Contents

Appendix B VHDL Entry—Tutorial 2

This tutorial provides a step-by-step instruction for the VHDL entry, synthesis, and simulation of a 4-bit binary counter circuit. However, no knowledge of VHDL is required to follow this tutorial. Tutorial 3 in Appendix C will show how this circuit can be downloaded to the PLD on the UP2 development board so that you can actually see this circuit executing in the hardware.

This tutorial is very similar to Tutorial 1 in Appendix A for schematic entry. The main difference is in using the Text Editor rather than the Graphic Editor. The procedures for project creation, synthesis, and simulation are the same in both cases. Even if you do not intend to write VHDL code, you should go through this tutorial so that you can continue on to Tutorial 3 in Appendix C and learn how to download a circuit to the PLD on the UP2 development board.

B.1 Getting Started

B.1.1 Preparing a Folder for the Project

1. Each circuit design in MAX+plus II is called a project. Each project should be placed in its own folder, since the synthesizer creates many associated working files for a project. Using Windows File Manager, create a new folder for your new project. This tutorial uses the folder called `counter` created in the root directory of the C drive.

2. The VHDL source code for the counter circuit can be found on the accompanying CD-ROM in the file `counter.vhd` located in the directory `<CD-ROM drive>:\VHDL Examples\Appendix B VHDL Entry Tutorial 2\Source`. Using Windows File Manager, copy this file to the new folder `c:\counter` that you created in Step 1.

B.1.2 Starting MAX+plus II

After the successful installation of the MAX+plus II software, there should be a link for the program under the Start button. Click on this link to start the program. You should see the MAX+plus II Manager window, as shown in Figure B.1.

