# New Java Tool Increases Pipelining *Speed*

***Genesis One Technologies has developed a Java-based tool that automatically creates pipelining in Xilinx designs.***

*by Ben Israel, President, Genesis One Technologies, Inc, ben@genesis-one.com*

AutoPipe is a new tool that reads Xilinx-generated EDIF simulation files, rebuilds the structure, and writes out a pipelined Verilog file with the same logic. This verilog file is then re-synthesized with FPGA Express. AutoPipe is written in Java, and is controlled from a command line. The number of stages, the EDIF input file, and the Verilog output file are its only parameters; you apply it to your time-critical modules and no source code changes are necessary. The result is greater design speed and less design time.

Pipelining is basically an assembly line; results are produced every clock. In a pipelined design the first stages can begin processing new inputs while the last stages are finishing outputs. Without it, input stages must wait unproductively, for the output stages to finish

To use pipelining without an automated process, you must first select the number of stages, then place or schedule all the logic in the appropriate stage. Each time you change the number of stages, all the logic must be rescheduled. This greatly complicates the design process, is error prone, and is the most likely reason why most designs today are not pipelined.

Pipelined designs get maximum speed in the least area. Figure 1 shows the design tradeoffs for a network switch. This example has 12 logic levels, was synthesized and implemented in a Virtex V300FG456-6 FPGA with automatic placement and routing.

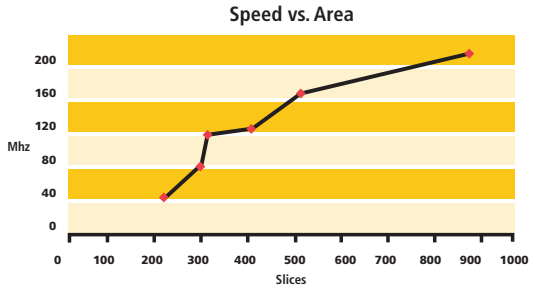


*Figure 1 - Speed vs. Area Tradeoffs*

Running AutoPipe with one stage is the equivalent of no pipelining. This creates a version of 218 slices that runs at 45 Mhz. A three stage version has 310 slices and runs at 102Mhz; over twice the frequency with a less than 50% area increase. A maximally pipelined version with12 stages runs at 177Mhz using 889 slices; four times the frequency with four times the area.

## Conclusion

With AutoPipe, you don't have to think about pipelining until your design is complete. Then, you can test different versions and choose the most efficient one. It can take a single design and easily create pipelined versions with up to four times the clock frequency. Because AutoPipe is applied after your design is done, you do not have to change the way you work to see the benefits. Finally, AutoPipe is not subject to human error and can even be applied to old designs.

For more information on AutoPipe contact Ben Israel President at: ben@genesis-one.com; 561-852-2146