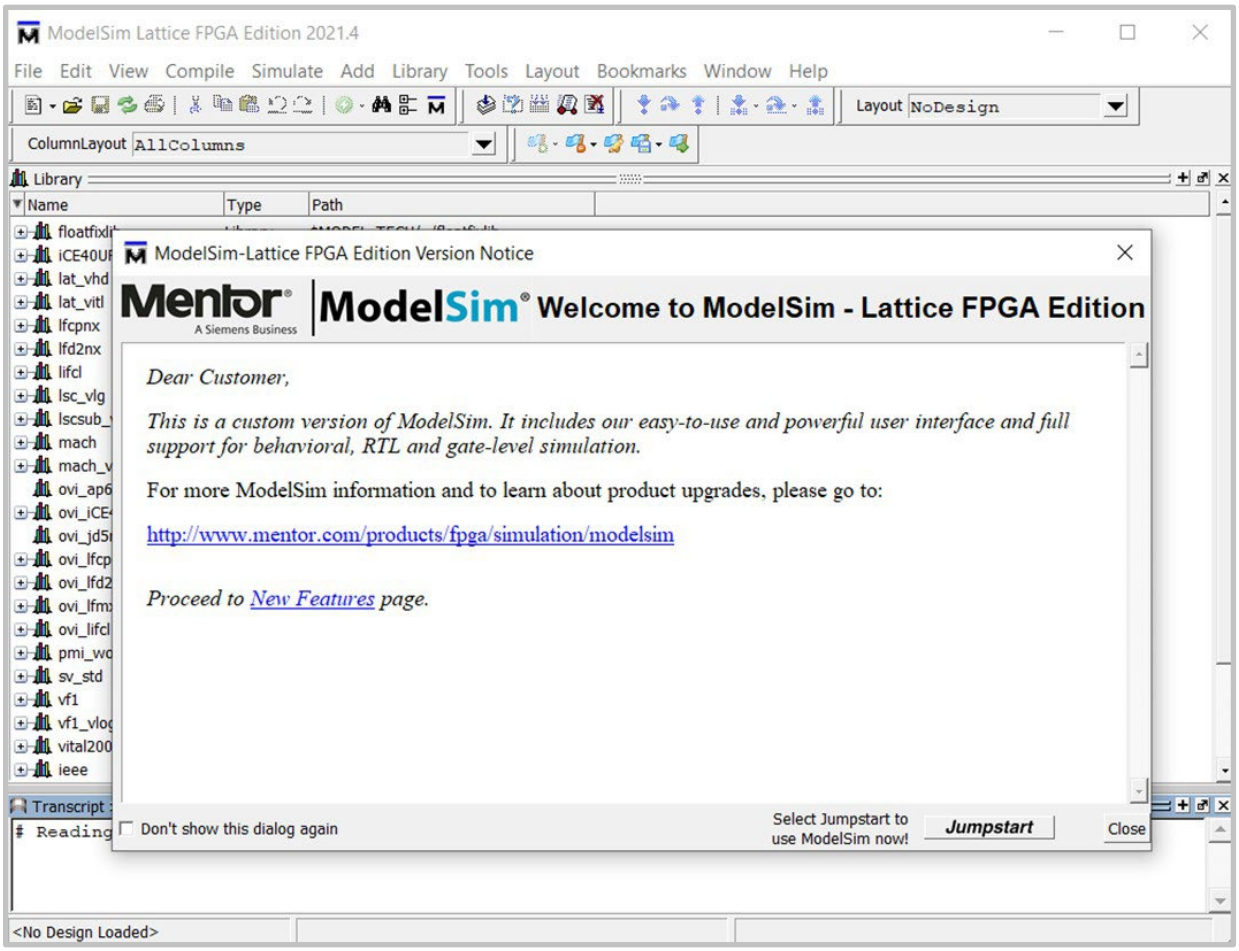

Figure 2: ModelSim-Lattice Project Creation

4. Specify the project name as **counter_sim_debug**, navigate to the **Project Location** of your choice and click **OK,** see **Figure 3**.

