Columbia University  
Department of Electrical Engineering  
EECS E4340. Synthesis and FPGA Place and Route Tools.

In this guide, we describe the tools flow for synthesizing your VHDL for implementation in the Xilinx XC4000E FPGAs. For additional reference please consult the following manuals, which can be found in the “Tools Manuals” binder in the Hardware Prototyping Lab (Mudd 1211):

- FPGA Compiler II User Guide from Synopsys.
- ISE 4 User’s Guide from Xilinx.

We will be using two FPGAs for our PDP-8: an XC4003E will be used to hold the dataflow and an XC4010E will be used to hold the controls. In our flow, we will be using the Sun Workstations for synthesis and mapping to the FPGA. We will then transfer the “bit stream” file for the slave serial programming of the FPGA over the the PCs in Mudd 1211. The “Impact” utility on the PCs will be used to download the FPGAs through the Multilinux cables. A lot has been upgraded this year: we have new tools, new download cables, and new project boards. Please be patient as we work out the bugs. Please contact me or the TA for help.

To test out the design flow, you can walk through a simple design found in the library light in /u2/ee4340/cdslibs or you can work with the datapath of your PDP/8 directly. You will need to make a few modifications to the design to handle the clocks. We will be using the 8MHz local oscillator on the XC4003E FPGA and use this to clock the logic in both FPGAs. To do this, you will need to create a component in your schematic called osc4 with a single output (no inputs) called f8m. This output should be connected to the clock of your design.

To synthesize your VHDL with Synopsys Design Compiler, bring up the VHDL toolbox from Cadence by choosing **Tools → VHDL Toolbox**. Then from the VHDL Toolbox, choose **Setup → Synthesis**. On the form that appears, choose “Synopsys” as the synthesis tool. Then choose **Commands → Synthesis**. Your design will be “checked-and-saved” again and the Cadence-To-Synopsys Interface (CSI) form will appear. From the form, choose the option to create a command file. In your CSI run directory, you will now have a file called “netlist.inc.” We will refer to this file to find the paths to all
the VHDL files and the order in which these files have to be analyzed by
the synthesis tool. Unfortunately, the syntax used is that for Synopsys Design
Compiler, and Synopsys (in their infinite wisdom) decided to use a different
syntax for FPGA Compiler II.

Copy the template command files into your run directory and edit them
to add in the VHDL files and Copy over the template into your run directory
and edit it as necessary. There are two template command files for you to
use: xc4003.cmd and xc4010.cmd. These files are in /u2/ee4340/public.
You must modify these files for your design. The VHDL path names can be
found in the “netlist.inc” file. To execute the command file, type

fc2_shell < xc4003.cmd > fc2.out &

This will run FPGA Compiler II, storing the results in the log.

You must carefully scrutinize the log for error and warning messages
which may indicate problems in your design. Please see me if you have error
messages that you do not understand. The other very important thing that
you need to scrutinize is are the tables of “inferred memory” devices that
come from synthesizing your VHDL process statements. You must make
sure that FPGA Compiler II created the latches you expected only for the
process statements for which you intended to create latches. You should not
have latches from the process representing the combinational logic of your
ASM. If you do, it means that you broke one of the rules (e. g., haven’t
activated the process by all inputs, haven’t covered all cases with a “default”
section in the top of the process).

The basic clock frequency of the design is specified in the create_chip
command. The template command files specify a 8-MHz clock. We have a
simple flop-based design, in which this clock cycle time (125 nsec) constrains
most of the paths. To constrain paths from the primary inputs and outputs,
you would in general need to specify arrival times on inputs or required arrival
times on outputs. For the time being, we will not consider how to add these,
but it may become necessary later to do this. These timing constraints (as
well as the clock constraint) are exported to a constraint file with the *.ncf
extension. The netlist for the design is found in the *.edf file. This file is in
EDIF (Electronic Design Interchange Format). Both the *.edf and *.ncf
files will be imported into the Xilinx ISE tool.

To place and route the designs, bring up the Xilinx ISE tool by typing
ise& at the UNIX command prompt to bring up the ISE GUI. Choose File
→ **New Project.** For Project Location, use the run directory you have been working in. Choose an appropriate project name. Select the Device Family of XC4000E. The Device is xc4003e-3pc84 for the XC4003 part and xc4010e-3pc84 for the XC4010 part. The Design Flow is EDIF. Now choose **Project → Add Source** to attach both the *.edf and *.ncf files to the project. Now in the “Processes for Current Source” window, select “Implement Design.”

With the right mouse button, select “Run.” This will run technology mapping and place-and-route your FPGA design. If this completes without errors, select “Generate Programming File.” With the right mouse button, select “Run,” which will generate the bit file for download to the FPGAs. This file will have the *.bit extension in your run directory. You may also wish to invoke the timing analyzer to check the timing of your design. Please work with me or the TA to understand how to do this.

Now, you will have to copy your bit stream file over to the PC’s in Mudd 1211. Please see me or the TA to go over the procedure here.