University of Toronto, Faculty of Applied Science and Engineering
Department of Electrical and Computer Engineering

ECE 1387F - CAD for Digital Circuit Synthesis and Layout Handout #8

Exercise #1 - Simulated Annealing Cooling Schedules and Timing-Driven Routing

October 1999 J. Rose

Assignment Date: October 13
Due Date: October 20 (before lecture begins)
Late Penalty: **-1 mark per day late, with total marks available = 10**

The purpose of this exercise is to use an automatic placement tool based on the Simulated Annealing optimization strategy, and to gain some familiarity with the properties of that strategy, in particular the cooling schedule.

The placement tool that you will use in this exercise is called "vpr." It is a tool created by Vaughn Betz, as part of his Ph.D. thesis.

1 Access to Software

The executable can be found in ~jayar/1387e1/vpr on eecg, and ECF, and ECE ugsparcs, and is available at the web address http://www.eecg.toronto.edu/~jayar/courses/ece1387/e1/testfiles/ as an executable for both Sun Sparcstations and windows PCs. (I'm not 100% sure that the windows version will work properly, so you may have to use ECF).

2 Instruction Manual for VPR

See the handout attached. Section 5.2 on page 6 has most of the parameter information you'll need. Other sections describe the router and the routing parameters, which you'll need to know about. The complete manual describes other aspects of VPR and VPACK, a tool that packs LUTS into clusters.

3 Netlist File and Support File to Use

Use the file ~jayar/1387/e1/alu4.net as the input netlist file, and the lut44.arch as the architecture file. You may find reading the arch file interesting, as it gives the description of an FPGA architecture.

4 Exercise A - Placement and Simulated Annealing

The vpr program allows the user to set various Simulated Annealing parameters: the starting temperature, the ending temperature, the rate at the temperature decreases, the number of move generations per cell, and the initial random seed.

1. Run the vpr program once, with its default parameters, on the alu4.net netlist,

using the place_only option to prevent routing. Plot the score (cost function) versus the temperature. If you turn on the “toggle nets” option, you can see the rat’s nest of wires become less tangled. (use the auto 1 option). Indicate on the plot the temperature at which the placement appears to “freeze.”

Note to avoid the graphics display, use the no_disp option.

2. Run the vpr on the circuit ten times with the default parameters and calculate the mean and standard deviation of the resulting final scores.

3. Simulated Annealing run at temperature zero is called a “quench”. Run a quench 10 times with different random seeds (using the seed parameter) and record the final scores. Calculate the mean and standard deviations. How does it compare to the results from part 1? What feature of Simulated Annealing does this illustrate?

4. Using no more than 30 runs of vpr, generate a plot of starting temperature (using the init_t num parameter) versus final score achieved. Do enough runs per temperature to that you can get a reasonable value for the mean and standard deviation at each temperature. Comment on the results.

5. Using no more than 30 runs of vpr, generate a plot of rate of cooling (using the alpha_t parameter) versus final score achieved. Use the same starting temperature as the default run. Do enough runs per temperature to that you can get a reasonable value for the mean and standard deviation at each temperature. Comment on the results.

6. As in parts 4 and 5 determine the effect of the number of moves per cell, using the inner_num parameter.

5 Exercise B- Timing-Driven Routing

1. VPR also does complete timing-driven routing using an improved version of the pathfinder approach described in class. Run the placement tool (with the -fast option) *and* the timing-driven router. Report what the critical path delay is.

2. Explore the different parameters of the router and try to improve the critical path delay using those parameters, for the alu4.net circuit.