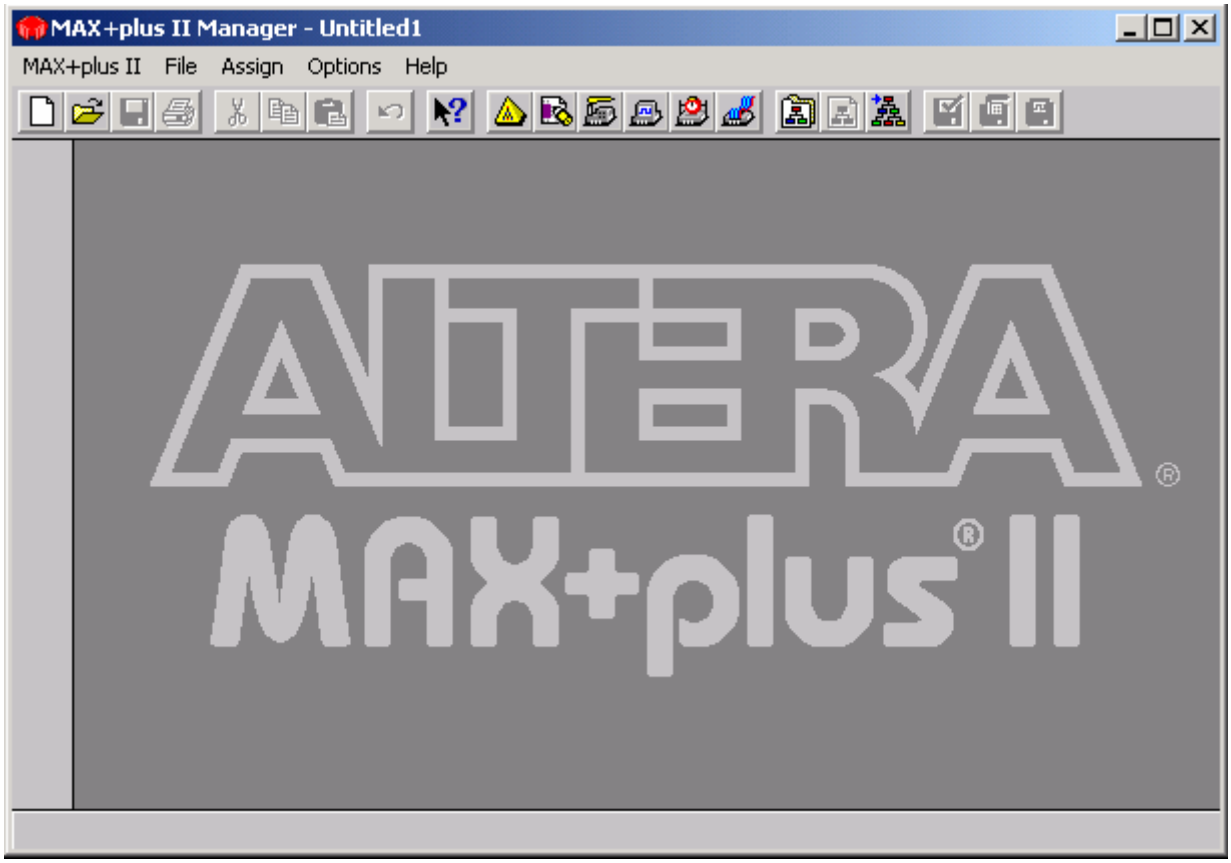

Figure B.1. The MAX+plus II Manager window.

Figure B.2 shows the toolbar for accessing the main development tools. The buttons from left to right are:

- *Hierarchy display*—to show the design files used in the current project
- *Floorplan Editor*—to map the I/O signals from the circuit to the pins on the PLD chip
- *Compiler (synthesizer)*—to synthesize the circuit to its netlist
- *Simulator*—to perform circuit simulation
- *Timing analyzer*—to perform circuit timing analysis
- *Programmer*—to program the circuit to the PLD chip
- *Open existing or new project*—to select a new project
- *Change project name to current filename*—to use the current file as the new project
- *Open top-level design file*—to open the top-level design file for the current project

Figure B.2 The MAX+plus II development software tool bar.

In the MAX+plus II software, different commands in the menus are available when different windows are activated. This might cause some confusion at first. If you cannot find a particular command from the menu, make sure that the correct window is the active window.

B.1.3 Creating a Project

1. From the Manager window menu, select File | Project | Name, or simply click on the icon  . You should see

the Project Name window similar to the one shown in Figure B.3.

2. Select the C drive from the Drives dropdown list.

3. Move to the `counter` directory on the C drive and double-click on it. You should see the file `counter.vhd` listed in the Files box.

4. Select the file `counter.vhd`. The filename will be copied to the Project Name text field.

5. Click OK. The MAX+plus II Manager window title should now show `c:\counter\counter`.

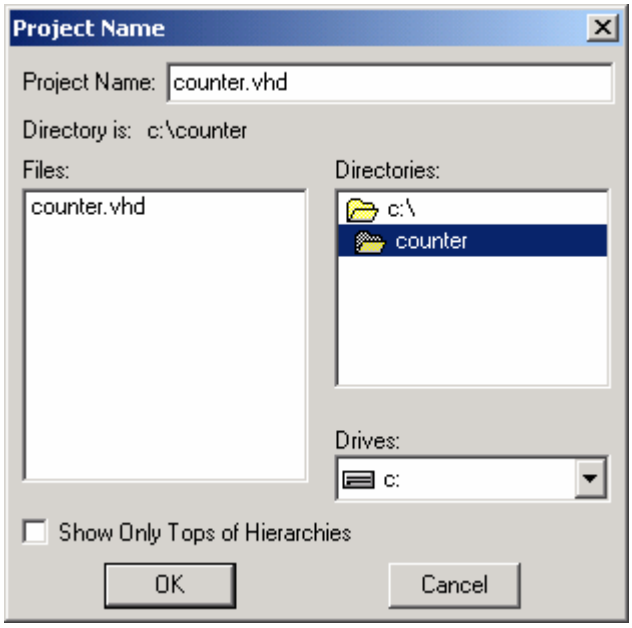

Figure B.3 Project name window for creating a new project.

- Alternatively, you can open any VHDL entity source file (with the extension `.vhd`) using the Manager menu command File | Open. With the VHDL entity source file in the active Text Editor window, select File | Project | Set Project to Current File, or click on the icon  to make that particular file the top-level project file. Note that the name of the file must be the same as the name of the entity that is in this file.

B.1.4 Editing the VHDL Source Code

- From the Manager window menu, select File | Hierarchy Project Top, or click on the icon  to open the VHDL source code for the counter. Notice that the entity name for this circuit is also **`counter`**. The top entity name for the project must be the same as the project name and the file name.

- You can use this Text Editor to modify the code if necessary. For this tutorial, we will not make any modifications, so you can close this Text Editor window.

- If you need to create a new VHDL source file, select File | New from the Manager window menu. Select Text Editor file, and click OK.

