SYNOPSYS®
Accelerating Innovation

Synopsys and Friedrich-Alexander-Universität (FAU)

Computer Architecture Research Group Applies Synopsys FPGA-Based Prototypes to Prove High-Performance Video Processor Tile Design

The Synopsys FPGA-Based Prototyping Solution provides our research staff high performance with design visibility and system control. These attributes allow us to rapidly bring up design variations and operate them at multi-megahertz speeds.

Dr.-Ing. Frank Hannig
Architecture and Compiler Design group leader at Department of Computer Science at Friedrich-Alexander-Universität (FAU)

Academic Research

University of Erlangen-Nuremberg (German: Friedrich-Alexander-Universität Erlangen-Nürnberg) better known as FAU is a public research university in the cities Erlangen and Nuremberg in Bavaria, Germany. Research into parallel computing and video processing applications by the Computer Science department has led to the adoption of FPGA-based prototyping as a way to confirm algorithm function and characterize performance using hardware/software co-design techniques.

Challenges

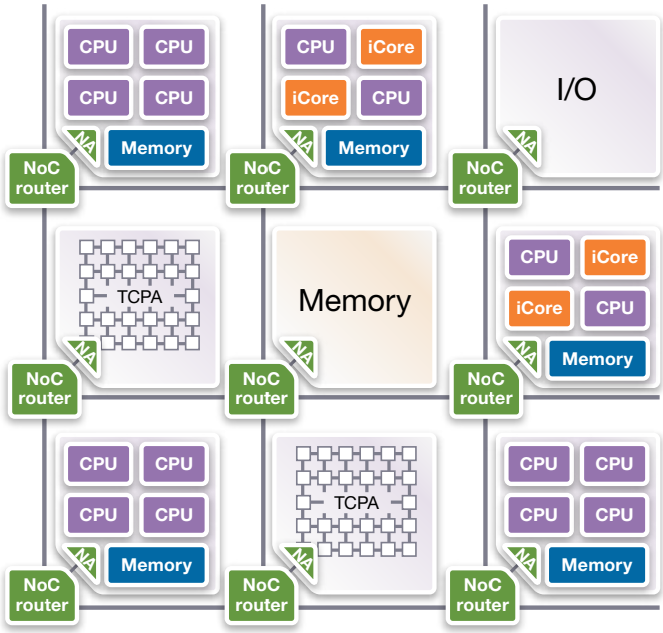
- Process high volumes of image and video data at resolutions of at least 1024x768 at near real time speed
- Display processing results in order to make qualitative review possible
- Allow rapid changes to be applied to base hardware or firmware algorithms
- Connectivity options for Digital Video Interface (DVI) Rx and Tx
- Visibility and control via external workstation

FPGA-Based Prototyping Solution

- Synopsys FPGA-Based Prototyping System providing multi-million ASIC gate capacity
- Certify Multi-FPGA prototyping environment and Identify RTL debugger software tool flows
- Universal Multi-Resource Bus (UMRBus) Interface Kit for host workstation connection
- Hybrid prototyping AMBA transactors for software application command and monitoring of prototype

Benefits

- Design planning and FPGA software flow with re-programmable prototyping platform for easy implementation of design revisions and architecture variation
- Prototype delivers processor frequencies of 55 MHz (pixel processing rate: ~47M Pixel/s) for near real-time performance versus traditional simulation kHz performance
- UMRBus for workstation connectivity enables software-based control and monitoring using familiar bus protocols