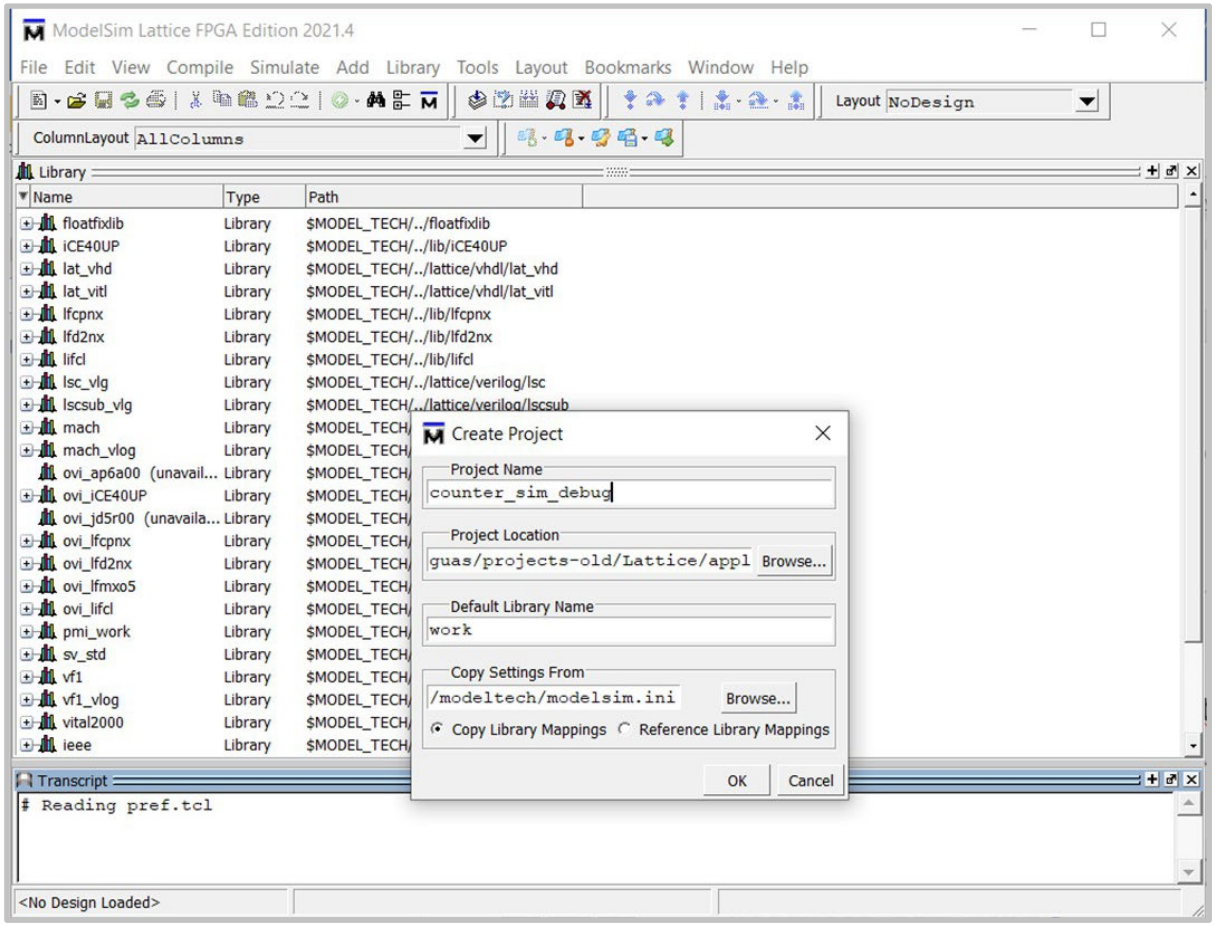

Figure 3: Naming the Project

5. Click **Add Existing File** and add **counter.v** and **tcounter.v** from <*project dir*>*Lattice**basicSimulation*, see **Figure 4**.

