NEW YORK CITY COLLEGE OF TECHNOLOGY

THE CITY UNIVERSITY OF NEW YORK

Department of Computer Engineering Technology

300 Jay Street, Brooklyn, NY 11201-1909

LAB REPORT

CET 3510 – OL71

(MICROCOMPUTER SYSTEMS TECHNOLOGY LABORATORY)

LAB #6

Arithmetic Operations: Simple Calculator

Name: Puja Roy

Date: 10/17/21

Due Date: 10/24/21

Table of Contents

Objective:

The objective of this lab is to be familiar with and explore arithmetic instructions for the 80x86, such as addition, subtraction, multiplication, division, and module. I wrote an in-line assembly language by using general purpose registers. There are eight 32-bit registers, eight 16-bit registers that also have, and eight 8-bit registers. You cannot use arbitrary registers as you can with other operations, such as addition and subtraction.

Materials:

- Microsoft Visual Studio C++ Community Edition 2019

Procedure:

1. First, open Microsoft Visual Studio C++ Community Edition 2019
2. Then, type program#1, compile and run the program.
3. Then modify the code.
4. Lastly, analyze the output.

Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <iostream>
#include <time.h>
using namespace std;

// addition for signed integers
void addition (short int x, short int y);
//subtraction for signed integers
void subtraction (short int x, short int y);
//addition for unsigned integers
void un_addition (unsigned short int x, unsigned short int y);
//subtraction for unsigned integers
void un_subtraction (unsigned short int x, unsigned short int y);

int main()
{
    //Declare variables here
    char ch, ch1, ch2, ch3;
    signed short r1, r2;
    unsigned short ur1, ur2;
    cout<< "Start your calculator Y/N, enter Y(y) or N(n)"<<endl;
        cin>>ch;
    ch1 = ch;
    while (ch1=='Y'||ch1=='y')
    {
        Menu:
        cout << " Menu:\n";
        cout << "1, Signed Integer Arithmetic Operation (16-bit)\n";
        cout<< "2, Unsigned Integer Arithmetic Operation (16-bit)\n";
        cout << "3, Exit\n";
        cout << "Menu Option:\n ";
        std::cin>>ch;
        ch2 = ch;

    Submenu:
        if (ch2=='1')
        {
            cout << "Submenu - input your choice\n";
            cout << "a, Input two 16-bit signed number operands for addition, and display\n"
                << " the sum in decimal and hexadecimal format, respectively.\n";

            cout<<"b, Input two 16-bit signed number operands for addition, and display\n"
                << " the sum in decimal and hexadecimal format, respectively.\n";
            cin>>ch;
            ch3=ch;
            switch(ch3)
            {
               case 'a':
               {
                cout << "Input two signed number operands in decimal format\n";
                cin >> r1 >> r2;
                addition(r1, r2);
                cout << "=========================\n";
                break;
```

```
51                  cout << "Input two signed number operands in decimal format\n";
52                  cin >> r1 >> r2;
53                  addition(r1, r2);
54                  cout << "=========================\n";
55                  break;
56              }
57              case 'b':
58              {
59                  cout << "Input two signed number operands in decimal format\n";
60                  cin >> r1 >> r2;
61                  subtraction(r1, r2);
62                  cout << "=========================\n";
63                  break;
64              }
65              default: goto Menu;
66              }
67          }
68          else if (ch2=='2')
69          {
70              cout << "Submenu-input your choice\n";
71              cout << "a,Input two 16-bit unsigned number operands for addition and display\n"
72                  << " the sum in decimal and hexadecimal format, respectively.\n";
73
74              cout<< "b, Input two 16-bit signed number operands for addition, and display\n"
75                  << " the sum in decimal and hexadecimal format, respectively.\n";
76              cin >> ch;
77              ch3 = ch;
78              switch (ch3)
79              {
80                case 'a':
81                {
82                  cout << "Input two unsigned numnber operands in decimal format\n";
83                  cin >> r1 >> r2;
84                  un_addition(r1, r2);
85                  cout << "=====\n";
86                  break;
87                }
88              case 'b':
89              {
90                  cout << "Input two unsigned numnber operands in decimal format\n";
91                  cin >> r1 >> r2;
92                  un_subtraction(r1, r2);
93                  cout << "=====\n";
94                  break;
95              }
96              default: goto Menu;
97            }
98          }
99          else
```

