Multi-User FPGA Co-simulation Over TCP/IP

By D. Denning, J. Irvine, D. Stark and M. Devlin

Presentation by Martti Huttunen

martti.huttunen@ee.oulu.fi

Introduction

- FPGA Co-simulation
  - An efficient method of verifying designs and increasing simulation speed
- Co-simulation environment
  - Xilinx System Generator tools and Matlab used as simulation environment
  - DIME-II hardware platform with FUSE Matlab toolbox
- Case study
  - Co-simulation from multiple workstations
  - AES-128 and Camellia encryption algorithms

DIME-II hardware platform 1/2

- Components
  - Xilinx XtremeDSP kit
  - Nallatech BenOne motherboard
  - BenADDA module: Virtex-II with 2 A/D and D/A converters
- DIME-II standard
  - Defines rules for high-bandwidth communication between motherboard and modules as well as clocking issues and hardware interfaces
  - Standardised API (FUSE) available as a C library and a Matlab extension toolbox

DIME-II hardware platform 2/2

- TCP/IP interface
  - A separate add-on board for DIME-II motherboards
  - Rabbit Semiconductor RabbitCore R3000 programmable controller
  - Allows communication directly to DIME-II platform, no PC required

FUSE toolbox for Matlab

- Control, configuration and location of DIME-II boards
  - Location of cards per IP address
  - Configuration of boards
  - Down/uploading of bit files to FPGA devices
- Matlab mex-extensions
  - Platform-dependent compiled extensions (as opposed to regular Matlab script extensions)
  - Implements interfaces between the Matlab-code and the FUSE library

Hardware Co-Simulation 1/2

- Directly supported in Xilinx System Generator
  - Simulink simulation environment
  - Testbench data and results are transferred between the Simulink and the actual FPGA
  - Increased simulation speed (up to 100 times)
    - Dependent on the algorithm and required amount of data transfers
- Saves local resources for post-processing etc.

Hardware Co-Simulation 2/2

- Xilinx System Generator limits data width to 32 bits
  - In cases of longer data, a split/join chain must be implemented to overcome this limitation
  - 32-bit values are stored in parallel buffers and fed to the algorithm

Co-Simulation over TCP/IP 1/2

- Setup 1: Platform board in a PC
  - The platform FPGA card is connected to a PC via PCI or similar extension bus
  - TCP/IP access is implemented on the PC platform
- Setup 2: Direct TCP/IP access to FPGA board
  - The service program is implemented on the network card of the DIME-II platform
- The board may be clocked synchronously or the board clocks itself and the data input becomes asynchronous

Co-Simulation over TCP/IP 2/2

- Multi-User Co-Simulation
  - A single platform FPGA is shared by multiple workstations in the network
  - Only one user may use the board at a time
- Operating sequence
  - Location of FPGA board
  - Board allocation
  - Board configuration and device programming
  - Simulation
  - Board deallocation
- GUI extension to Xilinx System Generator for user

Case study: encryption cores

- AES-128 and Camellia encryption algorithms
  - 128-bit data encryption methods
- Test case 1: single user
  - Direct access to platform FPGA, no server PC involved
  - Geographic distance between workstations ca. 600 km
  - Average simulation speedup 15 times faster using remote FPGA hardware
- Test case 2: multi-user
  - Same cores tested with one user in local network and one in the case 1 network
  - No queuing mechanism: first-come, first-served