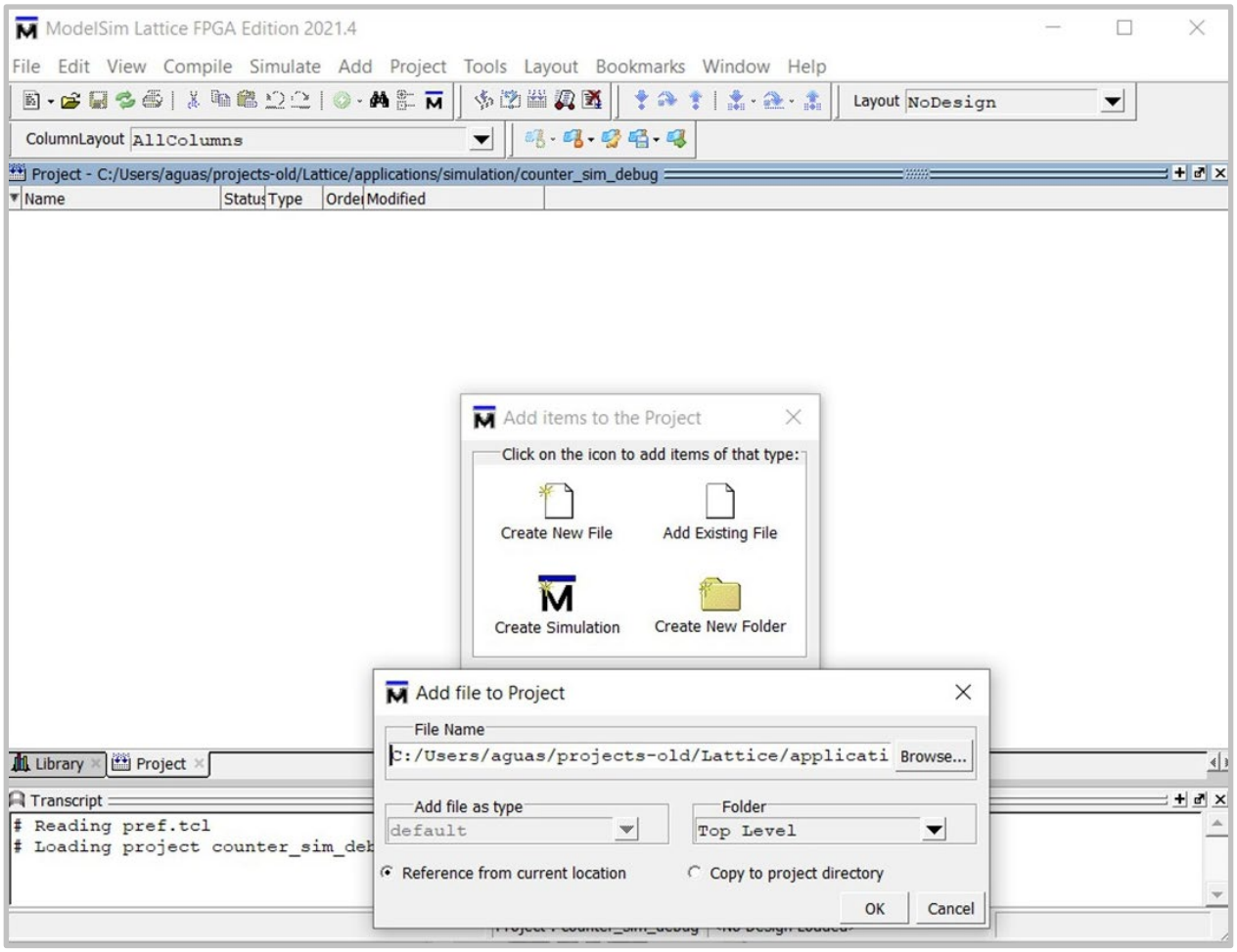

Figure 4: Adding Files to the Project

6. Form the top menu, select **Compile > Compile All** to compile **tcounter.v** and **counter.v**, see **Figure 5**.