B.2 Synthesis for Functional Simulation

1. From the Manager window menu, select MAX+plus II | Compiler, or click on the icon  to bring up the Compiler window.

2. From the Compiler window menu (that is, with the Compiler window selected as the active window) select Processing | Functional SNF Extractor so that a check mark appears next to it. To actually see whether there is a check mark or not, you need to select the Processing menu again. The Compiler window for functional extraction is shown in Figure B.4.

3. Click on the Start button to start the synthesis. You will then see the progress of the synthesis.

4. At the end of the synthesis, if there are no syntax errors, you will see a message window saying that the compilation was successful. Click OK to close the message window.

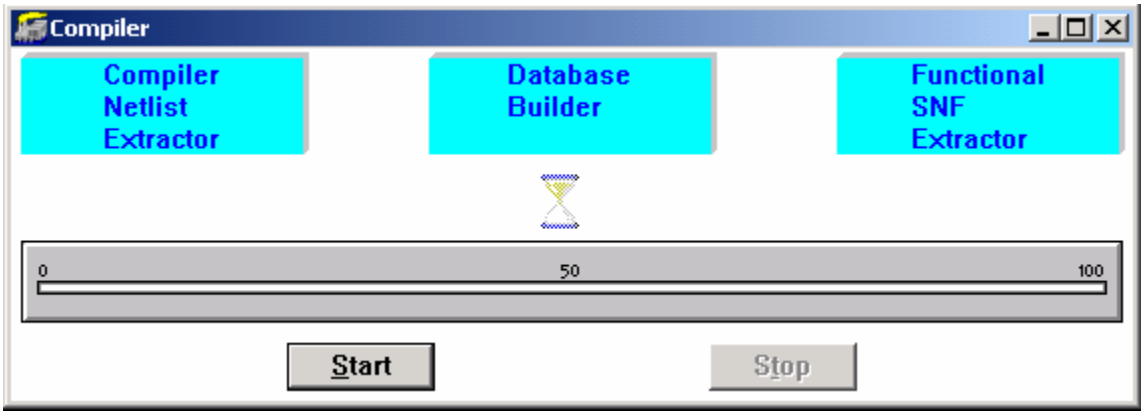

Figure B.4. Compiler window for functional extraction.

B.3 Circuit Simulation

B.3.1 Selecting Input Test Signals

1. Before you can simulate the design, you need to create test vectors for specifying what the input values are. From the Manager window menu, select MAX+plus II | Waveform Editor.

2. From the Waveform Editor window menu, select Node | Enter Nodes from SNF. You can also right-click under the Name section in the Waveform Editor window, and select Enter Nodes from SNF from the pop-up menu. You will see something similar to the Enter Nodes from SNF window shown in Figure B.5.

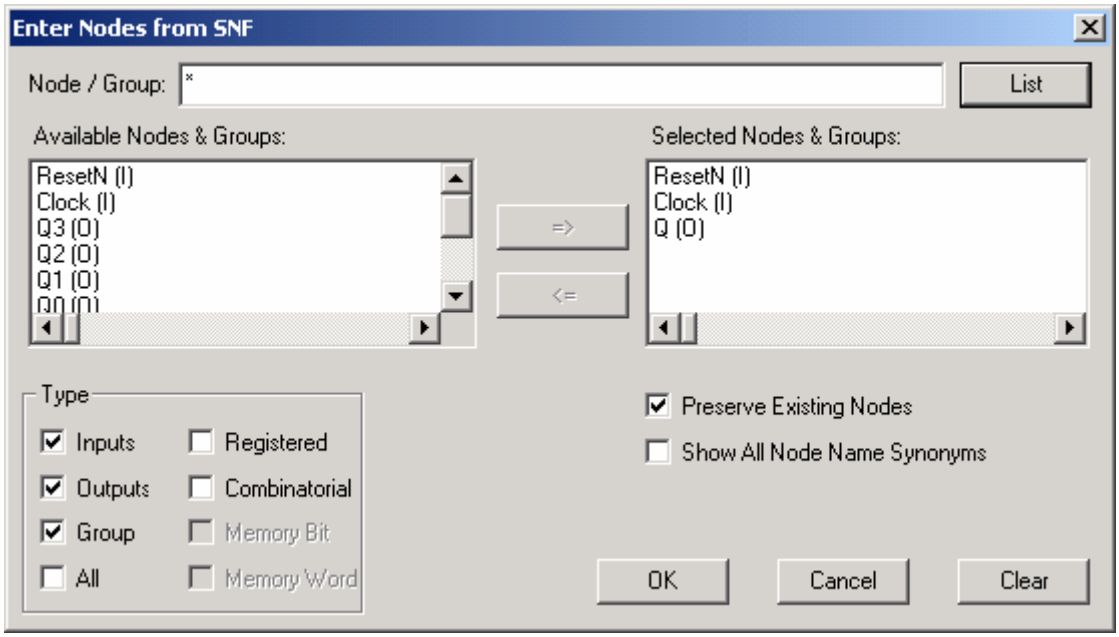

Figure B.5 Window for adding signals for simulation.

