California State University, Northridge

ECE 422L - Computer Architecture

Fall 2022

Lab 6

Matrix Multiplication with ARM Assembly

March 22, 2023

Instructor: Dr. Shahnam Mirzaei

Mark Attia

**Table of Contents**

**Introduction 2**

**Procedure 2-3**

**Testing Strategy 4**

**Results 4-5**

**Conclusion 6**

**Appendix 6**

**Introduction**

The purpose of this experiment is to learn how to manipulate matrices with our microcontroller. We will initialize them in memory and use previously learned ldr and str instructions to be able to perform different operations on these matrices. We will add, subtract, and multiply the matrices as they are fundamental data structures that we must be familiar with.

**Procedure**

1. Create Project Lab5

2. Ensure all details are correct for compiling code

3. Simulation debugger

4. Drivers for board installed -Using Tiva Core

5. Use sample program provided in manual and test for proper execution

6. Carefully monitor memory activity in step-by-step debug mode

7. Complete task 1.

8. Complete task 2

9. Complete task 3

Diagram

Description automatically generated

**Testing Strategy**

The testing strategy for this experiment comes from being able to understand what should be happening inside of our memory based on what we have learned about using ldr and str to pull from and send to memory. Keil has a useful feature of memory view, where we can simply enter any address we’d like (usually SRAM\_BASE) and view the values at that location. This feature will be our primary method to verify that our matrix operations are correct.

**Results**

Graphical user interface

Description automatically generated with low confidence

Table

Description automatically generated

Figure 1: Registers Task 1

Graphical user interface, text, application, chat or text message

Description automatically generated

Figure 2: Memory Result SRAM\_ BASE

Graphical user interface

Description automatically generated

Text

Description automatically generated with medium confidence

Figure 2: Matrix Multiplication Result

Logic stays identical except for adding a delay to debounce button presses.

<https://photos.app.goo.gl/fs1UeRV7MT87dotH9>

**Questions**

There are benefits and drawbacks for either method of performing these operations. If we use multiple blocks, it allows the CPU to access multiple sets of data at once (in parallel) and therefore increases speed and most likely reduces conflicts as everything has its own dedicated space. However it could still end up being slower as the CPU will have to access the memory more times to reach all of the data that it needs which could be bad if the CPU needs more than one memory location for a single operation. Overall, it would mostly depend on the use case for which one was better, so my preference would usually be to use the one that is easier for my specific application.

**Conclusion**

This experiment allowed us to get refreshed with our ldr and str instructions, as well as begin manipulating larger datasets. It helped us bring together all the skills we’ve been working on in ARM assembly programming and add to it using very import data type of matrices. Doing this will act as a steppingstone to future experiments where we must manipulate even larger sets of data using similar concepts such as loops, ldr, str, and DCW/DCD (memory allocation).

https://github.com/EgyptiansFTW/Computer-Architecure-Lab.git