A Hardware Platform for VLIW Based Emulation of Digital Designs

G. Haug[1,3]
[1]Forschungszentrum Informatik
Haid-und-Neu-Str. 10-14
Karlsruhe/Germany
haug@fzi.de

U. Kebschull[1,2]
[2]Universität Leipzig
Augustusplatz 10/11
Leipzig/Germany
kebschull@fzi.de

W. Rosenstiel[1,3]
[3]Universität Tübingen
Sand 13
Tübingen/Germany
rosenstiel@fzi.de

Abstract

In [2] the concept of a very long instruction word (VLIW) *processor based system to emulate synthesized RT-level descriptions has been presented. As described in [2] the* RAVE System *(**R**T-**A**rchitecture-**V**LIW-**E**mulator) overcomes many of the problems common to FPGA based emulation and prototyping systems. Particularly, these are area problems in conjunction with large data paths, long turnaround times and low emulation clock frequencies. This abstract briefly describes the hardware of the RAVE System.*

1 Introduction

Most emulation systems work on a very low level of abstraction: they emulate a gate net list. Assumed the design flow starts with an algorithmic specification, the first task to be accomplished is high level synthesis. This step introduces a clock and produces an RT-level specification. As long as no constraints given to the high level synthesis tool are violated, the tool is free to allocate functional units and schedule the operations as it wants to. Next step of the design flow is logic synthesis, producing the gate-level net list. To implement a data path with 32 bit word length in FPGA technology is very challenging if possible at all, as one might run into area problems even with the largest chips available. If the FPGAs are large enough to accommodate the design, still two problems remain: Long turnaround times and slow emulation clocks. Therefore, a VLIW based emulation platform is proposed here. This means an RT-level description (data path and controller) has to be translated into a VLIW program.

2 Hardware of the RAVE System

The heart of the RAVE-HW module is a Texas Instruments TMS320C6201 (C6x) DSP running at 200 MHz [1]. The instruction set of the C6x is more like that of a general purpose RISC processor than a classical DSP. Up to eight instructions can be executed in parallel (fetch packet). Regardless of the instructions contained, a fetch packet takes one clock cycle, i. e. 5 ns at 200 MHz. This property makes the behaviour of the code highly predictable. In case a multi VLIW processor/FPGA emulation structure is required, the processors amongst each others as well as the processors and the FPGAs have to be synchronized. The problem is solved by replacing the clock source of the FPGAs. At the end of each assembly sequence representing a transition of the controller an access to the synchronization device (a memory mapped I/O device in the address space of the processor) is inserted. This access triggers a pulse of programmable length on an output signal of the device. This signal is distributed to the FPGAs of the emulation structure as one of their clocks. Besides the generation of the clock for the FPGAs the synchronization device is responsible for the synchronization of the multiple processors (up to six) in emulation structures with more than one RAVE-HW module (each module contains a synchronization device). Therefore, the mapping software inserts read accesses (load instructions) at the end of each assembly sequence. The synchronization device is able to stall off the read access by deasserting the ready signal of the C6x. The emulation I/O unit is a programmable parallel I/O device. It has to be highly flexible and must support a large number of I/O lines. None of the readily available chips did fit these requirements. Therefore, the emulation I/O unit was implemented using four FPGAs, one for each connector.

References

[1] *TMS320C62x/C67x CPU and Instruction Set*. Texas Instruments Incorporated, Houston, Texas, 1998.

[2] T. Buchholz, G. Haug, U. Kebschull, G. Koch, and W. Rosenstiel. *Behavioural Emulation of Synthesized RT–level Descriptions Using VLIW Architectures*. Proc of the 9th RSP, Leuven, 1998.