Instituto Tecnológico de Costa Rica
Computer Engineering Academic Area

Proyecto de Diseño en Ingeniería en Computadores

Project Plan

Daniel Moya Sánchez

February 15, 2018

1 Name of the project

Design of Application Specific Instruction Set Processors (ASIPs) for Approximate Computing

2 Name of the institution

Chair for Embedded System (CES), Kalrsruhe Instituteof Technology (KIT), Germany, and Laboratorio Sistemas Embebidos y Electrónica Digital (SEED-Lab) of Instituto Tecnológico de Costa Rica (ITCR).

3 Confidentiality requirements

Due to the academic nature of the project, there are no special confidentiality requirements, however, results will not be published until the end of the project's work

4 Problem description

As the name says it, ASIPs are processors that use an application specific instruction set; this means that, although they can execute a wide range of applications, they are optimized for a specific type, in which ASIPs can execute with a greater energy and speed results compared to a General Purpose Proccesors (GPP). On the other end, these results are not as good as the ones achieved by Application Specific Integrated Circuits (ASICs) but this trade-off comes with a better flexibility of ASIPs above ASICs. Optimizations for an ASIP can be seen in different forms, which include: [CITA]

- Instruction extension: Customized instructions can be made to extend the base Instruction Set Architecture (ISA).
- Inclusion or exclusion of predefined blocks: Not only specific software can be added to extend an architecture but also customized hardware in the form of specialized blocks; also, regular blocks that are not used can be excluded.
- Parameterization: Certain variables like cache sizes or number of registers can be customized to adjust for a specific application.

In a summarized way, ASICs represent a hardware solution to a problem which is very limited and have high costs and a high time-to-market but achieves the greatest performance, contrary, GPPs are seen as a software solution which are very flexible but are the least efficient. ASIPs are in the middle of these two as they balance flexibility and peformance to have a good trade-off between those variables.

ASIPs can also be used to adjust the balance between acceptable amount of error vs the cost (economic, area, execution time) of an application; which is is the main focus of study of the project. Since different types of applications vary signifcantly in their requirements and specifications (e.g. where the error-resilient section is), different ASIPs have to be build for each one of them so that the best balance between cost savings and amount of error is achieved. This project focuses on that goal, to design ASIPs for a set of error-tolerant applications.

The environment in which the ASIPs will be developed consists in several software tools which include Design Compiler and Prime Time, Synopsys, ModelSim, Mentor Graphics, ASIPMeister, CoSy compiler, Xilinx ISE and the hardware platform will be a Xilinx Virtex-V board. Regardless these limitations, the project will allow for a custom hardware components choice with its design for specific sections. Also, for the error tolerant applications found, the software implementation and the tests for the general system will be able to be choosen between different options.

Since approximated computing is in its infancy, it still needs a lot of research and testing, so the users of the ASIPs developed will be the same groups of investigation of which this project is part of; it is expected that this helps to make approximated computing a more solid tendency.

5 Objectives

5.1 General objectives

To explore the design of Application-Specific Instruction Set Processors (ASIP) to be used in error tolerant applications.

5.2 Specific objectives

The project has the following specific objectives:

1. To select 3 error tolerant applications to be evaluated.
2. To develop at least 3 instances of approximated hardware for tolerant error sections for each of the selected applications.
3. To develop ASIPs configurations using specific approximated instructions.

Table 1: Deliverables with the corresponding criteria of acceptance

Name	Description	Criteria of acceptance
Requirement 1.1	Instances of approximated hardware	[criterio]
Requirement 2.1	Configuration of approximated ASIPs	[criterio]
Requirement 3.1	Data of execution time, area and power	[criterio]
Requirement 3.2	Comparison and analysis of the obtained results	[criterio]

4. To evaluate, with the use of approximated instructions, a set of applications in terms of execution time, area and power consumption vs its error for the approximated ASIP.

6 Project stakeholders

Since this is an investigation project, there is only a few stakeholders, who are described below:

- Jorge Castro: The project's supervisor, has the general idea about the project itself and guides its course. He wants to create new knowledge with the use of ASIPs, so that this area grows and future processes of creating approximated applications become more automated.
- Sajjad Hussain: He works with Jorge Castro on the general guidance of the project, helps with any issue on the server in Germany so that the process of using the software platform (ASIPMeister, Dlxsim, etc.) remains smooth. He has the same interest as Jorge Castro with the project.
- Jeferson González: This project's supervisor at the ITCR, and the person in charge of the SEED laboratory, from where he occasionaly provides some resource (like the equipment in the lab) or guidance.