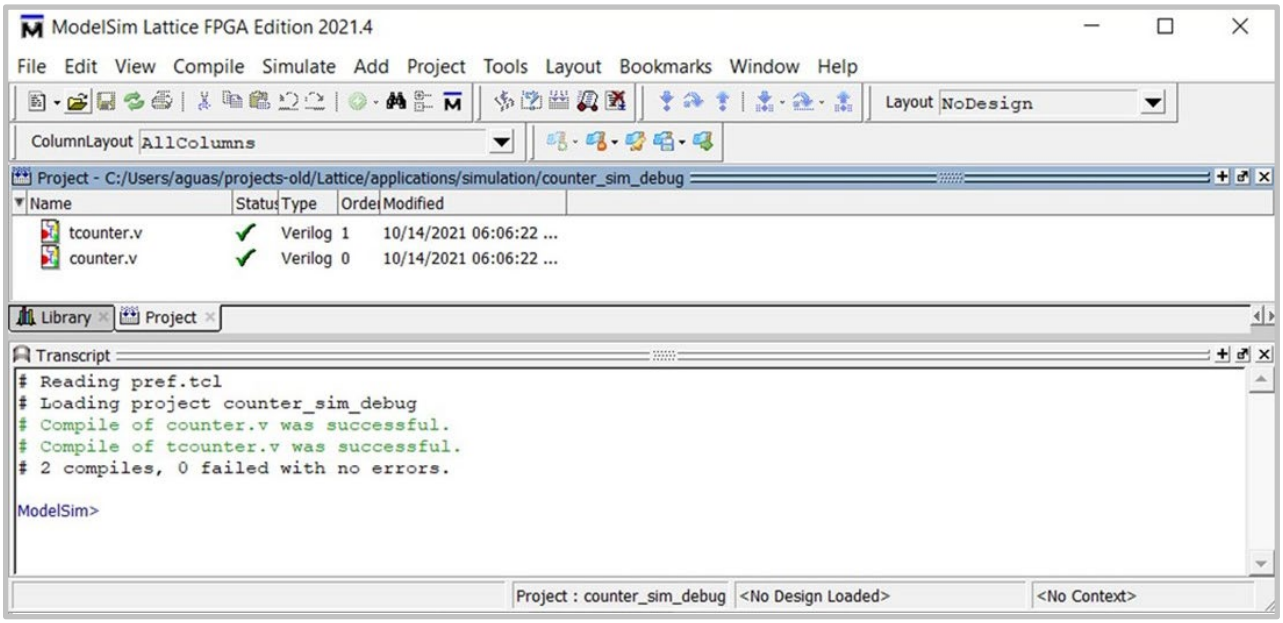

Figure 5: Compiling the Project

7. Form the top menu, select **Simulate > Start Simulation…** and select **test_counter.v** in the work library and click **OK,** see **Figure 6**.

