Informal Startup Guide to Ambit BuildGates (Now BuildGates Extreme!)

Bryan Audiffred

February 19, 2004

1 Introduction

This document should get a reader "up and running" with the Ambit BuildGates synthesis tools. It will cover launching the program and its basic usage. You should consult the existing manuals for full documentation.

2 Launching BuildGates

BuildGates is located in `/opt/local/cadence/spr/BuildGates`. It may be launched with the command: `bgx_shell -gui` for the GUI version. It may also be run as a command line tool. All paths should be setup when you log into the systems.

3 Reading Source HDL

Source Verilog must be read and parsed with the command `read_verilog`. This will syntax check the file. Note any errors. Alternatively, you may open the file with the BuildGates editor and press the parse button. Both will perform the same action. At this point, the design is ready for synthesis.

4 Generic Synthesis

For a generic synthesis run, use the command `do_build_generic`. This will automatically locate the top module in your source and synthesize from there. If a top module is not evident, i.e. there are no instantiations of lower

modules, you must specify the top or desired module with the `-module` option. For example, if you have a file of modules, and you wanted to synthesize module "mA1", you would enter `do_build_generic -module mA1`.

You may view the results by double clicking the module in the hierarchy window and selecting the schematic tab in the right window.

5 Optimization

Optimization is accomplished with the `do_optimize` command. There are many options, so you may want to press the optimize button in the toolbar and use the GUI for options. Optimization is dependent on a technology library. The only reasonably complete one available is the lca300kv library located in
`/opt/local/cadence/spr/BuildGates/version/lib`
`/technology/ambit/alf/lca300k.alf`.
You may want this read in at startup for convenience. You may accomplish this by adding the command `read_alf <library path>` to the file `~/.ambit/.acshrc`. This file is parsed on program launch without the `-no_init` option.

6 Performance Estimates

For performance estimates, you will need to provide some data on clocks at a minimum. You must first define an ideal clock with the command.
`set_clock <ideal clock name> -period <#>`
The ideal clock name can be anything. The next command connects it to a design.
`set_clock_root -clock <ideal clock> -pos <clock pin>`
The following two commands may also be accomplished with the GUI.

The basic performance report is accessed with the `report_timing` command. If given without options it lists the worst path and a summary of the other paths.

7 Functional Synthesis Verification

To verify the functionality of a synthesized design, you will need to write out a verilog netlist. This is easily done with the `write_verilog` command. The `-equation` option writes out a technology independent netlist. This method does not require simulation libraries for technology specific blocks.

On the downside, there is no timing information in the simulation. Full verification with timing is a more complicated matter, but it is possible.

8 Further Reading

There are three key manuals for the tool:

esug.pdf This is the general BuildGates manual

syncomref.pdf This is the command reference (very large document).

synhdlmod.pdf This is the synthesis style guide.

All of these are found in the `../cadence/spr/doc` directory.