TÉCNICO LISBOA

Development of an Ethernet/UDP IP Core

Guilherme Batista Guerreiro

January 8, 2023

Project Report (PIC2)

Department of Electrical and Computer Engineering

Supervisors:

José Teixeira de Sousa

Examiners:

Abstract

The Abstract is a summary of the work written for specialists. Its ideal size is one-third of a page, and it should quickly explain the problem, the current work, the proposed solution, the results obtained, and why they improve the state-of-art.

Contents

List of Tables

List of Figures

1 Introduction

This section should be written *after* the main sections are concluded. The suggested order for writing this document is the following:

1. The Background knowledge section: this section is mandatory for both PIC2 and the dissertation, which can reuse and improve it.
2. The Present Work description section: if this is the PIC2 report, this section may not exist if one still has not done any work. In the dissertation, this section is mandatory.
3. The Experimental Results section: if this is the PIC2 report, this section may not exist if one still has not done any work. In the dissertation, this section is mandatory.
4. The Introduction section: this section.
5. Write the Conclusion section.
6. Write the Abstract section.

In this section, one should write the following:

- Motivation: explain the problem one is trying to solve and why it is important. Describe the main existing works on this topic (summarize from the Background section).
- Objective: Explain this work's goal and how it tackled the problem to achieve it. (summarize from the Present Work section).
- Document outline: Describe in one paragraph the contents of each main section of the document.

This document should include tables and figures to ease the explanation. All tables and figures must have a caption, and a label to be referenced in the document and further explained. The reference to the table or figure is mandatory and unreferenced tables or figures should be removed. An example is shown in Table 1.

Table 1: Example table

Name	Direction	Width	Description
clk	INPUT	1	System clock input.
rst	INPUT	1	System reset, asynchronous and active high.

2 Background

2.1 Intellectual Property core

An IP core, or intellectual property (IP) core, is a pre-designed and tested digital logic circuit that can be easily integrated into a larger system on a chip (SoC) or field-programmable gate array (FPGA). IP cores are designed to perform specific functions, such as data processing, communication or system control, and can be used to quickly add these functions to a design without having to design them from scratch.

2.1.1 Make vs buy decision

The decision of whether to make or buy an IP core for a hardware design can depend on a number of factors. This factors range from cost, time-to-market, quality, customization and retaining the intellectual property of the product.

Developing an IP core in-house can seem cheaper as opposed to buying it from a third-party company in the first place. However, in the long run, resource and time costs for designing and testing the core may become greater than buying. Buying can also lead to significantly lower time-to-market.

Buying an IP core from an established developer can give the guarantee of a higher quality product with less defects, given that the company specializes in core design and continuously works on perfecting their products.

In cases of very specific uses, where there aren't any commercially available cores, it may be needed to develop a custom design in-house. Making this choice gives the benefit of maintaining the IP rights to the core and opens the possibility of later retailing it.

Ultimately the decision of making vs buying will depend on the specific needs of the project and its final goal.

2.1.2 IP core Business Model

The IP core business model is a business strategy that focuses on creating, licensing and selling Intellectual Property assets. Dealing in the IP business has the advantage of being able to generate revenue from selling use licenses of IP assets, without ever having to actually manufacture the product. Companies in the IP market can remain focused on researching and developing new products without worrying with production and distribution of said products.

2.2 The Internet of Things

The Internet of Things (IoT) is generally defined as the extensibility of network, computation and sensor capabilities to everyday objects, typically not capable of such features. This allows for an inter-connectivity between this devices and the rest of the internet, making them capable of generating, exchanging and consuming data between themselves and the cloud.

2.2.1 Network Interface IP cores in IoT

Even though IoT devices exist across all types of applications and use cases, all of them share the need to be connected. Network Interfaces provide this necessary connectivity through hardware and software designed to handle the networking protocols.

The use of IP Network Interface cores greatly simplifies the design and development of IoT devices, as they provide a tested solution that can be easily installed in the system through standard protocols. This IP cores also ensure that the device is interoperable with other systems, using standard networking protocols.

