Turun yliopisto
University of Turku

A NoC-AXI Interface for an MPSoC Platform on FPGA

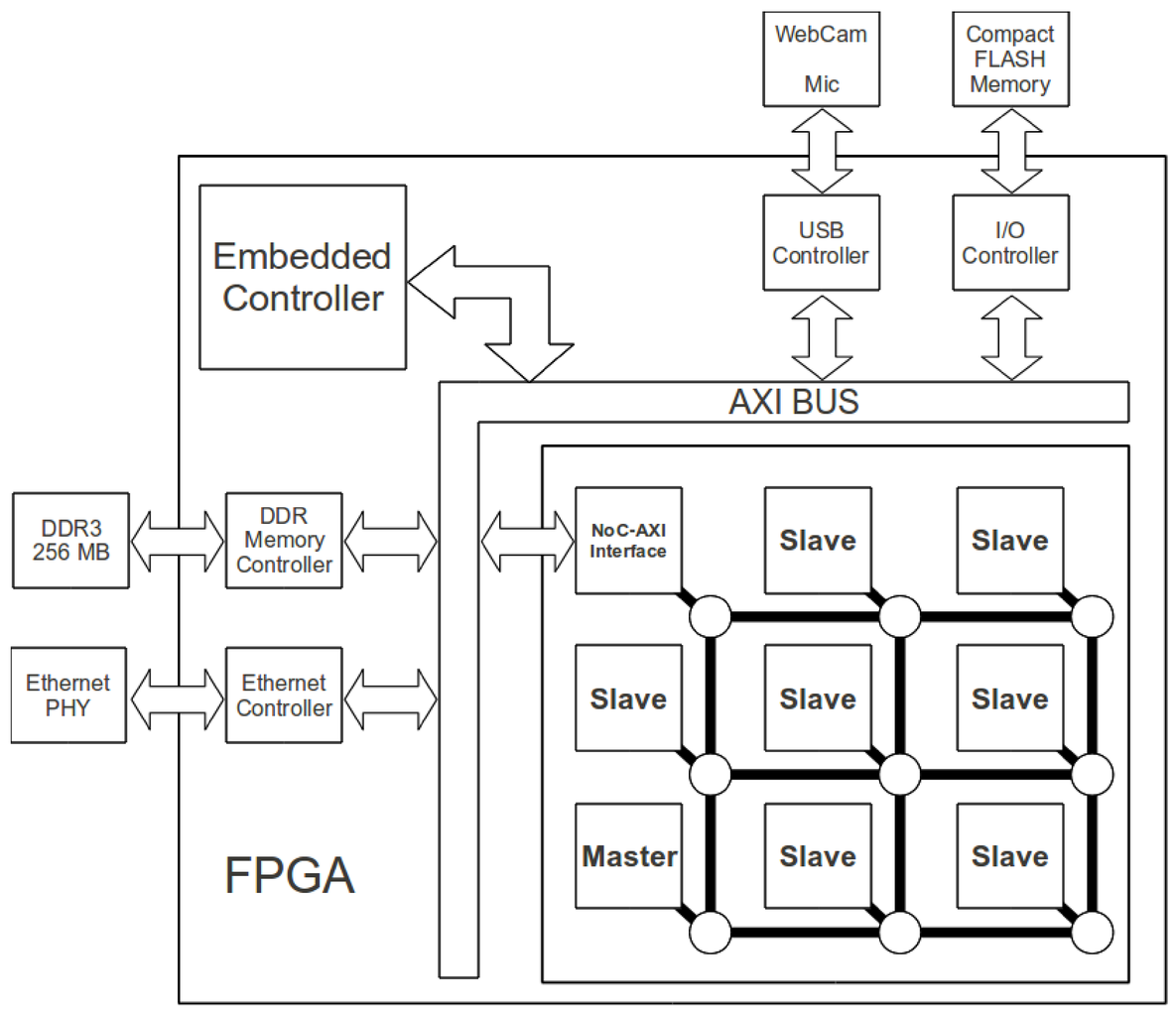
Marco Ramirez, Masoud Daneshtalab, Juha Plosila, Pasi Liljeberg

Introduction

IP-based Digital Video Broadcasting, Video on Demand and on-line gaming are some examples of popular services on the Internet. Video streaming is a very resource-demanding process since it involves the execution of complex algorithms for video compression as well as networking tasks.

Providing the necessary throughput while meeting the timing constraints is necessary to maintain an acceptable quality of service. These stringent requirements make single-processor architectures inadequate for the video streaming so that multi-processor systems-on-chip (MPSoCs) have been proposed as an alternative approach.

The HeMPS framework is a popular MPSoC research tool. It generates a parameterizable architecture based on the Plasma processor and the HERMES NoC. A typical instance of the architecture has three main components: processing elements, routers and a task repository. In this work we present a HeMPS-based MPSoC platform on FPGA, named AXI-based MPSoC (AXIM).



AXIM

The AXIM platform has three main components: a cluster of processing elements, the AXI subsystem and an embedded controller.

Cluster of PEs (CPE): Implements the same type of processing elements and NoC used by HeMPS platform.

AXI Subsystem: Incorporates controllers for external DDR memory, Ethernet, FLASH memory and USB host. The AXI subsystem constitutes the set of sinks and sources for the streams to be processed.

Embedded Controller: Runs an operating system with the required drivers in order to configure the system's peripherals during the initialization phase. EC also runs a web server in order to handle the job requests received via HTTP.

NoC-AXI Interface

The NoC-AXI interface creates the communication channel between the CPE and the AXI bus. It is also responsible for allocating part of the system's task repository. The NoC-AXI functionality is provided by several internal units:

Network Interface: It handles the credit-based data flow control between the router and the rest of the internal units.

AXI Units: Two units are in charge of generating transactions on the AXI bus, one unit for each type of operation (read/write). The AXI units are independent thus allowing parallel execution of transactions.

Control Unit: It reads the incoming packets from the network interface, processes the encapsulated commands and instructs other units accordingly through the control bus.

Local Memory Unit: Stores information related to the set of applications available in the repository. The information is kept as pairs of integers describing each task's base address and block size.