Design and Comparison between PVCI, BVCI and OCP Hardware Reuse Interfaces Mapped to FPGA

Marcelo Porto [1], Thaísa da Silva [1], Roger Porto [1], José Luís Güntzel [1], Ivan Silva [2], Sergio Bampi [3], Luciano Agostini [1,3]

[1] Grupo de Arquiteturas e Circuitos Integrados (GACI)
Departamento de Informática – Universidade Federal de Pelotas (UFPel)
[2] GME – Universidade Federal do Rio Grande do Sul (UFRGS)
[3] DIMAp – Universidade Federal do Rio Grande do Norte (UFRN)

{porto, thleal, rogerecp, guntzel, agostini}@ufpel.edu.br, ivan@dimap.ufrn.br, bampi@inf.ufrgs.br

ABSTRACT

The study of hardware reuse standards becomes extremely important with the microelectronics industry growth. Amongst these standards we can point out the Open Core Protocol (OCP) and Virtual Component Interface (VCI). This paper presents the design of PVCI, BVCI and OCP standard interfaces and a short comparison between then. The interfaces were described in VHDL and synthesized for Altera FPGAs.

1. INTRODUCTION

The study on hardware reuse standards is a very important issue in current microelectronics designs. This is mainly due to the increase in complexity of systems on chip (SoCs) and the time-to-market requirements [1]. In this work we will focus in two of the most used hardware reuse interfaces: OCP (Open Core Protocol) [2] and VCI (Virtual Component Interface) [3].

OCP is one of the most used hardware reuse standard. OCP is an open and free standard, differently to the VCI standard. The VCI standard, by its turn, was developed by VSIA (Virtual Sockets Interface Alliance) and it has three interfaces in its family: PVCI (Peripheral VCI), BVCI (Basic VCI) and AVCI (Advanced VCI).

This paper presents the design of PVCI, BVCI and OCP interfaces, establishing some comparisons between their synthesis results targeting FPGA implementation.

2. INTERFACES DESIGN

All the interfaces designed in this work have two main units one that makes requests and another one that accepts or not these requests. These units are called Master and Slave for OCP interface and Initiator and Target for VCI interfaces. The interfaces were described in VHDL using QuartusII environment and with synthesis directed to FLEX10KE Altera FPGA devices. The functionality and the communication protocol of the interfaces were validated through simulations.

2.1 PVCI Interface Design

The PVCI provides a simple interface and protocol. This interface is used for applications that do not need all the functionalities of the BVCI. This standard defines two main signals: **VAL** and **ACK**, which establish a *handshake* communication protocol. Moreover this standard has the buses those carry the contents of requests and responses.

The Initiator sends a signal VAL for the Target informing that there are valid values in its interface, which must be used for the Target. In turn, the Target answers for Initiator through a signal ACK, signaling that a transfer between the Initiator and the Target was finished successfully [3].

2.2 BVCI Interface Design

The BVCI defines an appropriate interface for the majority of the real applications. This protocol has a powerful and not very complex set of rules. The communication in BVCI interface happens between the Initiator (that is responsible for the content requests) and the target (that is responsible for the content responses). These contents are transferred separately under the control of a *handshake* protocol. The request and response messages are completely independent. The use of two communication channels is defined in the BVCI standard, but is not present in the PVCI or OCP standards. The request contents flow from the initiator to the target by the activation of the **CMDVAL** and **CMDACK** signals, and the response contents flow from the target to the initiator, by the activation of the **RSPVAL** and **RSPACK** signals.

When the **CMDVAL** signal is activated it means that the Initiator is requesting a read data or is sending a write data to the Target. The activation of the **CMDACK** signal by the Target, in answer to the request of the Initiator, indicates that the transfer can be carried through. Then the Target actives the **RSPVAL** signal indicating a desire to execute the transfer of a response data for the Initiator. The Initiator, in turn, actives the **RSPACK** signal to indicate that it is ready to receive the content response. After the receiving the content for the Initiator, the transference is finished [4].

2.3 OCP Interface Design

The OCP is a standard that consists of a set of signals and communication protocols. Thus, for an IP (Intelectual Property) to be compatible with the OCP standard it is necessary that it follows a minimum set of rules, like having, at least, the defined signals as "basic signals" [2], always keeping the defined protocol and obeying the signals temporization established by the standard [3]. To establish a communication, the Master unit sends its request through the **MCmd** signal, and the Slave unit answers, whether it can or not accept this solicitation, through the **SCmdAccept** signal.

