University of Dhaka

Department of Computer Science and Engineering

CSE-3113 : Microprocessor and Assembly Language Lab

Experiment 4: To implement various types of Arithmetic Operations in ARM Assembly Language

Submitted By:

Name: Mahmudul Hasan

Roll No : 20

Submitted On :

February 27, 2023

Submitted To :

Dr. Upama Kabir, Professor

Dr. Md. Mustafizur Rahman, Professor

1 Introduction

Assembly language programming is a low-level programming language that enables programmers to write instructions directly for a computer's processor. The Cortex M4 processor is a 32-bit micro-controller commonly used in embedded systems. This report aims to provide an understanding and familiarization of register-based assembly programming for the Cortex M4 processor, specifically for arithmetic operations. In this lab, we aim to understand and familiarize with register-based assembly programming for the Cortex M4 processor for arithmetic operations.

2 Objectives

The objectives of this lab are

1. To gain an understanding of register-based assembly programming and its benefits for the Cortex M4 processor.
2. To become familiar with the Cortex M4 processor's general-purpose registers and their use in arithmetic operations.
3. To learn the instruction set for arithmetic operations in assembly language programming, including addition, subtraction and multiplication.
4. To gain hands-on experience in writing assembly code for arithmetic operations using the Cortex M4 processor's registers.
5. To understand the importance of register-based assembly programming for optimizing code performance and efficiency in embedded systems.
6. To develop the skills necessary to write efficient and effective assembly code for arithmetic operations on the Cortex M4 processor.

3 Theory

In register-based assembly programming, instructions are executed using processor registers instead of memory locations. This approach provides faster and more efficient execution compared to memory-based programming. The Cortex M4 processor has several general-purpose registers (r0 to r15, specifically r0 to r12) that can be used for arithmetic operations. The Cortex M4 processor supports a variety of arithmetic operations, including addition, subtraction, multiplication, and division. These operations can be performed using the appropriate instruction set in assembly language programming. In assembly programming, operands for arithmetic operations are typically stored in registers. The results of these operations are also stored in registers.

4 Methodology

1. Set up the development environment for the Cortex M4 processor, including the appropriate software tools and libraries.
2. Write assembly code for arithmetic operations, including addition, subtraction and multiplication.
3. Measure the performance and efficiency of the code by timing the execution of the arithmetic operations and comparing it to other implementations.
4. Evaluate the code and analyze the results to understand the register-based assembly programming for arithmetic operations on the Cortex M4 processor.

5 Result

5.1 Task 01

An assembly language to perform all the logical operations (AND,OR,NOR,NAND,XOR,XNOR) on two 16-bit variables. Repeat it for two 32-bit variables.

