VHDL ROM Design for PLD

Download:

- The Quartus project archive *Board_ROM.qar*
- The Excel spreadsheet *ROM_contents.xlsx*

ROM Inputs:

- CLK – Memory input clock (rising edge)
- OE_L – Output enable (active-low). When OE is false, the output of the memory is high impedance.
- CE_L – Chip enable (active-low). When CE is false, the output of the memory is high impedance.
- A[14..0] – Memory address bus (many of these bits must be fixed to Vcc or GND for this to fit in our device)

ROM Outputs:

- D[7..0] – Data bus

Understanding the VHDL created ROM:

1. Un-archive the project *Board_ROM.qar*.
2. Look at the top-level (BDF) file, *ROM_test.bdf*. You will need to do similar things in your top-level BDF file for your lab. Note how the address bus and PC are handled. You will need to make some of the address pins connected to Vcc and some to GND.
3. The *board_vhdl_ROM.vhd* file is probably beyond your understanding, but this file uses the file *board_ROM_vhdl_init.vhd*, a file that has the contents of the ROM contents. Open the file *board_ROM_vhdl_init.vhd* and see how the data is organized.
    a. If you needed to change the ROM contents, you could either change the data column in the Excel file or change it directly in the *board_ROM_vhdl_init.vhd* file.
4. Perform a **full** compile of the project (with *ROM_test.bdf* set for the top-level entity).
5. Open the simulation file *Waveform.vwf*. Select Simulation | Simulation Settings, and then click on the "Restore Defaults" button at the bottom. Then perform a timing simulation.
6. View the simulation results and see how they correspond to the contents in *board_ROM_vhdl_init.vhd* and in the Excel spreadsheet.

To include this design in your design:

1. Add both *board_vhdl_ROM.vhd* and *board_ROM_vhdl_init.vhd* to your Quartus folder.
2. Download the Excel spreadsheet *ROM_contents.xlsx*. Edit the file with the necessary data that you will need to match the data you want to put in the ROM for your lab solution.
3. Copy the columns of the Excel spreadsheet (without the top header row), overwriting the equivalent columns of *board_ROM_vhdl_init.vhd*. Alternately, you could change the data directly in the *board_ROM_vhdl_init.vhd* file and not use the Excel file at all.
4. Set *board_vhdl_ROM.vhd* set for the top-level entity and then do a full compilation.
5. Create a symbol for *board_vhdl_ROM.vhd* (as is done in the example BDF *ROM_test.bdf*).
6. Change the top-level entity to your relevant design. If *board_vhdl_ROM.vhd* and the **symbol file** for this are not already in the list in `Project | Add/Remove Files`, then add them.
7. Add the symbol and make the appropriate connections in your BDF.
8. Compile your design and simulate as normal.

WARNING:

By itself, the ROM is too big for our PLD if all (or most) of the address pins are used. You must limit the usable address space as described in the Lab 6 document for it to fit on your board. It is still possible to **simulate** the full size of the ROM.