[](http://www.kustar.ac.ae/default.aspx)

**Module Name: Microprocessor Systems Laboratory**

**Module Code: ELCE333**

**Laboratory Experiment No. 1**

**Pre-Lab Report**

# Experiment Title: Microcontroller Assembly Program Development

**Partners:**

Amal AlQasimi 100036830

Shaikha AlBesher 100037064

**Spring 2015**

Table of Contents

[Table of Illustrations 2](#_Toc367879991)

[Pre-Lab1: 3](#_Toc367879992)

[Pre-Lab1 Questions: 5](#_Toc367879993)

## 

## Table of Illustrations

[Figure 1: Step 2, location $1000 filled 4](#_Toc367879862)

[Figure 2: step 5, location $1010 filled 5](#_Toc367879863)

[Figure 3: Memory map of MC9S12DP256 6](#_Toc367879864)

[Table 1: The contents of elements at each step 3](#_Toc367879874)

## Pre-Lab1:

The aim of this pre-lab is to simulate and analyze the basics of the assembly language.

1. Simulating the following program.

; Include derivative-specific definitions

INCLUDE 'derivative.inc'

; export symbols

XDEF Entry

; Insert here your data definition.

; code section

Entry:

LDAA #$FF; Initialize Accumulator A with 0xFF

STAA $1000; Store the value 0xFF in memory location 1000

CLRA; Initialize Accumulator A with 0x00

STAA $1010; Store the value 0x00 in memory location 1010

MOVB $1000,$1010; Move a byte from memory location $1000 to $1010

LDAB $1010; Load the content of memory location $1010 to Accumulator B

LDX #$0011; Initialize the index register X with 0x0011

EXG X,B; Exchange the contents of Accumulator B and index register X

1. After simulating the code, we were able to examine the contents of the memory locations 0x1000 and 0x1010, index register X, and the accumulators A and B before and after stepping in each line of the program. Data are filled in table 1:

Table 1: The contents of elements at each step

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Instruction** | **Memory**  **Location**  **0x1000** | **Memory**  **Location**  **0x1010** | **Accumulator**  **A** | **Accumulator**  **B** | **Index**  **Register X** |
| **LDAA #$FF** | -- |  | FF |  |  |
| **STAA $1000** | FF |  | FF |  |  |
| **CLRA** | FF | uu | 0 |  |  |
| **STAA $1010** | FF | 00 | 0 |  |  |
| **MOVB $1000,$1010** | FF | FF | 0 | CB |  |
| **LDAB $1010** | FF | FF | 0 | FF |  |
| **LDX #$0011** | FF | FF | 0 | FF | 11 |
| **EXG X,B** | FF | FF | 0 | 11 | FFFF |

After the first step, at STAA $1000, the memory location is filled, which indicates storing the contents of A in location $1000 as shown in Figure 1. Then, the location $1010 is filled with FF after the execution of the MOVB $1000,$1010 instruction which means moving the contents of $1000 to $1010, the location of $1010 is then filled with FF as Figure 2 shows.