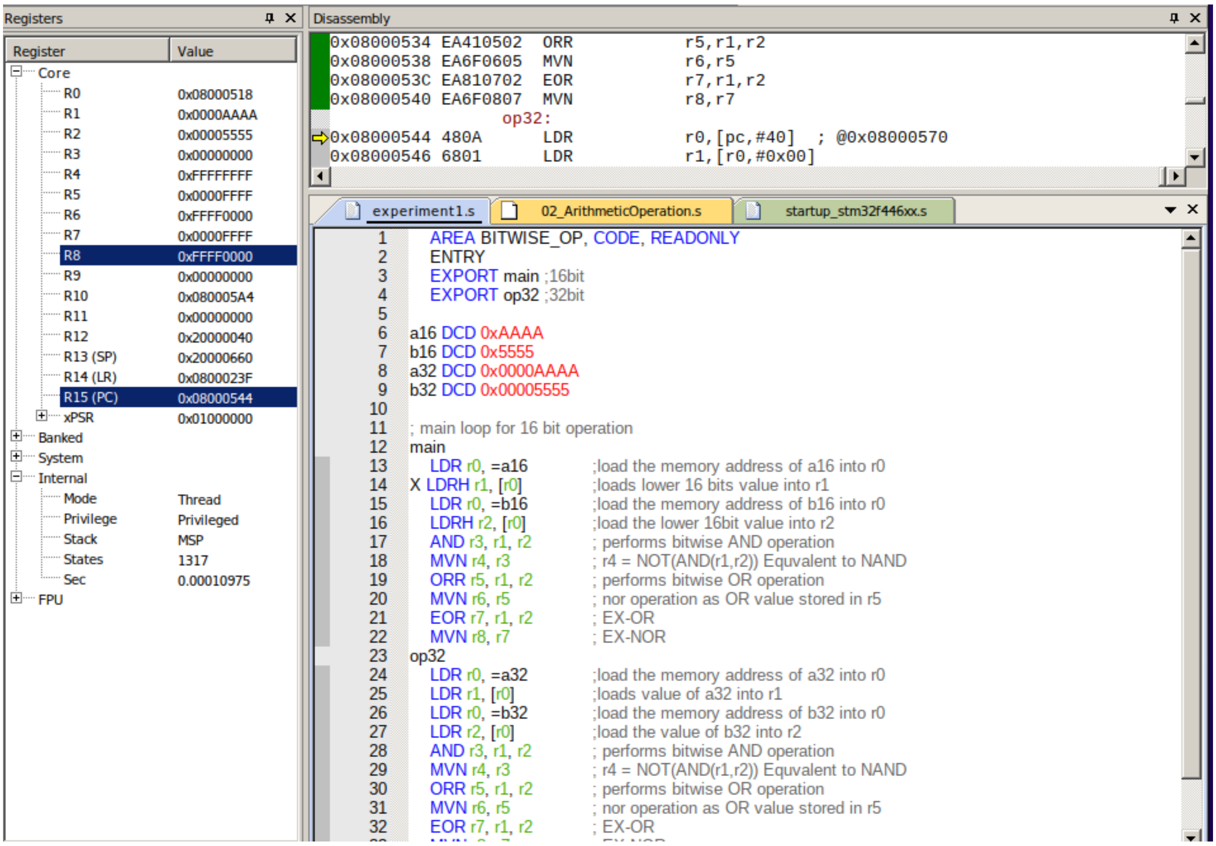

Figure 1: BitWise Operation

5.2 Task 02

An assembly language to perform all the shift operations (LSR, ASR, LSL) on a 32-bit variable.

