THE NATIONAL UNIVERSITY OF LESOTHO

Department of Mathematics and Computer Science

CS3520 - Computer Organisation and Architecture I

Project Handout

16th September 2025

1. Introduction

1.1 Context

The Fourth Industrial Revolution (4IR) is reshaping economies worldwide through advances in artificial intelligence, connectivity, and digital services. For countries such as Lesotho and Africa at large, the challenge is not only to be consumers of these technologies but also to actively design and develop solutions that are adapted to local realities.

In this project, students will design and simulate a RISC-V–based processor targeted at next-generation mobile telephony devices with built-in AI capabilities. Mobile devices are central to Africa's digital transformation, enabling communication, financial inclusion, and access to services. By studying representative AI-powered applications—such as voice recognition, biometric security, and intelligent connectivity—students will learn how to map real-world requirements to processor design, adapt the ISA, and create a microarchitecture capable of powering these critical applications.

1.2 Aim and Objectives

The aim of this project is to design, simulate, and evaluate a custom RISC-V–based processor for low-cost AI-enabled mobile phones, beginning from an analysis of representative applications and workloads and proceeding through ISA customization, microarchitecture design, and simulator development and testing. The specific project objectives are:

- Study the application context to identify representative applications and workload requirements for low-cost AI-enabled mobile phones.
- Develop representative high-level applications (e.g., in C/C++/Java) that characterise the identified workloads.
- Define and customise a RISC-V ISA that responds to the identified application and workload requirements.
- Design a microarchitecture that implements the customised ISA, including datapath, control, pipeline, and memory hierarchy.
- Implement a simulator that models the behaviour of the designed microarchitecture.
- Write assembly test programs to validate the ISA and microarchitecture.
- Run and evaluate the test programs on the simulator, analysing correctness, performance, and efficiency.
- Document and present the project through a final report, poster, source code repository, and oral presentation.

2. Project Deliverables

Deliverable	Task	Description	Deliverable
Week 5	Application Context & Workload Analysis	Study the mobile AI telephony context; identify representative applications and workload requirements.	Domain Analysis Report
Week 5	High-Level Application Prototypes	Develop simple high-level programs (C/C++/Java) to characterise identified workloads. Include brief notes on rationale.	Prototype Programs
Week 6	ISA Definition & Documentation	Define a customised RISC-V ISA suited to the workloads. Document base subset, extensions, encoding, and rationale.	ISA Specification
Week 7	Microarchitecture Design	Design datapath, control, pipeline, and memory hierarchy for the chosen ISA. Provide diagrams and explanations.	Microarchitecture Specification
Week 8-9	Simulator Core (MVP)	Implement initial cycle-accurate simulator supporting the core ISA. Must run simple programs and output traces.	Simulator Prototype
Week 10-11	Enhanced Simulator	Expand simulator to support full ISA, hazards, pipeline behaviour, and memory hierarchy. Run preliminary benchmarks.	Simulator v2
Week 12	Assembly Test Programs & Evaluation	Write assembly programs representing workloads. Run and evaluate them on the simulator for correctness and efficiency.	Test Suite & Evaluation Report
Week 13	Final Integration & Presentation	Consolidate all work into a final package. Prepare final report, poster, polished code repository, and presentation.	Final Report, Poster, Presentation, Source Code Repository

3. Project Guidelines

- **Version Control:** All teams must use a standard source code version control platform such as GitHub (or GitLab/Bitbucket). Repositories should reflect collaborative contributions with clear commit histories, branches, and documentation.
- **Agile Project Management:** Teams are expected to follow agile practices, with weekly deliverables and continuous feedback. Tools such as Jira (or Trello, GitHub Projects) should be used to plan sprints, assign tasks, and track progress.
- **Collaboration:** Each group of seven should define and rotate roles (e.g., researcher, ISA designer, microarchitect, simulator developer, tester, documenter). Weekly stand-ups or progress meetings are encouraged.