3. Click on the List button in the Enter Nodes from SNF window, and a list of available nodes and groups will be displayed in the Available Nodes & Groups box.

4. Select the signals that you want to see in the simulation trace. The signals that we want are: `ResetN (I)`, `Clock (I)`, and `Q (O)`. Be careful that it is `Q`, and not `Q0`, `Q1`, `Q2`, or `Q3`. The letters `I` and `O` in parenthesis next to each signal denote whether the signal is an input or output signal, respectively. Note that the signal name, such as **Q3** is bit three of the bus named **Q**. Multiple nodes can be selected by holding down the Ctrl or Shift key while clicking on the signal names.

5. After selecting the signals, click on the => button to move the selected signals to the Selected Nodes & Groups box.

6. Repeat Steps 4 and 5 until all of the signals that you want to see in the simulation are moved to the Selected Nodes & Groups box.

7. Click on OK when you are finished. The selected signals will now be inserted in the Waveform Editor window similar to Figure B.6.

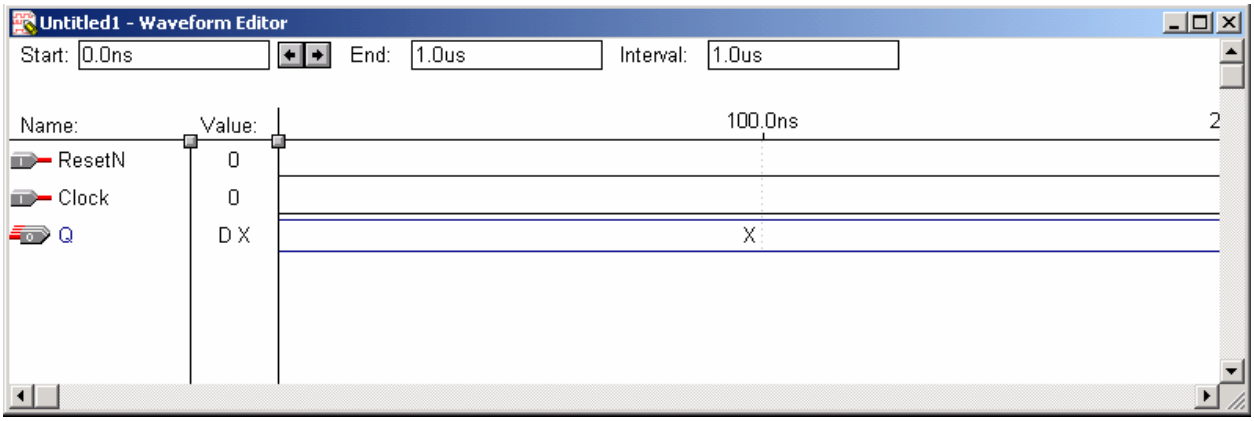

Figure B.6 Waveform Editor window for simulation.

B.3.2 Customizing the Waveform Editor

1. You can rearrange the signals in the Waveform Editor by dragging the signal icons like  up or down. Drag the `Clock` signal to the very top of the list.

- To delete a signal in the Waveform Editor, just select the signal by clicking on its name and press the Delete key.

- For signals that are composed of a group of bits (such as the counter output `Q`), you can separate them into individual bits or change the radix for the displayed value by first selecting that signal and then right-click the mouse. A drop-down menu appears. Select Ungroup to separate the bits. To regroup them, select the bits you want to group and then right-click the mouse. A drop-down menu appears. Select Enter Group. Type in a group name, and select the radix you want for the display.

2. We want to simulate for 2 microseconds. To change the simulation end time, select File | End Time from the Waveform Editor window menu.

3. In the End Time window, type in `2us`, and click OK.

4. To fit the entire simulation time range inside the window, select View | Fit in Window from the Waveform Editor window menu, or click on the icon  in the toolbar on the left. Your Waveform Editor window should now look like the one in Figure B.7.