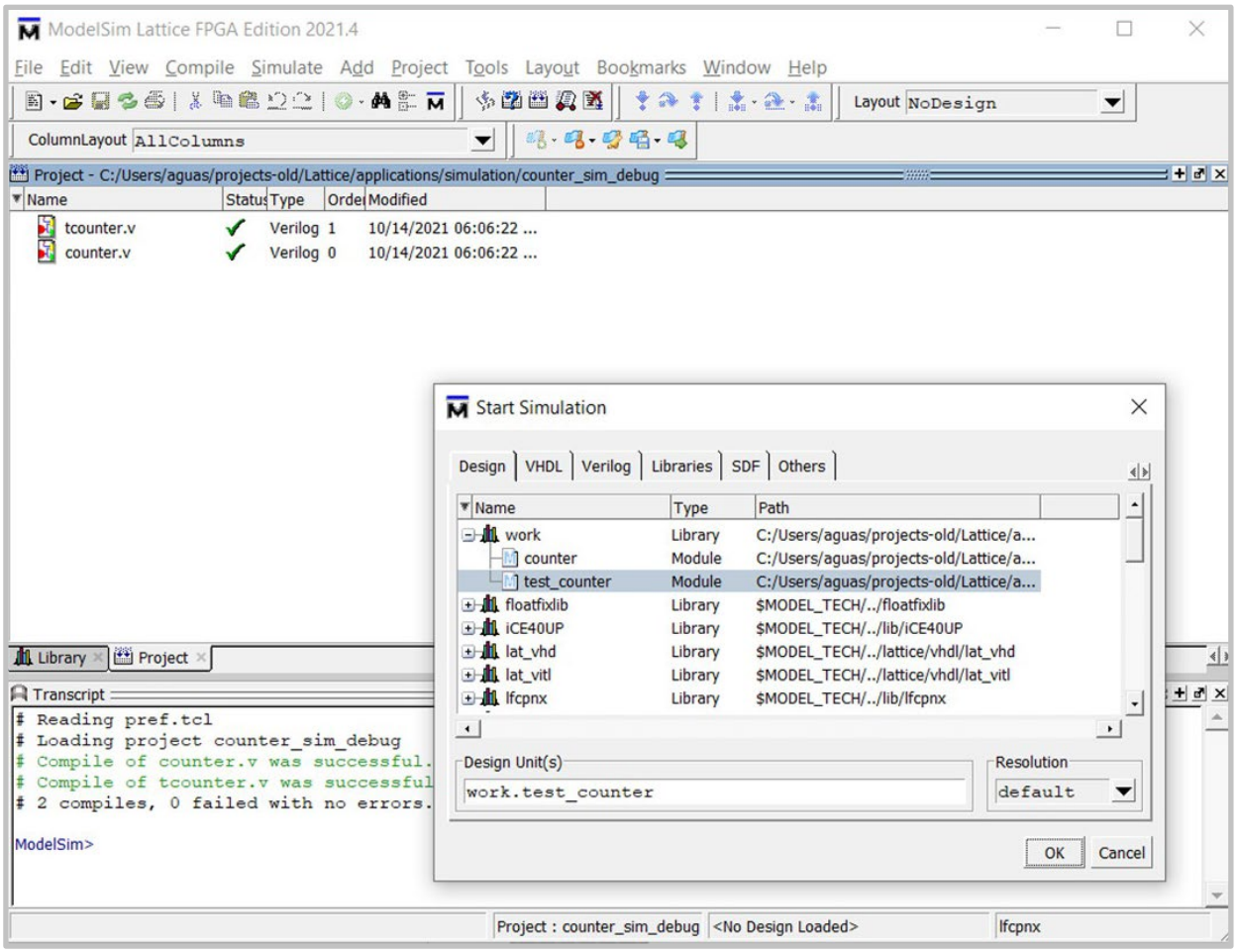

Figure 6: Starting a Simulation

3.0 Running the Simulation

3.1 Simulating Signals

8. From the **Objects** window, choose the top-level signals and drag them over to the **Wave** window. Run for 100 nanoseconds and the **Wave** window will display the timing diagram of the signals. Click **Zoom Full** to see the full picture, see **Figure 7**.

