Abstract

Architectural synthesis into a Field Programmable Gate Array (FPGA) from a behavioral description often fails to provide optimal performance, due to its inability to harness the high speed promise of the internal fabric properly. Coupled with this, FPGAs support only a handful logic cell configurations, for which not many high speed implementation variants exist for a particular design. By adopting state-of-the-art FPGA slice architecture as a representative example, this dissertation addresses the involved design challenges of speed-area efficient implementations and inclusion of testability with minimal overhead.

Our approach is directed towards realizing modular, cascadable, bit-sliced structures using Huffman style of digital design modelling, which enhanced the possibilities of logic optimization and superior FPGA mapping, often adorned with VLSI testability. The cascade connections have been carefully implemented using the hardwired fabric of the FPGA to reduce dependability on the slower programmable routing resources. We studied the percentage of underutilization in the instantiated FPGA slice primitives to append necessary functionalities and testable logic, all of which can be coalesced into the original design with minimal overhead. The addressed solutions culminate into superiorly crafted architectures for multiple datapath design units, equipped for maximum speed performance for a given FPGA fabric architecture. An informed choice of a proper algorithm and appropriate encoding schemes in digital arithmetic were executed keeping FPGA specific design optimization into consideration.

Unification of the supposedly disjoint concepts of efficient datapath design and testable logic insertion was carried out for a class of FSMs called cellular automata (CA), where scan insertion architecture which facilitates fault localization can also initialize the FSM to a new state as part of its original functionality requirements. CA is governed by nearest neighbourhood adjacency, best understood for one-dimensional structures involving adjacent neighboring cells to govern logic functionality and support scan, which makes it attractive for VLSI implementations. For a two-dimensional CA, neighborhood adjacency is schematically expressed through trees, graphs and square arrays. We have proposed algorithms to realize such geometries in hardware by efficient, predominantly columnar logic packing into FPGA slices, that retains maximal adjacency in physical implementations, and incurs less interconnection delays.

The outcome of such exercises resulted in C-testable designs, alternating logic realization for error detection, or scan path insertion, all through optimized logic configuration without any discernible speed or area penalty. Additionally, our proposed design automation facilitated architecture realization and predictability of implementation results.