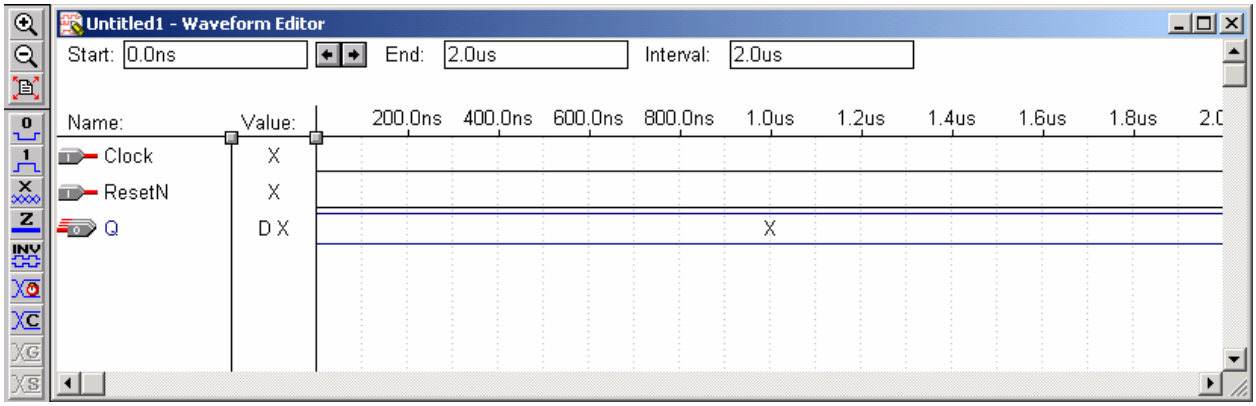

Figure B.7 Waveform Editor window after rearranging the `Clock` signal and fitting the entire time range inside. Notice also the toolbar buttons on the left.

B.3.3 Assigning Values to the Input Signals

The next thing is to assign values to all of the input signals.

1. Select the `Clock` signal by clicking on the signal name.

2. Click on the icon  in the toolbar on the left to define the `Clock` signal.

3. Click on OK to set the clock pulse.

4. Select the `ResetN` signal. The `ResetN` signal is active-low (i.e., a 0 value will enable the signal).

5. Click on the icon  in the toolbar on the left to set the signal to a 1 value.

6. Drag from time `1.2us` to `1.4us` for the `ResetN` signal only, as shown in Figure B.8.

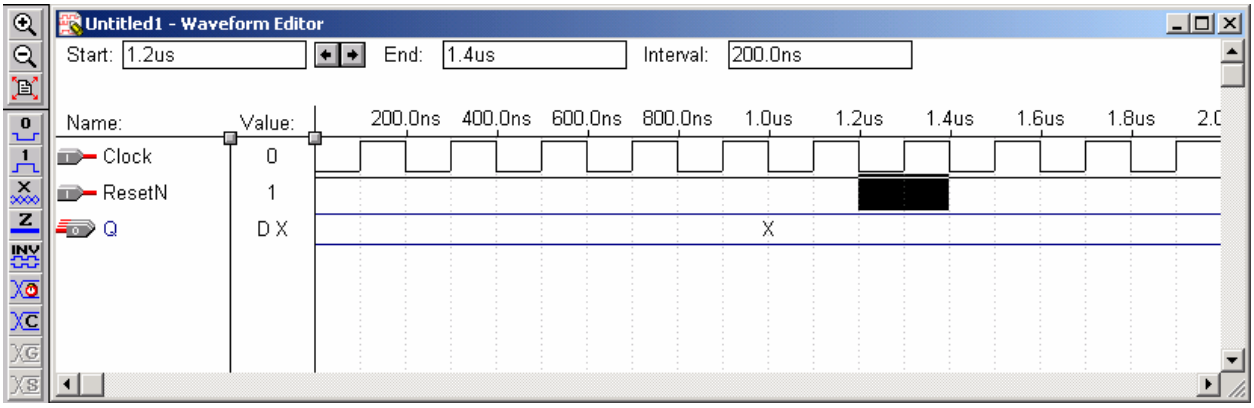

Figure B.8 Changing the Reset signal value between times 1.2 μs and 1.4 μs.

7. Click on the icon  in the toolbar on the left to set the signal in this selected time range to a 0 value.

B.3.4 Saving the Waveform File

1. Save the Waveform Editor window by selecting File | Save. The Save As window appears. Notice that the default file name is the same as the top-level entity name, and the extension is `.scf`. For this example, the name is `counter.scf`.