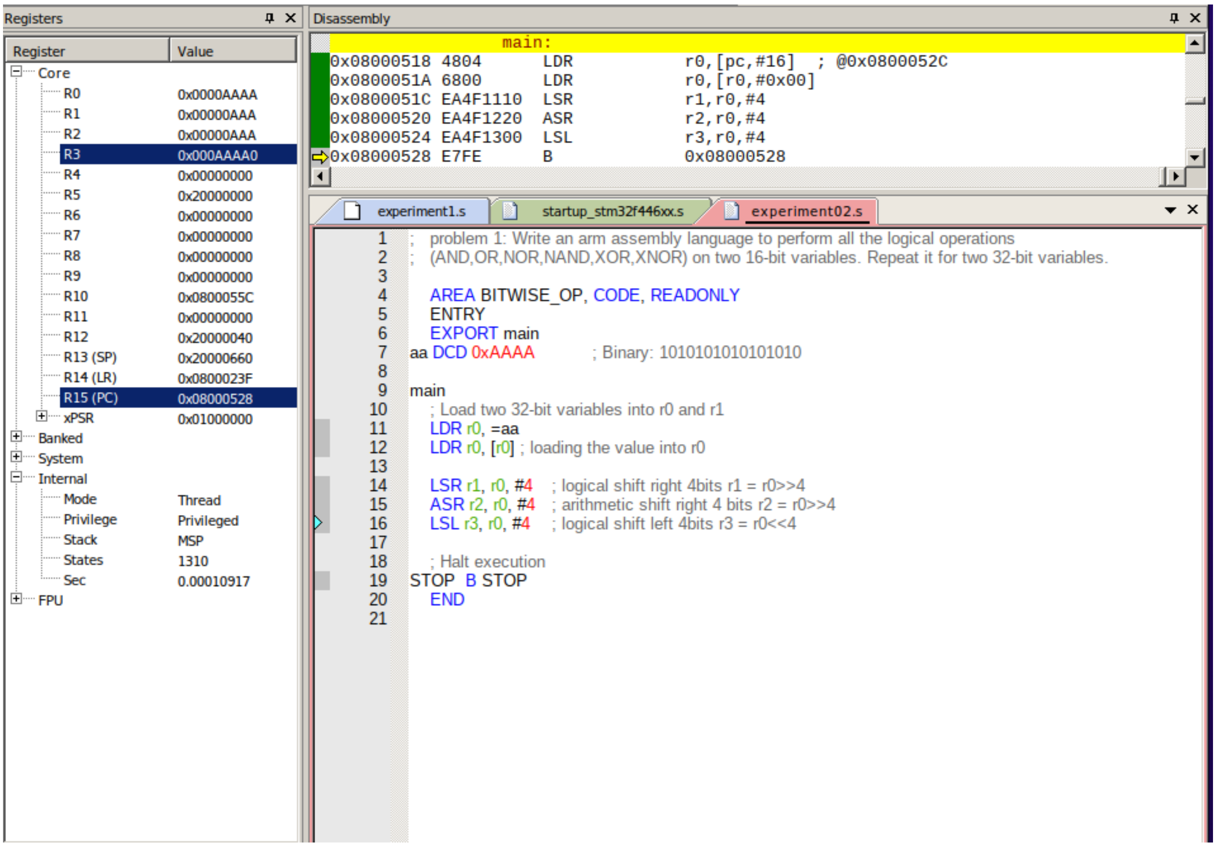

Figure 2: Shift Operation

5.3 Task 03

An assembly language to perform all the arithmetic operations (Addition, Subtraction, Division and Multiplication) on two variables. Restrict input values to avoid overflow. Repeat the same operations to handle overflow.

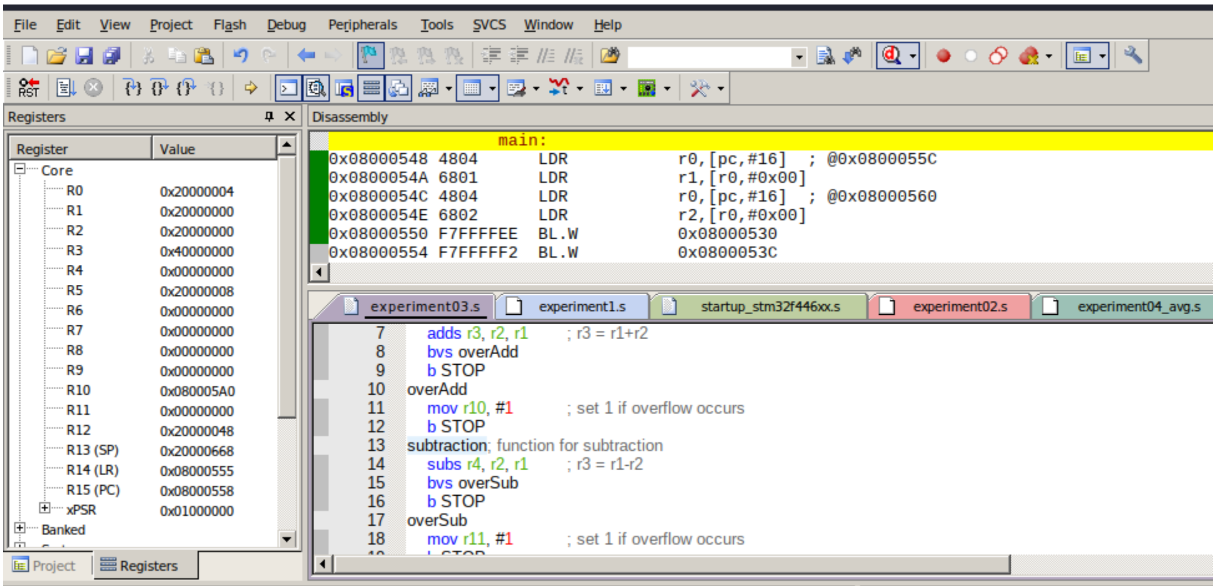

Figure 3: Program for Overflow Handeling

5.4 Task 04

An assembly language program to find the average of n numbers.

