Lab 01: Programming in Assembly Language using MASM

EE-222: Microprocessor System

February 13, 2018

Contents

Abstract

Microprocessor architecture is the actually the interface in between software and hardware. A microprocessor system consists of *Microarchitecture and ISA*, latter begin software while prior the hardware layer.
Since ISA is a part of microarcitectury, it is surely hardware-dependent as well. Each microarchitecture has its own ISA defined with its own pros and cons, as per trade-offs decided by the architect.
Here we explore the all-famous *x86 ISA* which is the intruction set of *80x86 series*, the core of this course.

1 MASM

The Microsoft Macro Assembler (MASM) is an x86 assembler for Microsoft Windows that uses the Intel syntax. Assembly language is a great tool to understand how a computer works and with the help of MASM you will be able to assemble and run your programs written in Assembly language.

2 Assembly Language

2.1 Addition Code

```
TITLE Add two registers (example.asm)
; The comments are given after the semi colon on a line
; This program adds 32-bit unsigned
; integers and stores the sum in the ecx register
Include irvine32.inc
.data
;variable declarations go here
.code
Main Proc
;instructions go here
Mov  eax, 30 ;Assembly Language is NOT case sensitive
Mov  ebx, 20
Add  edx, eax
Add  edx, ebx
Call  dumpregs ;displays the result on the screen by displaying all register values
Exit
Main endp
End main
```

2.2 Execution

```
EAX=0000001E  EBX=00000014  ECX=00000000  EDX=00401037
ESI=00000000  EDI=00000000  EBP=0012FF94  ESP=0012FF8C
EIP=00401023  EFL=00000202  CF=0  SF=0  ZF=0  OF=0
```

3 Debugging

3.1 Analysis

Resister	Content	Decimal
EAX	0000001E	30
EBX	00000014	20
EDX	00401037	4198455

3.2 Verification

- MOV
  - EAX stores 30
  - EBX stored 20
- ADD
  - The add command stores a garbage value in registem EDX

Reason The possible reason for this error is *already existing garbage value* in registex EDX

Solution This poblem can be resolved by *storing zero* in register EDX before addition. Thus garbage value will be cleared.

3.3 Modified Code

```
TITLE Add two registers (example.asm)
; The comments are given after the semi colon on a line
; This program adds 32-bit unsigned
; integers and stores the sum in the ecx register
Include irvine32.inc
.data
;variable declarations go here
.code
Main Proc
;instructions go here
Mov edx, 0 ;erasing the garbage value
Mov  eax, 30
Mov  ebx, 20
Add  edx, eax
Add  edx, ebx
Call  dumpregs
Exit
Main endp
End main
```

3.4 Debugged Execution

```
EAX=0000001E  EBX=00000014  ECX=00000000  EDX=00000032
ESI=00000000  EDI=00000000  EBP=0012FF94  ESP=0012FF8C
EIP=00401028  EFL=00000212  CF=0  SF=0  ZF=0  OF=0
```

3.5 Analysis

Resister	Content	Decimal
EAX	0000001E	30
EBX	00000014	20
EDX	00000032	50

4 Summary

The basic code in x86 version of assembly language has been debugged using hit-and-trial, and deductive method using DOS command-line assembler MASM.