2. Click on OK to save the file.

B.3.5 Starting the Simulator

1. We are now ready to simulate the design. From the Manager window menu, select MAX+plus II | Simulator, or click on the icon  to bring up the Simulator window.

- You can also save the waveform file and start the simulator in one step by selecting from the Manager window menu File | Project | Save & Simulate.

2. The Simulator window, as shown in Figure B.9, is displayed. Make sure that the Simulation Input filename is `counter.scf`. This is the same name as your top-level entity.

3. Click on the Start button and watch the progress of the simulation.

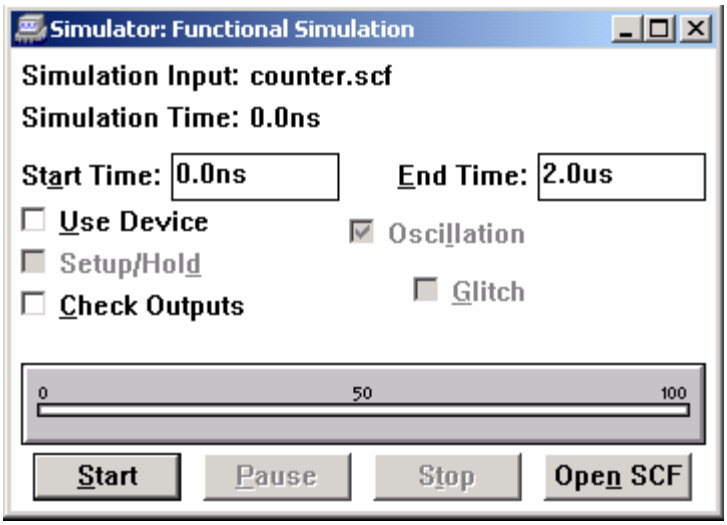

Figure B.9 Simulator window for the counter design.

4. At the end of the simulation, if there are no errors, you will see a message window saying that the simulation was successful. Click OK to close the message window.

5. Click on the Open SCF button in the Simulator window to bring up the Waveform Editor with the resulting simulation waveforms. The simulation result is shown in Figure B.10. The signal Q is the counter output.

Notice that when **ResetN** is de-asserted (va1ue 1), **Q** increments at the rising edge of each clock cycle. At 600 ns, the count is at 3. When **ResetN** is asserted at 1.2 μs, **Q** is immediately reset to 0. When **ResetN** is de-asserted again at 1.4 μs, the count starts again at the next rising clock edge.

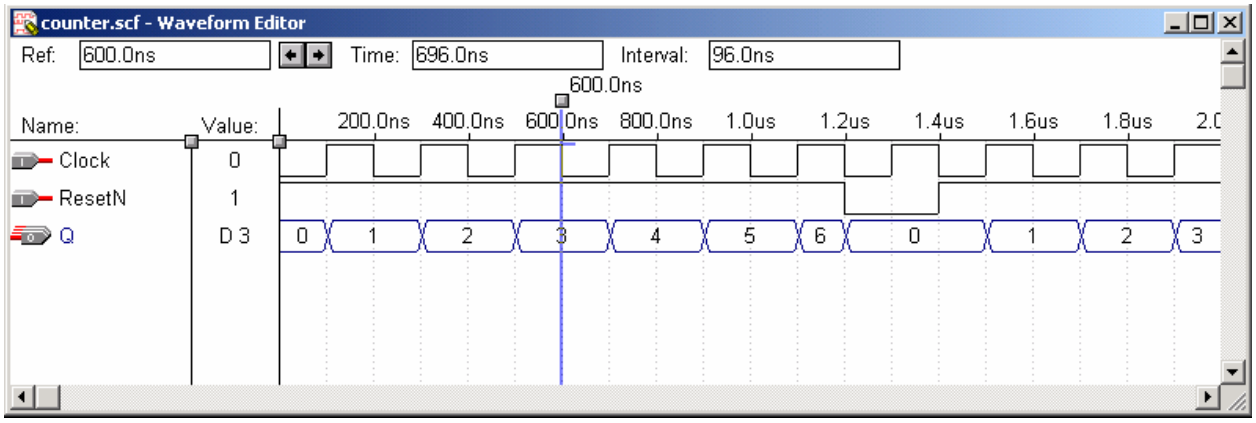

Figure B.10 Resulting waveform after the simulation.

6. You can change the **ResetN** signal values to something different and run the simulation again.