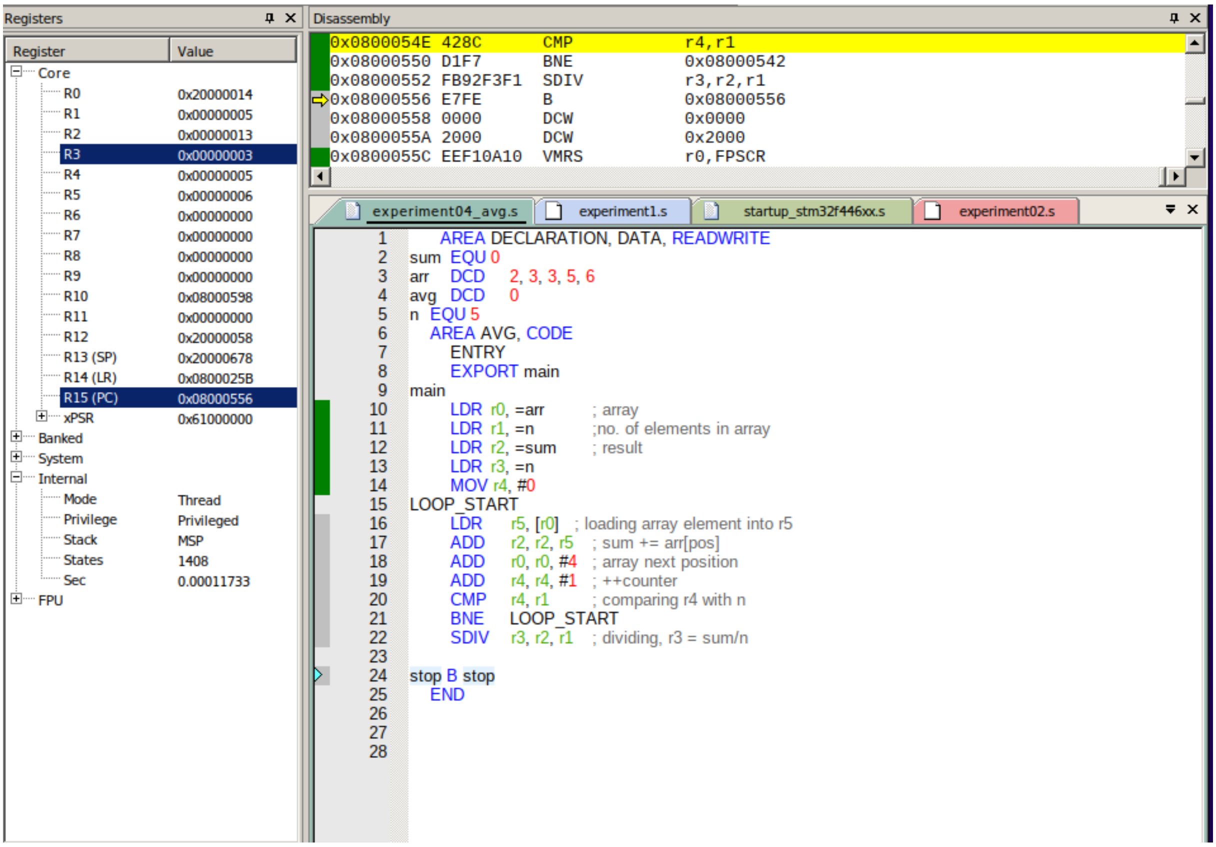

Figure 4: program to find the average of n numbers

5.5 Task 05

An assembly language program to find the largest among different numbers.