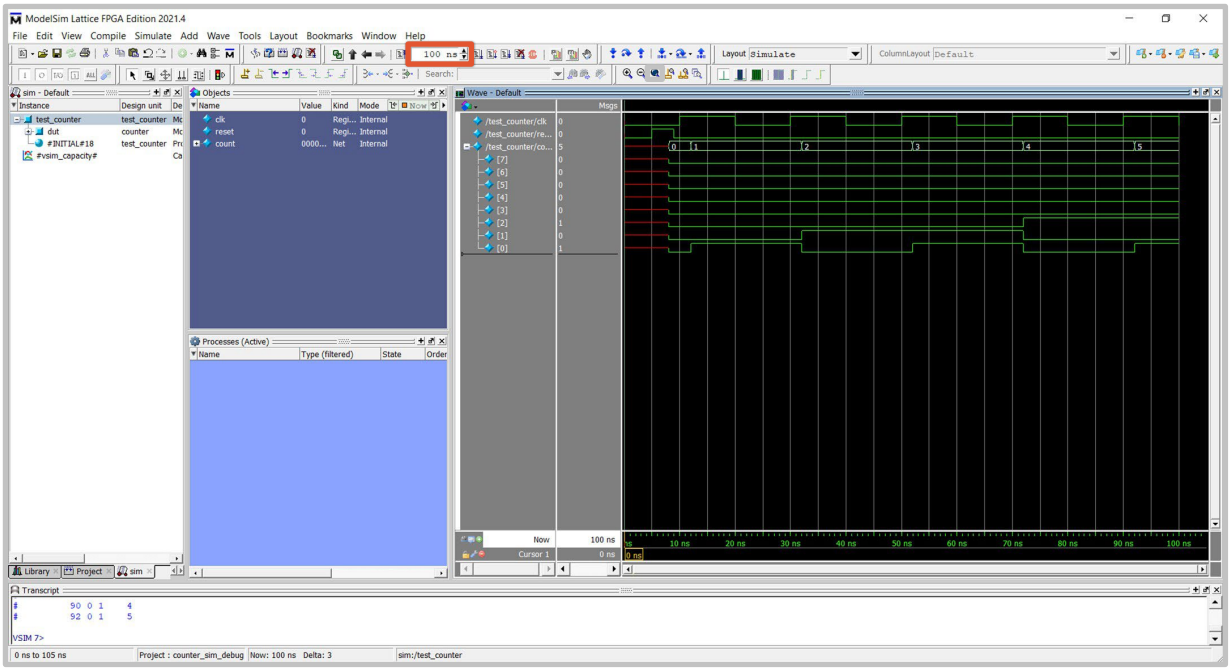

Figure 7: Zooming on the ModelSim Waveform

4.0 Debugging HDL Code

9. In the project window, right-click on **counter.v** and select **Edit,** the **counter.v** file opens next to the **Wave** window, see **Figure 8**.

