G Group: xliser

CISM 314: COMPILER DESIGN:

Lab 2: Assembly Language (Marie Simulator)

Introduction

In this lab we use assembly language using MARIE Machine Simulator. Marie simulator is written in the java language. In program execution the instruction appear on the monitor area and they are highlighted along with memory in the memory area where the instructions is accessing. When executing program instructions, message appear in the message area below the screen. After program execution ends you will see a “program halted normally” or “program halted abnormally” message. If you do not see this message, either your program did not start running, or it is looping and you need to halt it normally.

The Marie Editor

During program execution, we set our origin to 100 that is where our program will be located on our memory area. then, we stored in memory the parameters W it is the parameter to store “X” and “Y” and subtract “Z” that are declared as follows 25, 44 and 22 and the “X” Dec 25 represents the value of X with 25 in decimal form. The output area displays the results, and the halt operator is used to terminate the program. After the code was converted to assembly language by pressing assemble then assemble current file then saves the file as code1.max.

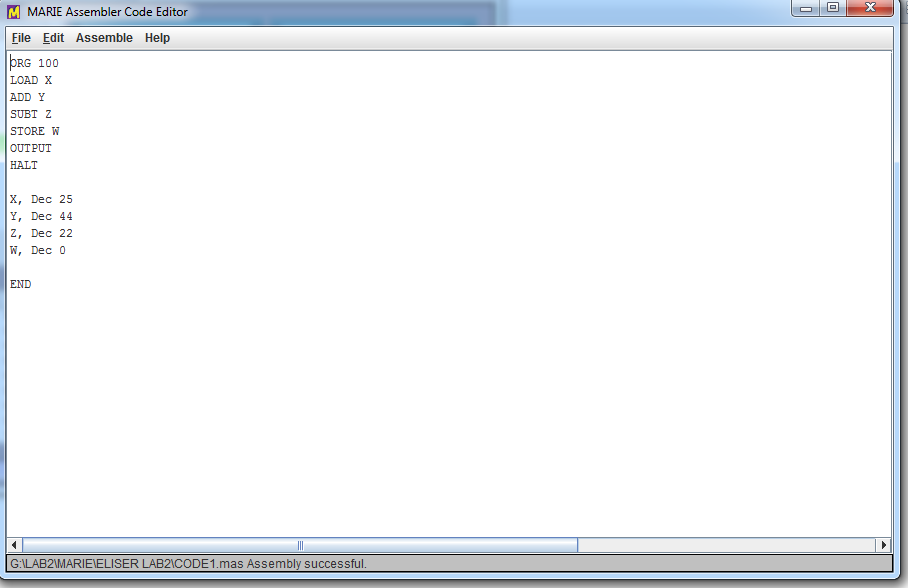


Figure 1: MARIE Assembler Code Editor

The assembled code was then loaded in the MARIE Simulator by selecting File then Load Menu from the Simulator. The program monitor screen displayed the assembly language status in Hexadecimal form. The left side of the monitor shows the program addresses and the statements which that have been executed by the simulator is highlighted in a green colour and when the program is loaded for the first time the green highlight will be on the statement of the first address on the program. The green highlight will move to different memory locations once the program runs, accessing many different storage locations.

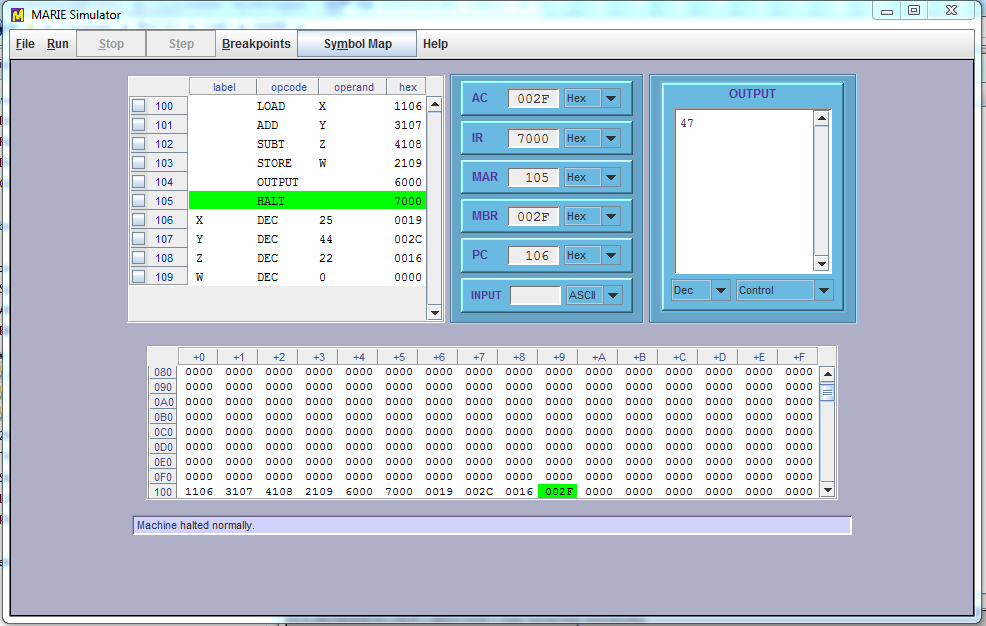


Figure 2: MARIE Simulator

The Data Path Simulator shows data movement in the MARIE machine. The MARIE data path runs only programs that have been assembled by the MARIE assemble and it load or executes object code that have been saved with the .MEX extension. In the process of program execution the instructions that are been executed are highlighted in the program monitor. The corresponding data movement operation is then animated in the graphical section of the screen. Every time a component is participating in the data movement operation it is rendered in a bright colour.