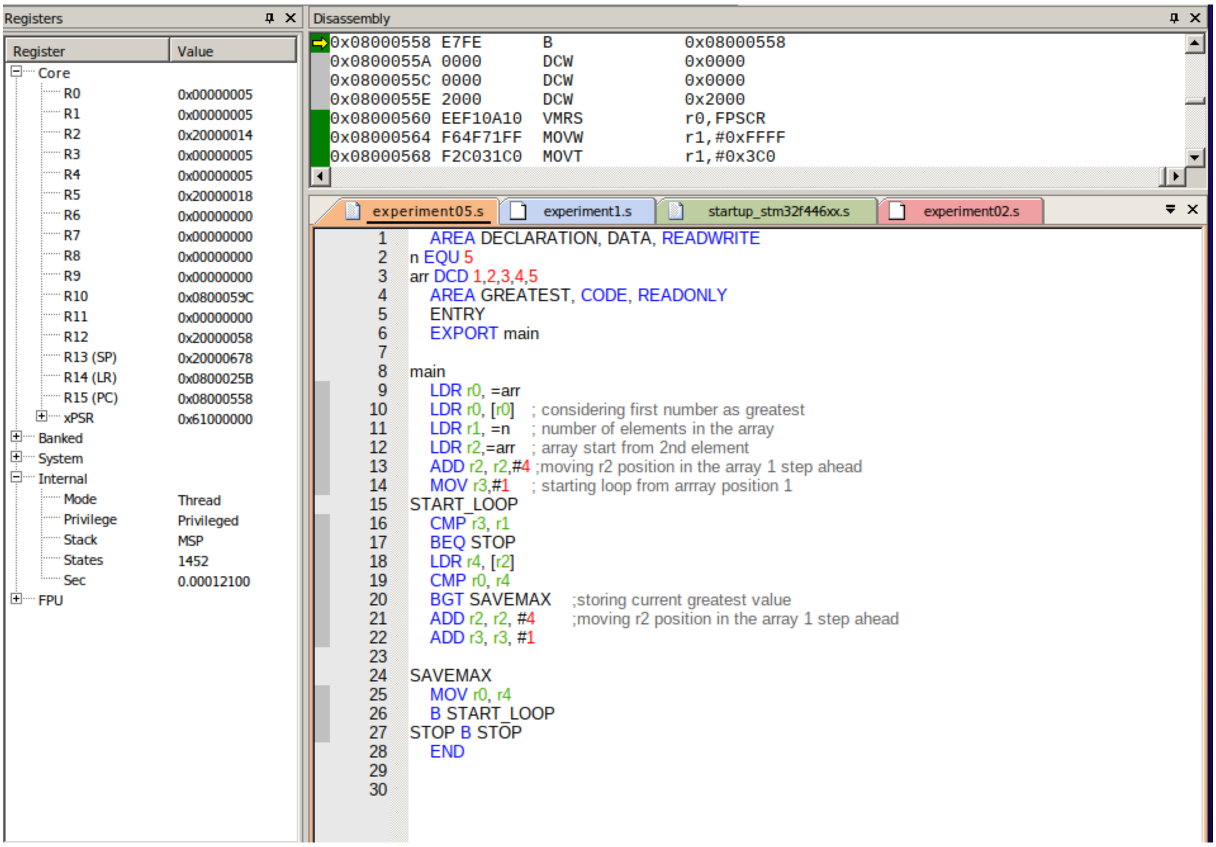

Figure 5: program to find the largest among different numbers

5.6 Task 06

an assembly language program to find the average of n numbers using function call.