The basic set of signals of OCP standard also defines three buses, the **MData** and the **Sdata,** for sending of data and the **MAddr** for sending the reading address and the **SResp** signal that the Slave sends for the Master to indicate that the reading request resulted in a valid data [5]. Tab. 1 shows all interfaces synthesis results.

3. SYNTHESIS RESULTS

Table 1 presents the results obtained for the VHDL description synthesis of the PVCI, BVCI and OCP interfaces [4], [5] directed to the EPF10K130EQC240-1 FPGA. The synthesis results of a JPEG compressor are also presented. The comparison between these synthesis results allowed the construction of a more accurate idea about the impacts that the designed interfaces will cause in the JPEG compressor.

The comparisons between the designed interfaces indicate that the OCP has an implementation complexity similar to the PVCI interface, which is the simplest interface of the VCI standard. Both interfaces have a very small consumption of logic cells and can operate in a frequency that is very closed with the maximum frequency allowed by the target device.

Table 1 – Synthesis Results

Hardware Block	Logic Cells	Frequency (MHz)	Period (ns)
Initiator PVCI	19	121.9	8.2
Target PVCI	8	333.3	3
Initiator BVCI	50	188.7	5.3
Target BVCI	26	163.9	6.1
Master OCP	28	333.3	3
Slave OCP	20	263.2	3.8
JPEG Compressor	4,363	31.1	32.2

However, in the comparison between the OCP and BVCI interfaces, we can notice a consumption of logic cells approximately twice higher by the BVCI interface. The operation frequency of the OCP interfaces is higher than the BVCI interfaces operation frequency, as showed in table 1. This best results founded to the OCP interfaces can be explained due to the fact that the BVCI interface owns a more complete and complex protocol.

From table 1 it is possible to notice that the PVCI, BVCI and OCP interfaces use about 70 times less logic cells than the JPEG compressor. Thus, it is possible to estimate that the insertion of these interfaces in this hardware block will generate a very small impact in terms of resources consumption. From the operation frequency point of view it is possible to notice that all designed interfaces could operate in speeds much superior than the JPEG compressor speed. All interfaces could operate in a frequency that is five times superior to the JPEG compressor frequency. These results indicate that the JPEG compressor will not present losses in its maximum operation frequency with the utilization of the designed PVCI, BVCI and OCP interfaces.

4. CONCLUSIONS AND FUTURE WORKS

This work presented the design of the PVCI, BVCI and OCP hardware reuse interfaces. Comparing the synthesis results of the interfaces, it is possible to note that the PVCI interfaces use the lowest amount of LCs, the BVCI interfaces use the highest amount of LCs and OCP interfaces is in an intermediate position in use of LCs.

Analyzing the operation frequency results, the PVCI and OCP interfaces exhibit very similar results. The BVCI interface has a lower performance due to its more complex communication protocol.

The comparison with the JPEG compressor synthesis results indicated that the use of LCs, in the worst case, is around 70 times less to the designed interfaces than to the JPEG compressor. In relation to the operation frequency we have in the worst case the interfaces operating approximately at five times faster frequency than the maximum operation frequency of the JPEG compressor. This result is very important because it indicates that the application of the reuse interfaces designed in this work in designs like the JPEG compressor would not cause losses in terms of operation frequency and would cause a minimum impact in terms of resources use.

5. REFERENCES

[1] M. Keating e P. Bricaud, *Reuse Methodology Manual for System-on-a-Chip Designs,* 2ª ed., Kluwer, EUA, 1999.

[2] OCP – Iternational Partnership. *Open Core Protocol Specification (Release 1.0).*2001350.

[3] VSIA. On-Chip Bus Development Working Group. *Virtual Component Interface Standard (OCB 2.1.0).* Março 2000.

[4] T. Silva, L. Agostini. Desenvolvimento e validação das Interfaces BVCI para o Reuso de Blocos de Hardware In: SIM2004 - XI Workshop Iberchip, Salvador, Brasil, Março 2005.

[5] M. Porto, L. Agostini, "Projeto, Síntese e Simulação das Interfaces de Reuso de hardware do Padrão OCP (Open Core Protocol)", XI Workshop Iberchip, Salvador, Brasil, Março 2005.