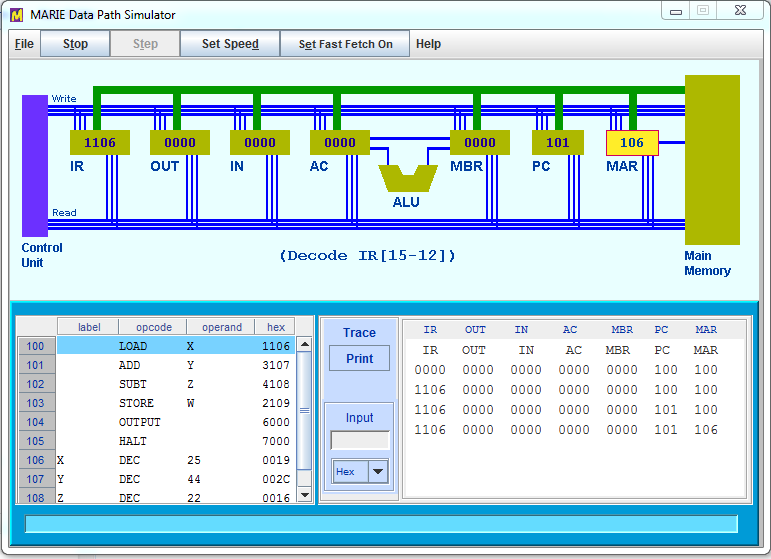


Figure 3: MARIE Data Path Simulator

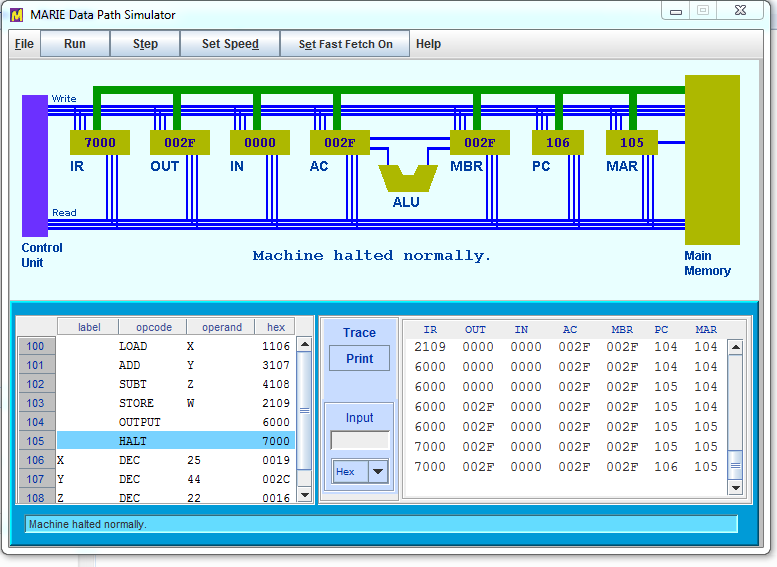


Figure 4: MARIE Data Path Simulator

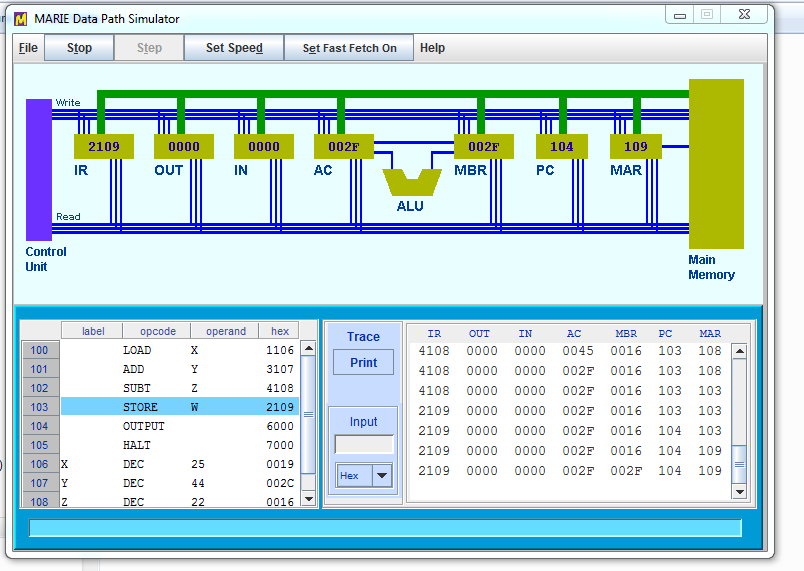
\

Figure 5: MARIE Data Path Simulator