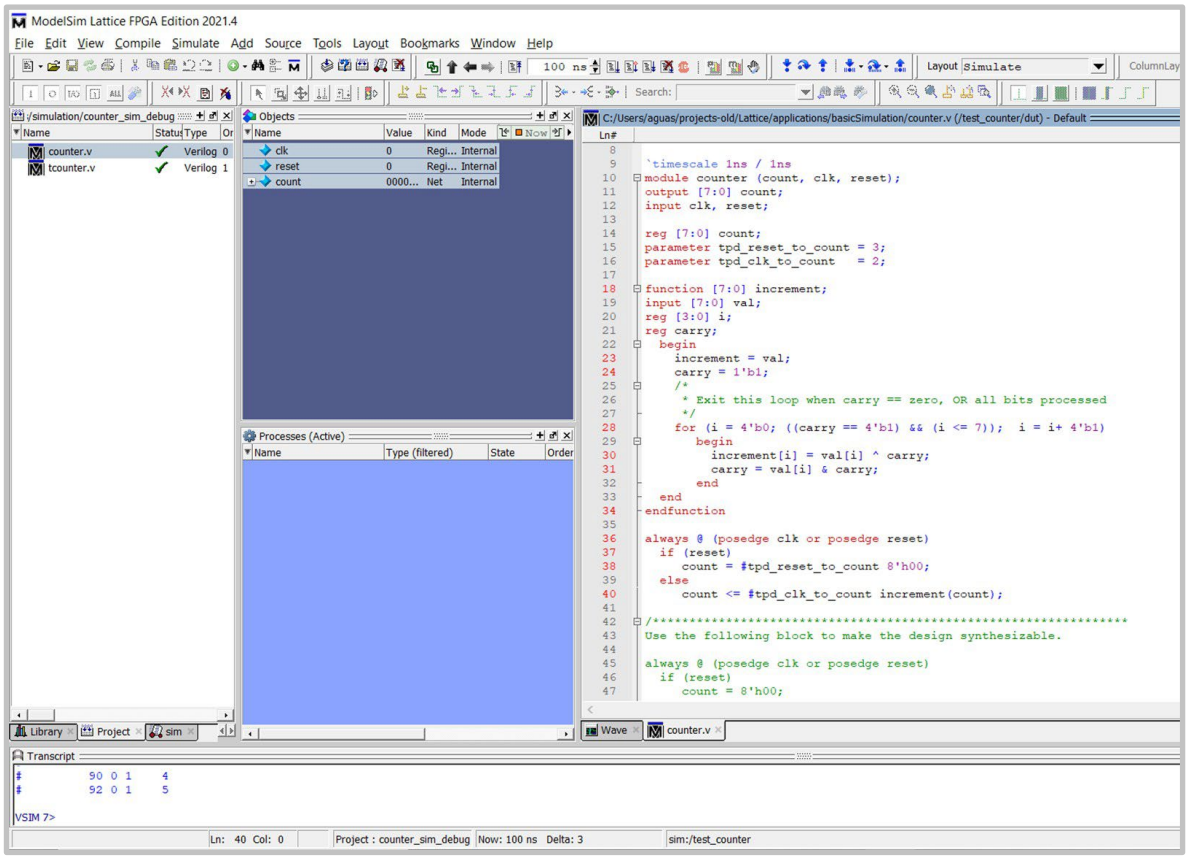

Figure 8: Debugging HDL Code

10. **ModelSim** allows to debug the **HDL** code with break points and stepping through the code. Double click on line 40 or right-click on it and select **Set Breakpoint;** the debugger sets a breakpoint at line 40, see **Figure 9**.

