ABSTRACT

This dissertation describes a methodology for generation of a custom memory interface and associated direct memory access (DMA) controller for FPGA-based kernels that have a regular access pattern. The interface provides explicit support for the following features: (1) memory latency hiding, (2) static access scheduling, and (3) data reuse. The target platform is a multi-FPGA platform, the Convey HC-1, which has an advanced memory system that presents the user logic with three critical design challenges: the memory system itself does not perform caching or prefetching, memory operations are arbitrarily reordered, and the memory performance depends on the access order provided by the user logic. The objective of the interface is to reconcile three problems described above and maximize overall interface performance. This dissertation proposes three memory access orders, explores buffering and blocking techniques, and exploits data reuse for the synthesis of custom memory interfaces for specific types of kernels. We evaluate our techniques with two types of benchmark kernels: matrix-vector multiplication and 6- and 27-point stencil operations. Experimental results show the proposed memory interface designs that combine memory latency hiding, access scheduling and data reuse achieve an overall performance speedup of 1.8 for matrix-vector multiplication, 2.2 for 6-point stencil, and 9.5 for 27-point stencil as compared to using a naïve memory interface.