B1: Analysis of FPGA-Based Reconfiguration Methods for Mobile and Embedded Applications

Darshika Gimhani Perera, University of Colorado, USA

As mobile and handheld devices are gaining popularity, many applications have found their way into these devices. Apart from optimized hardware-software architectures, new techniques and design methodologies are needed to support applications running on mobile systems. FPGA-based dynamic reconfigurable systems are currently the most feasible option to deliver embedded applications that have stringent requirements. There are different methods of reconfiguring the hardware on chip dynamically. Selecting a specific reconfiguration method and designing the corresponding hardware architectures for an application are important and challenging tasks in reconfigurable computing systems. In this work, we investigate different FPGA-based reconfiguration methods; study their features, advantages and disadvantages; and analyze the reconfiguration time and space overhead for each method.

B2: An EG-LDPC Based 2-Dimensional Error Correcting Code for Mitigating MBUs of SRAM Memories

Enver Çavuş, Yıldırım Beyazıt University, Turkey

In this paper, a 2-D error correction code architecture based on EG-LDPC and single parity check (SPC) code is proposed as a solution to MBU problem of SRAM memories. EG-LDPC codes have better multiple error correction and detection capabilities than conventional codes and they have low complexity decoders. Therefore, they are suitable for fault tolerant memory applications. The proposed architecture uses (15,7,5) EG-LDPC as row encoding and SPC code for column encoding. In order to minimize decoding complexity of 2D structure, a standard array decoder is utilized. The investigated architecture is compared with previously proposed Matrix code method. The proposed architecture is able provide over 95% error correction coverage up to 4 errors and a significant 100% error detection up to 12 bit errors. In terms of MTTFs, the proposed approach achieves 63% improvement over Matrix codes at fault rates of 10-4 and 10-5. Matrix codes and the proposed architecture are implemented using Xilinx XC6SLX16 FPGA and comparison results in term of implementation complexity are provided.
Index Terms— Multiple Bit Upsets (MBUs), EG-LDPC, two dimensional error correcting codes (ECCs), fault tolerant memories, soft errors.

B3: High Level Synthesis Based Hardware Accelerator Design for Processing SQL Queries

Gorker Alp Malazgirt, Bogazici University, Turkey

About three exabytes of data is created and stored in databases each day, and this number is doubling approximately every forty months. Querying this enormous amount of data has been a challenge and new methods have been actively researched. In this paper, we present hardware accelerators which are designed to speed up database analytics for in- memory databases. Unlike traditional hardware

accelerator designs, our hardware accelerators are composed using High Level Synthesis (HLS), which enables high level descriptions of functionality such as data filtering, sorting, equijoins to be targeted directly into RTL. We have simulated TPC-H benchmark queries using Xilinx Vivado HLS managed in our custom simulation software framework. Our results have demonstrated the capabilities of HLS in database accelera- tion domain; such that the 200MHz FPGA accelerator can provide two orders of magnitude performance improvement compared to PostgreSQL based full software implementation running on a modern multicore system.

B4: Comparison of Predictive-Corrective Video Coding Filters for Real-Time FPGA-based Lossless Compression in Multi-Camera Systems

Yimu Wang, Hasselt University, Belgium

Combining multiple cameras in a bigger multi- camera system give the opportunity to realize novel concepts (e.g. omnidirectional video, view interpolation) in real-time. The better the quality, the more data that is needed to be captured. As more data has a direct impact on storage space and communication bandwidth, it is preferable to reduce the load by compressing the size. This cannot come at the expense of latency, because the main requirement is real-time data processing for multi-camera video applications. Also, all the image details need to be preserved for improving the computational usage in a later stage. Therefore, this research is focused on predictive-corrective coding filters with entropy encoding (i.e. Huffman coding) and apply these on the raw image sensor data to compress the huge amount of data in a lossless manner. This technique does not need framebuffers, nor does it introduce any additional latency. At maximum, there will be some line-based latency, in order to combine multiple compressed pixels in one communication package. It has a lower compression factor as lossy image compression algorithms, but it does not remove human invisible image features that are crucial in disparity calculations, matching, video stitching and 3D model synthesis. This paper compares various existing predictive-corrective coding filters after they have been optimized to work on raw sensor data with a color filter array (i.e. Bayer pattern). The intention is to develop an efficient implementation for System-on-Chip (SoC) architectures to improve the computational multi-camera systems.