

# Implementation of Target Recognition Applications Using Pipelined Reconfigurable Hardware

Ben Levine  
Carnegie Mellon University

## 1. Motivation:

Intelligence, Surveillance and Reconnaissance (ISR) systems present unique challenges to digital designers. Computational hardware for ISR systems needs to be able to process large amounts of data, often in real time, while meeting stringent physical constraints. These data processing requirements are increasing rapidly as new sensors come online and increasing amounts of automation are desired. Commodity processors can be used in some cases, but their power requirements are high and they are often very inefficient for the highly parallel and repetitive algorithms common in ISR applications. ASICs can be used to meet performance and physical requirements in many cases, but they also have many drawbacks, especially high design cost, long design turnaround time, and inflexibility. Programmable logic devices (PLDs), such as FPGAs, are increasingly being used in place of ASICs because of their short design cycle, performance, and flexibility. However, implementing applications on an FPGA can be difficult and requires hardware design skills. FPGAs are also expensive and take a long time to be reconfigured for different applications. Further, FPGA designs are not scalable. An application designed for one particular size FPGA cannot be run on a FPGA with more resources and get better performance without changing the design.

## 2. PipeRench

The PipeRench architecture [Ref 1] was developed at Carnegie Mellon University (CMU) to address some of the problems of conventional PLDs. It is a reconfigurable pipelined architecture composed of eight-bit functional units, register files, and a rich interconnect network between pipeline stages. The PipeRench architecture was designed to implement high performance custom datapaths for a wide range of applications, include signal processing, image and video processing, cryptography, and image analysis. PipeRench uses a technique called pipeline virtualization to automatically virtualize hardware, allowing for the implementation of application pipelines that have more pipeline stages than are present on the physical chip. The same executable will run faster on a PipeRench chip with more physical pipeline stages than on a chip with fewer physical pipeline stages. This is possible because the PipeRench chip can reconfigure an entire pipeline stage in a single clock cycle (8 ns in the current implementation). This also means that PipeRench can switch applications in a single clock cycle. PipeRench is programmed using a high level language called DIL (Dataflow Intermediate Language) and has a robust compiler and assembler.

## 3. Project Description and Results

Northrop Grumman Corporation's (NGC) Image and Information Research group in Pittsburgh, PA, develops image and signal analysis applications for numerous ISR systems. Often they are limited by the amount of processing power of the target platform and are unable to fully utilize the algorithms they develop. In 2000, researchers from NGC and CMU began working together to determine if the PipeRench architecture was suitable for implementing applications of interest to NGC. An existing target recognition application for SAR imagery was chosen as a test subject. This application requires lots of computing power, as it was designed for accurate recognition of targets and had to be applied to large images at high data rates. The only specification of the application that was available was the original C source code. The first task was to analyze the source code and find the computationally intensive portions of the code, or kernels, typically an inner loop nest. Careful profiling of the code was performed using a large set of test data and it was determined that 13 kernels comprised 65% of the total application runtime. Of the remaining 35%, the vast majority of cycles were consumed by Solaris system calls; this overhead would be much less on more typical ISR hardware.

The C code for the kernels was analyzed and a high-level description of each kernel was developed and verified with the application designer. NGC engineers were able to learn DIL and begin coding kernels quickly from the high-level descriptions. All but one of the kernels could be implemented efficiently on PipeRench and was ported to DIL. Since the PipeRench chip had not yet been fabricated when this initial work was being done, a cycle accurate simulator was used to run each kernel. The final results showed that all of the ported kernels ran faster on PipeRench, with speedups ranging from 8x to over 300x versus the original C code running on a Solaris workstation.

In 2002, prototype PipeRench chips were available for use, and the project was resumed. The PipeRench prototypes were fabricated in a commercial 0.18 micron process, had 3.65 million transistors, and occupied a die area of 49 sq. mm. [Ref 2] Time constraints allowed for the implementation of only three kernels on the PipeRench hardware, due mainly to unforeseen difficulties with the hardware and software used to interface the PipeRench chips to the host computer. The PipeRench chips performed very well on the kernels implemented, with performance near what was predicted and without errors. While no direct power comparisons of the NGC application kernels were made to other implementations, the kernels implemented required less than one watt of power for the PipeRench chip.

#### **4. Conclusions**

Even though it was not designed specifically for target recognition applications, it was relatively easy to port an existing application to PipeRench, and significant performance speedups were observed, as well as low power consumption. The performance scaling and fast reconfiguration of PipeRench hold promise for many ISR applications.

#### **5. References**

- [1] Seth Copen Goldstein, Herman Schmit, Mihai Budiu, Srihari Cadambi, Matthew Moe, R. Reed Taylor, "PipeRench: A Reconfigurable Architecture and Compiler," in *IEEE Computer*, pp. 70-77, April, 2000.
- [2] H. Schmit, D. Whelihan, A. Tsai, M. Moe, B. Levine, R. R. Taylor, "PipeRench: A Virtualized Programmable Datapath in 0.18 Micron Technology," *Proceedings of the IEEE Custom Integrated Circuits Conference (CICC)*, 2002.