7 Solution description

8 Deliverables and criteria of acceptance

The expected deliverables are the following:

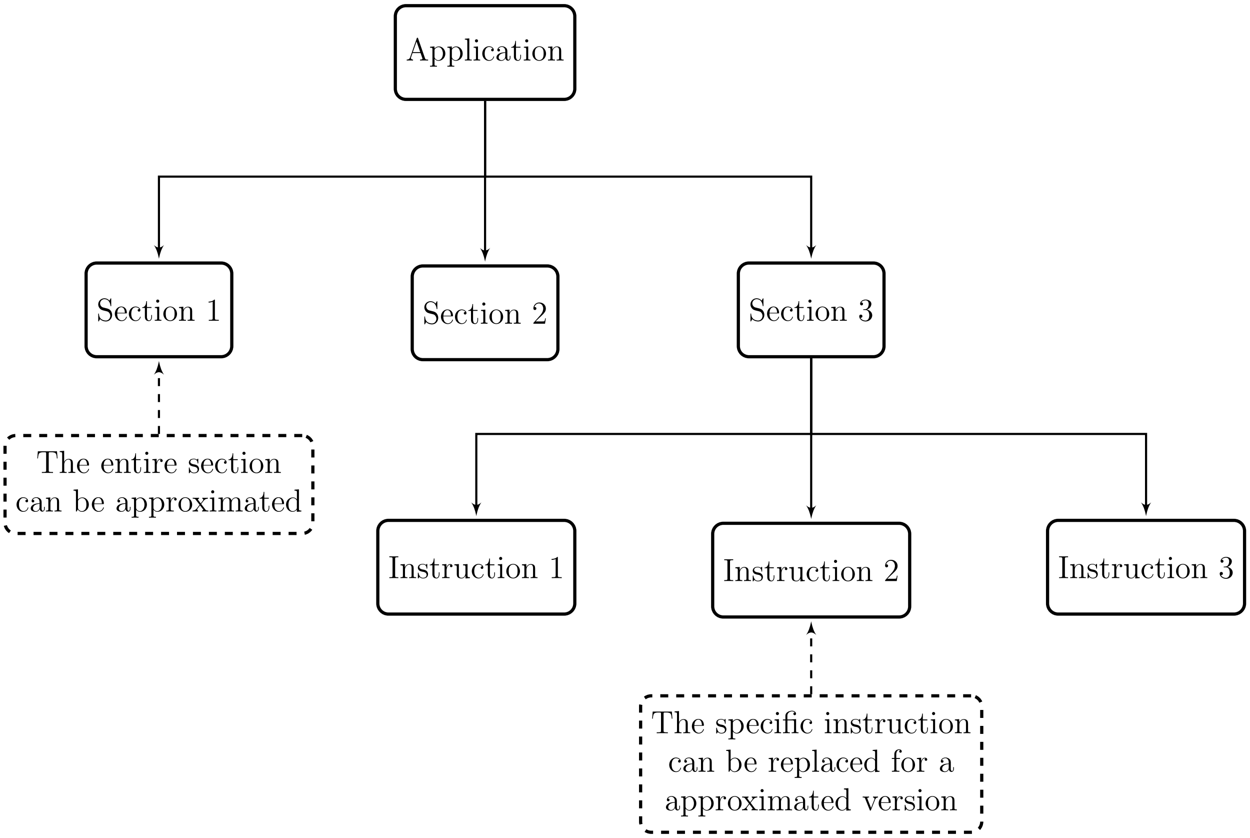
Application
Section 1
Section 2
Section 3
The entire section can be approximated
Instruction 1
Instruction 2
Instruction 3
The specific instruction can be replaced for a approximated version

Figure 1: Expected situation to solve with this project

Table 2: Risk analysis

Risk	Probability of occurrence	Impact (hours)	Risk exposure (hours)
Illness or any special medical condition	0.5	8	4
General server errors (missing files, permission restrictions, etc)	0.6	24	14.4
Delays when acquiring the hardware	0.25	8	2

Table 3: Activities and effort budget

ID	Activity	Engineering hours	Risk reserve (hours)	Total (hours)
001	Get to know the software platform	24	2	26

Table 4: Schedule for the entire project

	Week															
Activity	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Reading the corresponding literature about the project and understanding the general idea of ASIPs	■															
Execution of the laboratory script to get to know the software tools like ASIPMeister, Dlxsim, etc	■	■														
Delivery of the "Plan Project" document		■														
Delivery of the "Requirements" document			■													
Delivery of the "Design" document				■												
Delivery of the "Final report" document																■

9 Risk analysis

10 Activities and effort budget

The table 3 takes in consideration a total of 216 engineering hours.

11 Schedule

References