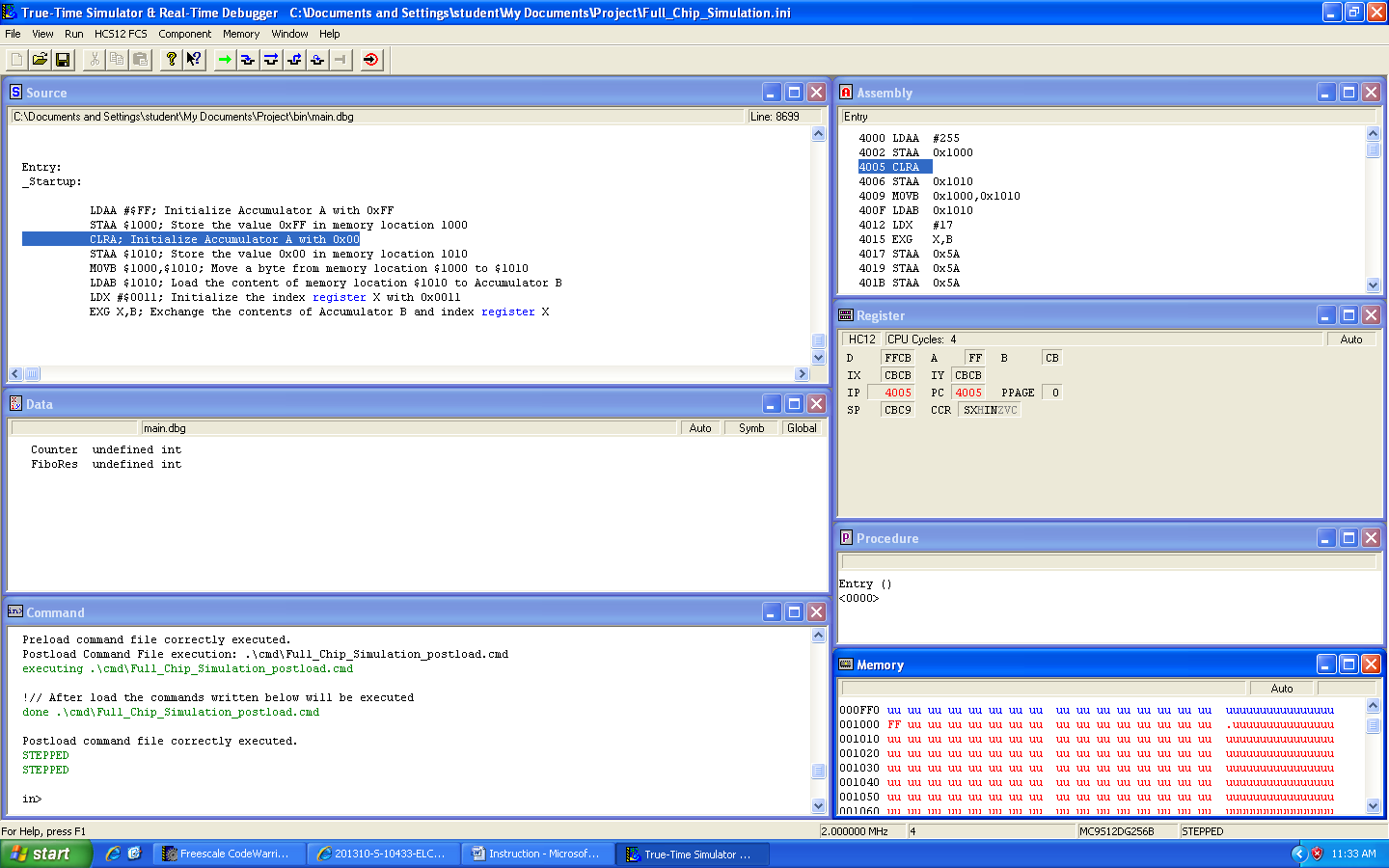


Figure 1: Step 2, location $1000 filled

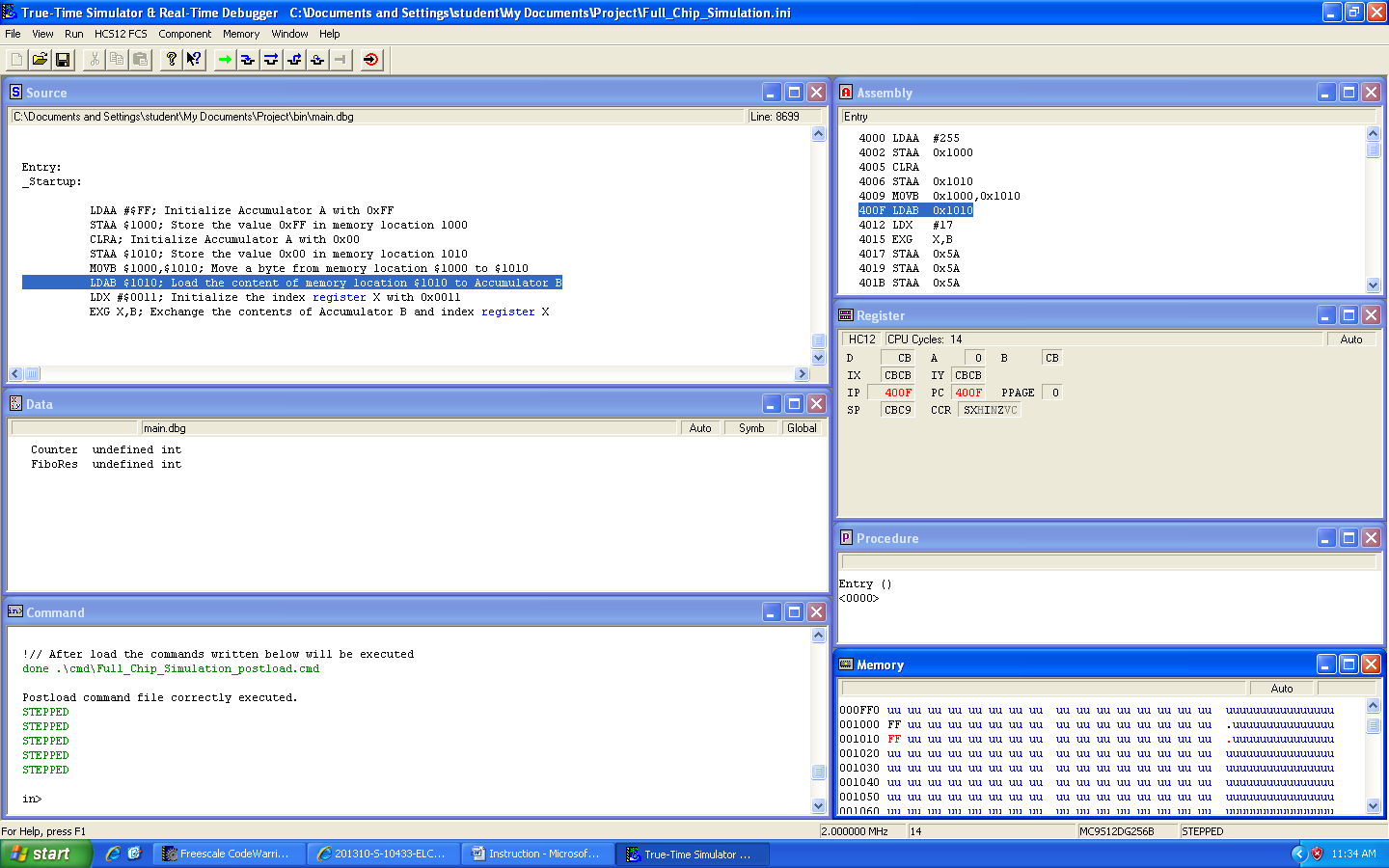


Figure 2: step 5, location $1010 filled

The addressing mode of the following instructions:

1. LDAA #$FF 🡪 Immediate: Loading the accumulator A with $FF

b) CLRA 🡪 Inherent : Clearing accumulator A

c) LDAB $1000 🡪 Extended : Load accumulator B with the contents of location $1000

## 

## Pre-Lab1 Questions:

**1. How much RAM does your microcontroller have? How much EEPROM and how much**

**Flash? Does it have any other kind of memory?**

RAM is 12 K bytes

EEPROM is 4K bytes

Flash EEPROM is 256 K bytes

**2. Draw the memory map of your microcontroller.**

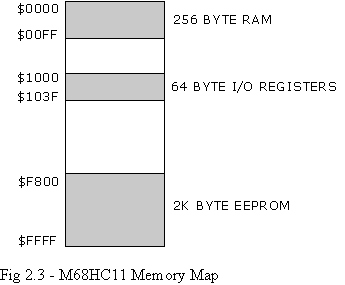


Figure 3: Memory map of MC9S12DP256

**3. What registers does your microcontroller have?**

* 8-Bit Accumulators A & B
* 16-Bit Double Accumulator D
* Index register X
* Index register Y
* Stack pointer
* Program counter
* Condition Code Register (CCR)