Given the variety of applications for IoT devices, it would be unfeasible for developers to design a new interface for each device. Luckily there already exist a vast array of Network Interface IP cores that are specialized and optimized in different aspects that range from speed of connection to power-saving utilities or even all sorts of network protocols.

2.3 Network Protocol Layers

As defined in the OSI (Open System Interconnection) reference model, the Network Protocol is comprised of 7 layers, where each layer is responsible for a specific aspect of communication.

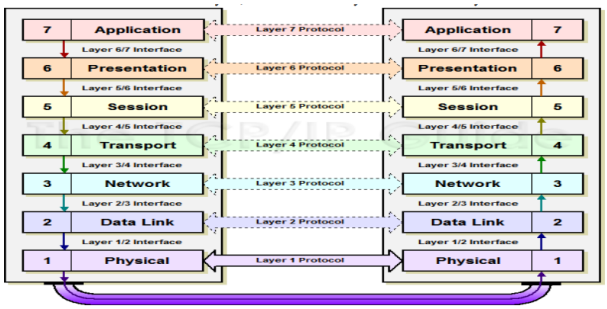

Figure 1: Network Protocol Layers

As seen in figure 1, layers 1 (Physical), 2 (Data Link) and 3 (Network) are at the lower levels of the network protocol. As such, Network Interface IP cores mostly belong within these layers.

2.3.1 Physical Layer

Normally implemented with a dedicated chip or a specialized IP core, this layer is responsible for defining the physical and electrical specifications between the device and the transmission medium. The modulation (or conversion) of information is also done at this level, depending on the physical communication channel.

2.3.2 Data Link Layer

The second lowest layer of the network protocol, the data link layer is responsible for providing reliable transmission of data between adjacent devices, using a set of protocols that are built on top of the raw and unreliable transmission service provided by the physical layer. To ensure reliable transmission, the data link

layer performs error detection and correction, typically using a Cyclic Redundancy Check (CRC). The Ethernet protocol runs at this layer.

2.3.3 Network Layer

While the previous 2 layers deal with transmitting/receiving data and checking for its integrity, the network layer is responsible for addressing the packets before transmission. Every device has a specific address with which it is known on a network and by effectively addressing each packet, the information can easily reach its intended destination within a network. This addressing is done through the Internet Protocol (IP protocol), that runs in this layer.

Specific UDP related hardware runs on this layer, on top of the IP protocol, and is responsible for dealing with low-level details of assembling and disassembling UDP packets.

2.3.4 Transport Layer

The transport layer is responsible for a number of things. It establishes, terminates and maintains connections, implements congestion, flow and error control mechanisms and most of all it is through this layer that the application programmers interact with the lower layers, so it must run the necessary software drivers for the underlying hardware.

2.4 IOBundle Ethernet core

citation tests [1] [2]

3 The Work

This section, or section group, should explain in detail **the solution** proposed by this work to solve the problem formulated in Section 2, why it is innovative and adds to the state-of-the-art. More sections containing partial descriptions of the solution may be added if that improves the explanation.

4 The Results

In this section, one should explain in detail the experiments designed to show the applicability or superiority of the proposed approach and the results obtained. Use tables or figures to facilitate a rapid apprehension of the results.

For example, if the goal is to design a new platform to accelerate a software application, the execution times of the application running on the existing and new platforms should be presented.

The resources used should be detailed: the frequency of operation, memory size, power, energy consumption, and communication bandwidth are examples. For FPGA implementations, for example, the number of block RAMs, LUTs, and DSPs should be presented in a Table for the configurations studied.

5 Conclusion

In this section, one should summarize the work: the problem, the current work, the proposed solution, the results obtained, and why they improve the state-of-art. Repeating what is written in the introduction is not a problem and is desirable, but one should try to use different and complementary explanations to help the user. Another vital part of this section is to explain the future work that can be done as a follow-up to this work.

References

[1] *Accelerating PNG Encoding in Hardware*, Lisbon, Portugal, Feb. 2020. Zenodo. doi: 10.5281/zenodo.3679358. URL https://doi.org/10.5281/zenodo.3679358.

[2] R.-V. Foundation. RISC-V Cores and SoC Overview, 2020. URL https://github.com/riscv/riscv-cores-list.