EEE174 –CpE185 Introduction to Microprocessors

LAB 1 part 3 – x86

**Lab Session: Wednesday 6:30PM - 9:10PM**

**Section 32385**

**Lab Instructor: Sean Kennedy**

**Student Name: Andrew Robertson**

Table of Contents

[Part 3 3](#_Toc509225808)

[Overview 3](#_Toc509225809)

[Pre-Lab 4](#_Toc509225810)

[Problem Definition: 4](#_Toc509225811)

[Flow Chart: 4](#_Toc509225812)

[Lab Discussion 5](#_Toc509225813)

[Work Performed / Solution: 5](#_Toc509225814)

[Listing Files(s): 5](#_Toc509225815)

[Conclusion 6](#_Toc509225816)

# Part 3

## Overview

## Pre-Lab

### Problem Definition:

### Flow Charts:

Initial Program:



When I first looked at the program I had no idea what MOV CX, 10 was for. I first left that instruction blank here then continued moving forward until I reached LOOPNE, once I looked that command up everything began to make sense. This program initializes a counter that starts from 10 as well as a pointer to the value of ASCII character 0. The pointer is incremented while the counter is decremented until the counter becomes zero.

Fixed Program:



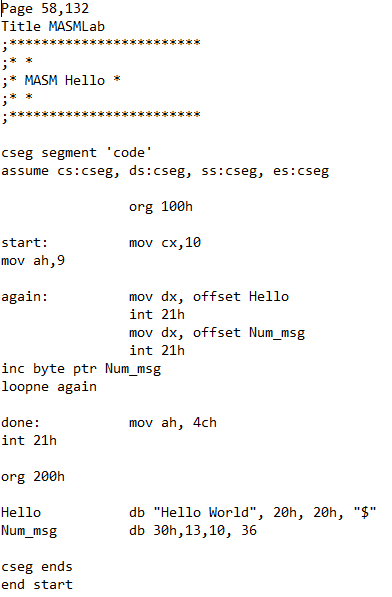
Looking at the previous program, I think the issue between runs is going to be the process of incrementing the byte pointer value without first setting it. After the program completes I believe the byte pointer will be left with whatever value it last had so the program will then display the next 10 ASCII characters. Luckily the loop is triggered off a decrementing counter that is set every time instead of comparing to a certain hex value, as that would mean a complete 128-character loop around the next run.

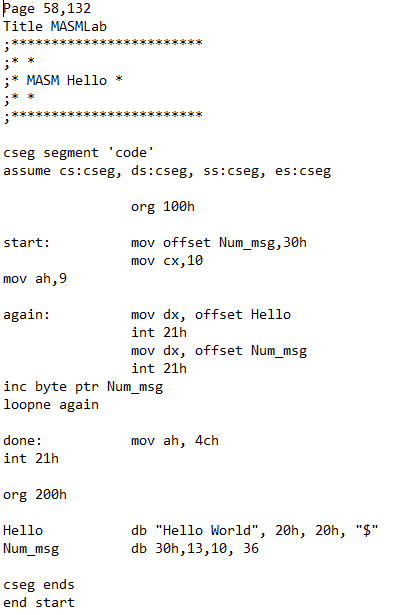
MASM complete program:



I don’t yet know the inner workings of how I should interpret the value the user inputs. What first comes to mind is using a buffered input, 8 bit rotates if there are two characters, and a temporary storage location for ASCII conversion of the individual characters if they don’t come in as ASCII characters already. An additional conditional section might need to be added for the handling of double digit numbers vs single digit numbers. I’m also thinking conversion may be needed after input to be able to have a value to compare to when incrementing and comparing the internal counter.

### .ASM Files:





## Lab Discussion

### Work Performed / Solution:

1)

2)

3)

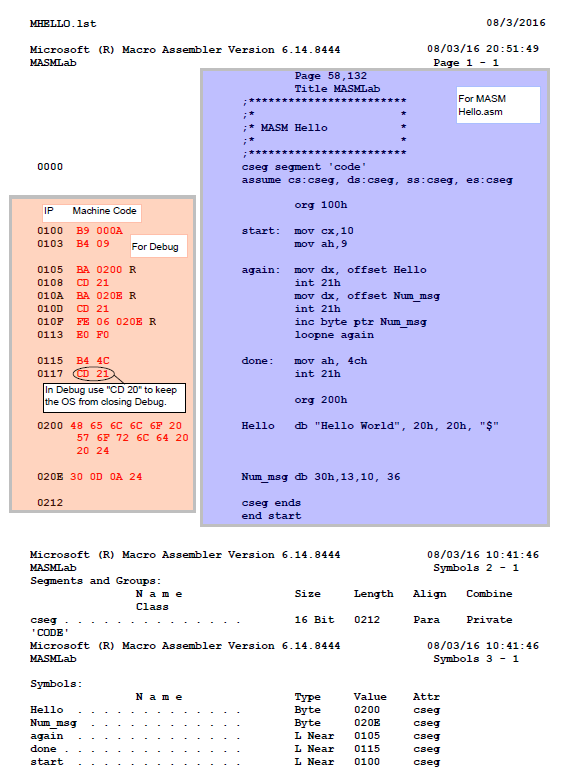
4)

5)

6)

### Listing Files(s):

INT21 descriptions: <http://spike.scu.edu.au/~barry/interrupts.html#ah0a>



# Conclusion