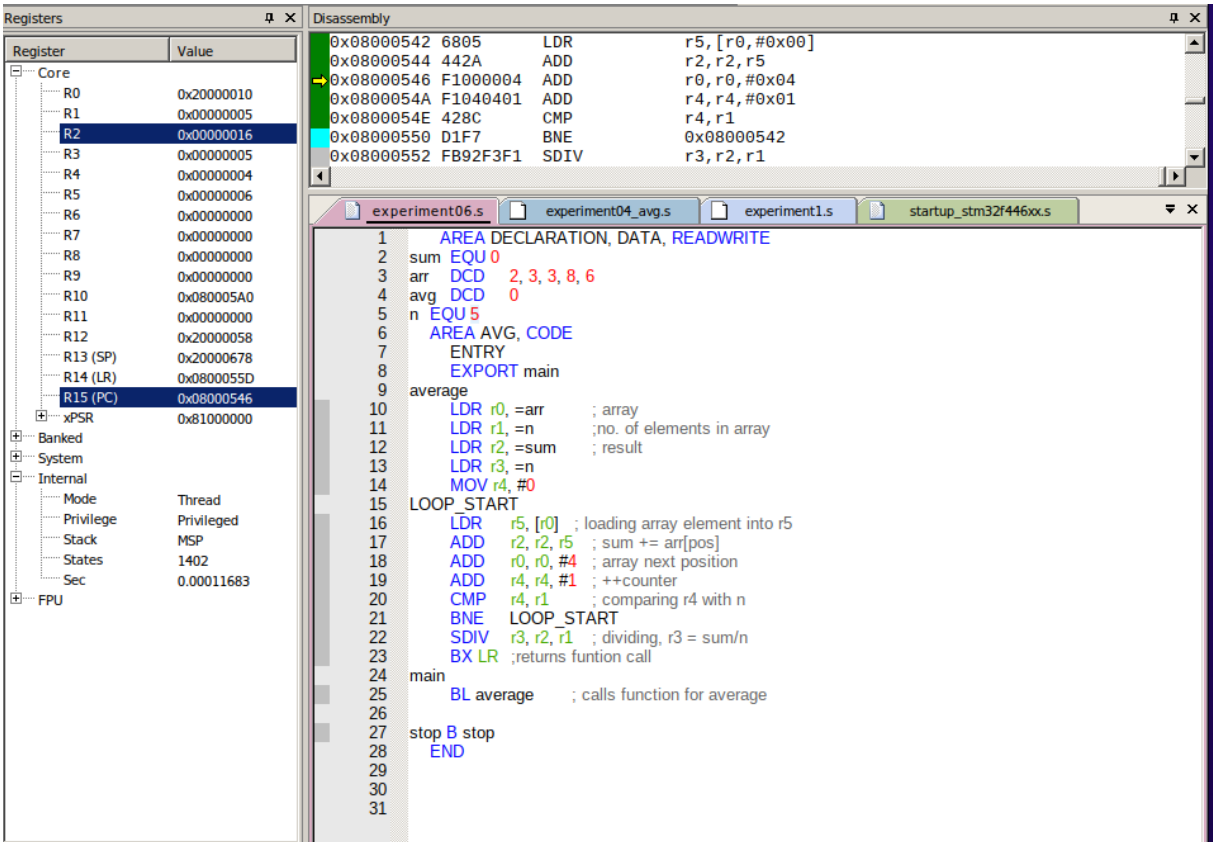

Figure 6: Program for finding average of n numbers using function call

6 Experiences

During the lab, we've to face various challenges to setup Keil uVision IDE including solving debugger issues. After all, we could complete the experiment successfully and got output as expected. Before that lab, we would know about registers and their operations theoretically. But the lab experiment has given us the opportunity to observe running mathematical operations using registers directly.

7 Conclusion

The lab helped us to understand and familiarize with register-based assembly programming for the Cortex M4 processor for arithmetic operations. The results showed that this approach provides faster and more efficient execution compared to memory-based programming, making it a useful tool for embedded systems development.

In conclusion, the lab provides a solid foundation for further exploration of the benefits and limitations of register-based assembly programming for embedded systems development.