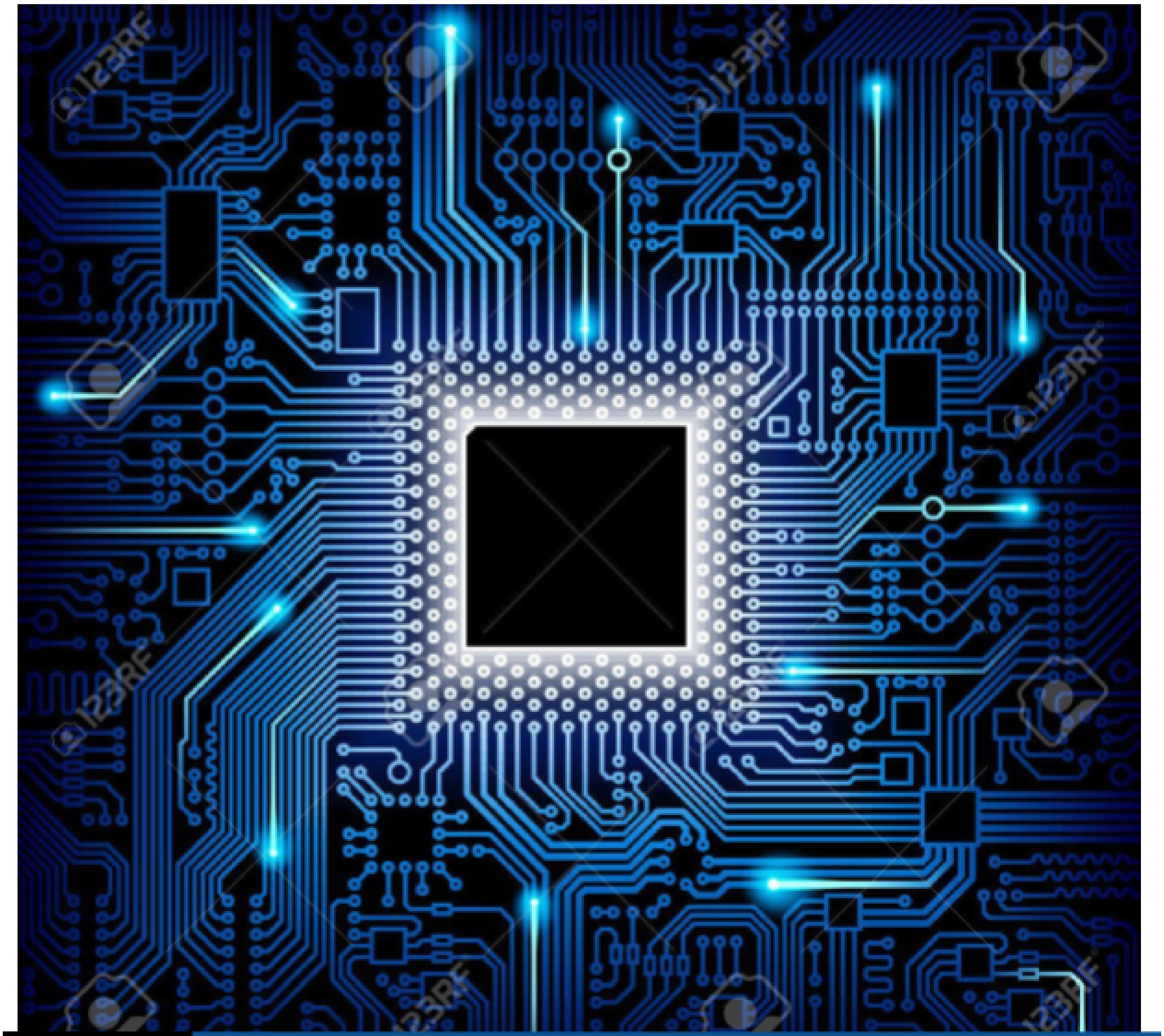
Universitat Autònoma de Barcelona

Facultat d'Enginyeria
Grau en Enginyeria Informàtica
Menció en Enginyeria de Computadors



TREBALL DE FINAL DE GRAU

AXI4 INTEGRATOR

Gerard Badia Arderiu

Directed by
Ricardo Martínez
Lluís Terés

Abstract

This project consists in the design of a tool which will be capable of interconnecting different components of a SoC (System-on-Chip) through their respective ports. This tool will be written in Python mostly but it will generate Verilog code as well in order to, later, connect the components. The components will be given in XML, using the protocol IP-XACT provided by Accellera. The user will be capable of editing the connections as well as making new ones.

Context

The project comes from a larger work with the objective of making an app user-friendly capable of interconnecting modules with AXI4 similar to apps that already exists and are available in the market. These modules, though, use the royalty-free RISC-V architecture. The project is being developed for different parties including the Autonomous University of Barcelona.

Objectives

There are different parts for this project. Firstly, there's the need to make an XML interpreter. To do so, as mentioned before, we will use the IP-XACT protocol which will facilitate the reading of the document and will give us a guide on how the information is stored. The information read will be stored in temporary variables by the program. Classes to organize the different information (modules, ports, connections, parameters) will be needed. Secondly, there will be a simple interface in which the user will be able to tell the program what to do. The mentioned interface will be done via the terminal because it is not the objective of this project to make it user-friendly. In third place, the program will make the connections that the user desires and write a Verilog code that will be written in a file, which will be stored in the proper directory. In order to accomplish this, the Python program will have to be the one able to write the appropriate code for the connections with the desired ports.

Tasks

First, we need to make a function capable of reading the modules written in XML using the IP-XACT protocol. Those modules may contain different amounts of ports, parameters and connections. The mentioned function will need to store each module with its ports and parameters as well as possible connections with other modules. This first task will be very time costly since an entire module needs to be both read and stored in the appropriate structures which also will need to be created.

In second place, interaction with the user will be required so a simple interface in which the user can input the information he/she desires needs to be done. The interaction with the user will be done through the terminal. Overall it is a straightforward task so no much time will be spent in this.

The third task will be connecting modules. Those modules will be connected both via Verilog (written in a file) and in the program (stored in the appropriate structures). This task requires the user to input the desired ports from the desired modules they wish to connect so an interface for that will be needed. Also a directory with the proper name will be created so that the project keeps an organized directory. It will be very time demanding

Fourth and last, organizing the functions to make it more readable as well as adding contingencies in the project so it doesn't crash whenever there's an error and instead notifies the user what the problem was.

Cost

Since this will be a royalty-free project using a royalty-free architecture it will be totally free and the only cost is the human labor of the designer of the project.

Bibliography

IP-XACT User Guide:

https://www.accellera.org/images/downloads/standards/ip-xact/IP-XACT_User_Guide_2018-02-16.pdf