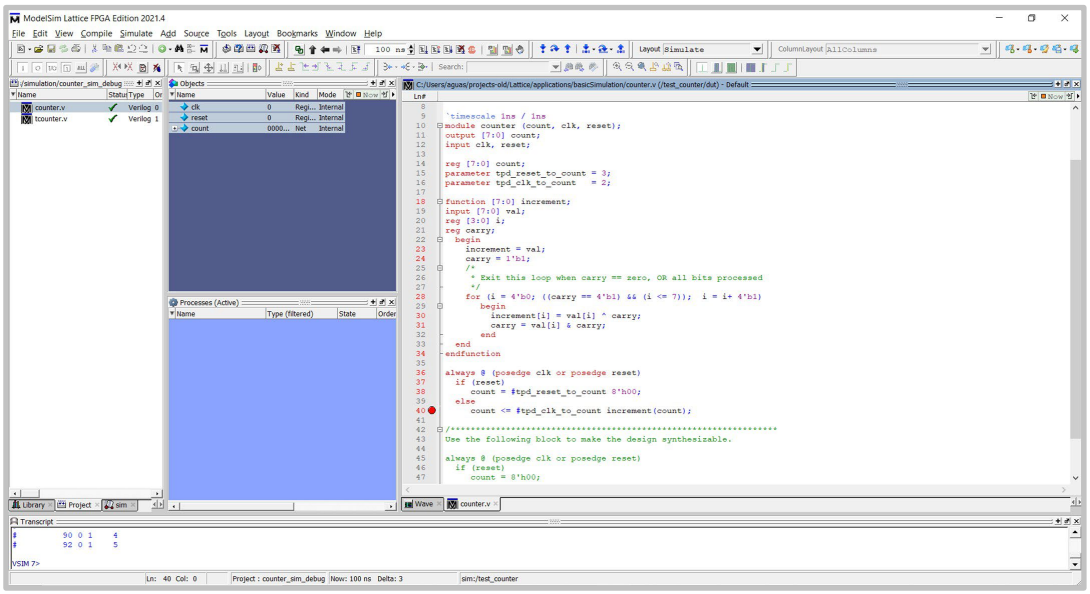

Figure 9: Setting a Breakpoint

11. Run for 100ns  and the debugger will stop at line 40; signal values can be seen in the Objects window. As with any other IDE running C code, you can **Step Into**  **, Step Over**  **,** and **Step Out**  **.** See **Figure 10**.

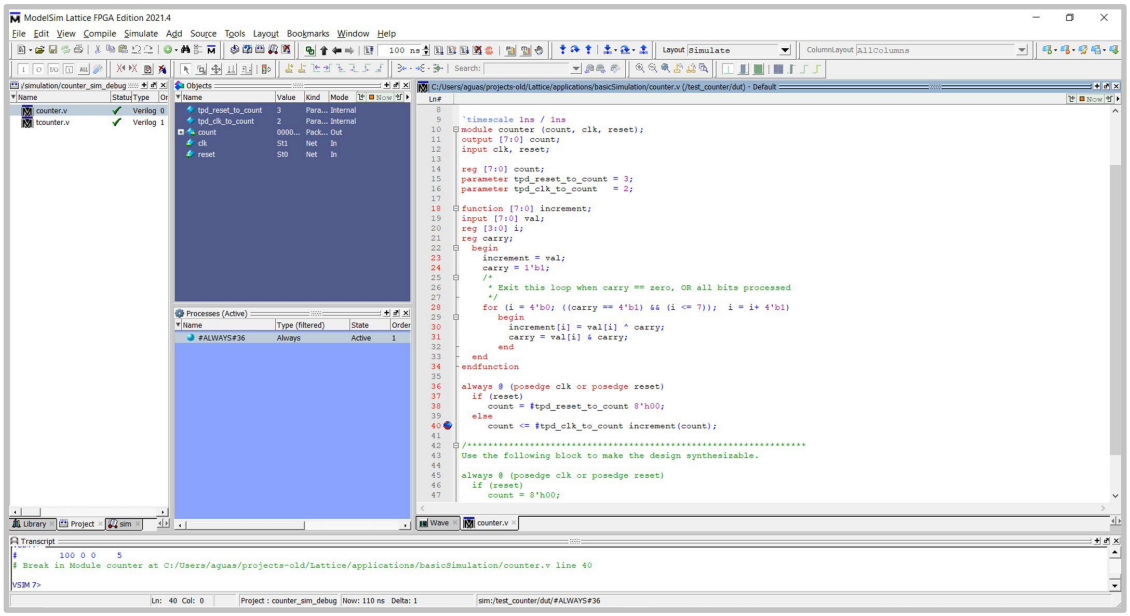

Figure 10: Stepping Around a Breakpoint

Revision History

Date	Revision #	Author	Change Description	Page #
3/18/2022	1.0.0	JA	Initial Release.	

Datasheet Definitions

	Definition
Advanced Datasheet	Frontgrade reserves the right to make changes to any products and services described herein at any time without notice. The product is still in the development stage and the **datasheet is subject to change**. Specifications can be **TBD** and the part package and pinout are **not final.**
Preliminary Datasheet	Frontgrade reserves the right to make changes to any products and services described herein at any time without notice. The product is in the characterization stage and prototypes are available.
Datasheet	Product is in production and any changes to the product and services described herein will follow a formal customer notification process for form, fit or function changes.