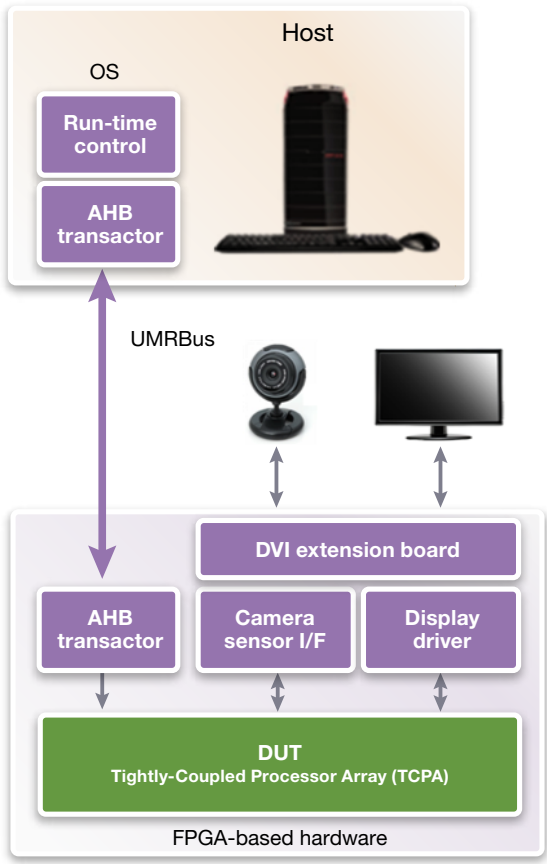
Overview

FAU's research team has developed a novel paradigm, called invasive computing for the exploitation of run-time parallelism of multi-processor SoC architectures through resource-aware programming and dynamic reconfiguration of the underlying architectures. The invasive algorithm organizes the computation task based on the state and availability of resources among a configurable array of processors. Further details are available at invasic.informatik.uni-erlangen.de.

The architecture research includes the design of a heterogeneous tiled-architecture. Different tiles such as RISC-based processors, accelerators and peripherals are connected using a proprietary network-on-a-chip (NoC). Within a tile, a shared memory and an AMBA bus are used for communication.

The hardware/software prototype utilized a combination of a Synopsys FPGA-based prototyping system connected to a host workstation to model the run-time system interactions with TCPA (Tightly-Coupled Processor Array) design under test (DUT). A run-time control application communicates with the prototype using an abstraction of the AMBA AHB protocol interface enabled by a combination of the physical link (UMRBus Interface Kit) and an AHB transactor.

The data exchange link allows the host workstation to issue resource request/release commands, configure the video processor algorithm (Sobel, Laplace, etc.) and manage data streams for a camera sensor and video display.



Before adopting the Synopsys prototyping system FAU relied on FPGA vendor produced evaluation boards to test research projects. And while the evaluation board could provide good performance for video processing they could not scale to support larger implementations of the design. Evaluation boards typically do not provide connectivity options for testing and validation, which required FAU to rely on synthesizable test jigs for the TCPA processing tiles and implemented them directly onto the host FPGA. This consumed precious resources that would otherwise be used for the DUT. The Synopsys prototyping system with UMRBus and AMBA transactors allowed FAU to use a host workstation to provide external control and monitoring using software applications, which are far more flexible and less disruptive to the prototype.

Leading FPGA-Based Prototyping Solution

Synopsys' integrated system and tool flow including synthesis and debug software accelerated FAU's time to an operational prototype. The complete prototyping solution was also easily transported for demonstrations at industry conferences, meeting FAU's need for a portable solution. They found that the UMRBus Interface kit with the high-level API into the DUTs embedded SoC bus allowed easy host workstation design visibility and control of the prototype not possible with traditional methods.

Example: Input/Output Video Frame Comparison - Partial TCPA array utilization with Laplace 3x3 edge-detect algorithm applied.

FAU confirmed that FPGA-based prototypes are feasible for MPSoC validation and demonstration:

- High performance FPGA synthesis implementation supports video processing up to 47 MPixel/s
- Well documented prototype hardware I/O interface eased expansion to Digital Visual Interface (DVI) Rx/Tx peripherals
- Scalable hardware capacity and prototype planning automation allowed multi-million ASIC gate designs to be partitioned across multiple FPGAs
- Hybrid prototype solution enables user applications to communicate FPGA-based hardware system via C++ API and familiar AMBA bus protocol commands (Read/Write/Interrupt)
- Fast prototype turnaround time allows designers to illustrate various TCPA utilization and algorithm processing quality

Synopsys, Inc. • 700 East Middlefield Road • Mountain View, CA 94043 • www.synopsys.com

02/13.RP..CS2714.