```
100         {
101             goto EndLable;
102         }
103         cout << "Do you like to continue the arithmetic operation (Y/N)? Enter Y(y) or N(n)" << endl;
104         cin >> ch;
105         ch1 + ch;
106     }
107 EndLable:
108     cout<< "Exit program"<< endl;
109     system("pause");
110
111     exit(0);
112     return 0;
113 }
114
115 // addition for signed short integers
116 void addition(short int x, short int y)
117 {
118     short int r;
119     _asm
120     {
121         MOV AX, x;
122         MOV BX, y;
123         ADD AX, BX;
124         MOV r, AX;
125     }
126
127 cout << "The decimal sum of" << dec << x << "and" << dec << y << " is" << dec << r << endl;
128 cout << "The hexadecimal sum of" << hex << x << "and" << hex << y << " is" << hex << r << endl;
129 }
130 //subtraction for signed short integers
131 void subtraction(short int x, short int y)
132 {
133     short int r;
134     _asm
135     {
136         MOV AX, x;
137         MOV BX, y;
138         Sub AX, BX;
139         MOV r, AX;
140     }
141     cout << "The decimal subtraction of"<< dec << x <<" minus" << dec << y << " is" << dec << r << endl;
142     cout << "The hexadecimal subtraction of" << hex << x << "minus" << hex << y << " is" << hex << r << endl;
143 }
144
145 //addition for unsigned short integers
146 void un_addition(unsigned short int x, unsigned short int y)
147 {
148     unsigned short int r;
```

```
149     _asm
150     {
151         MOV AX, x;
152         MOV BX, y;
153         add AX, BX;
154         MOV r, AX;
155     }
156     cout << "The decimal subtraction of" << dec << x << "minus" << dec << y << " is" << dec << r << endl;
157     cout << "The hexadecimal subtraction of" << hex << x << "minus" << hex << y << " is" << hex << r << endl;
158 }
159  //addition for unsigned short integers
160 void un_subtraction(unsigned short int x, unsigned short int y)
161 {
162     unsigned short int r;
163     _asm
164     {
165         MOV AX, x;
166         MOV BX, y;
167         Sub AX, BX;
168         MOV r, AX;
169     }
170     cout << "The decimal subtraction of" << dec << x << "minus" << dec << y << " is" << dec << r << endl;
171     cout << "The hexadecimal subtraction of" << hex << x << "minus" << hex << y << " is" << hex << r << endl;
172 }
```

Output:

```
Start your calculator Y/N, enter Y(y) or N(n)
y
 Menu:
1, Signed Integer Arithmetic Operation (16-bit)
2, Unsigned Integer Arithmetic Operation (16-bit)
3, Exit
Menu Option:
 1
Submenu - input your choice
a, Input two 16-bit signed number operands for addition, and display
 the sum in decimal and hexadecimal format, respectively.
b, Input two 16-bit signed number operands for addition, and display
 the sum in decimal and hexadecimal format, respectively.
a
Input two signed number operands in decimal format
-2 -3
The decimal sum of -2 and -3 is -5
The hexadecimal sum of fffe and fffd is fffb
------------------------
Do you like to continue the arithmetic operation (Y/N)? Enter Y(y) or N(n)
y
 Menu:
1, Signed Integer Arithmetic Operation (16-bit)
2, Unsigned Integer Arithmetic Operation (16-bit)
3, Exit
Menu Option:
 1
Submenu - input your choice
a, Input two 16-bit signed number operands for addition, and display
 the sum in decimal and hexadecimal format, respectively.
b, Input two 16-bit signed number operands for addition, and display
 the sum in decimal and hexadecimal format, respectively.
b
Input two signed number operands in decimal format
-2 -3
The decimal subtraction of -2 minus -3 is 1
The hexadecimal subtraction of fffe minus fffd is 1
------------------------
Do you like to continue the arithmetic operation (Y/N)? Enter Y(y) or N(n)
y
 Menu:
1, Signed Integer Arithmetic Operation (16-bit)
2, Unsigned Integer Arithmetic Operation (16-bit)
3, Exit
Menu Option:
 2
Submenu-input your choice
a,Input two 16-bit unsigned number operands for addition and display
 the sum in decimal and hexadecimal format, respectively.
b, Input two 16-bit signed number operands for addition, and display
 the sum in decimal and hexadecimal format, respectively.
a
Input two unsigned numnber operands in decimal format
2 3
The decimal subtraction of 2 and 3 is 5
The hexadecimal subtraction of 2 and 3 is 5
-----
Do you like to continue the arithmetic operation (Y/N)? Enter Y(y) or N(n)
y
 Menu:
1, Signed Integer Arithmetic Operation (16-bit)
2, Unsigned Integer Arithmetic Operation (16-bit)
3, Exit
Menu Option:
 2
Submenu-input your choice
a,Input two 16-bit unsigned number operands for addition and display
 the sum in decimal and hexadecimal format, respectively.
b, Input two 16-bit signed number operands for addition, and display
 the sum in decimal and hexadecimal format, respectively.
b
Input two unsigned numnber operands in decimal format
2 3
The decimal subtraction of 2 minus 3 is 65535
The hexadecimal subtraction of 2 minus 3 is ffff
-----
Do you like to continue the arithmetic operation (Y/N)? Enter Y(y) or N(n)
y
 Menu:
1, Signed Integer Arithmetic Operation (16-bit)
2, Unsigned Integer Arithmetic Operation (16-bit)
3, Exit
```

Conclusion:

From this experiment, I learned and explored arithmetic instructions for the 80x86, such as addition, subtraction, multiplication, division, and module. I learned how to write an in-line assembly language by using general purpose registers. There are eight 32-bit registers, eight 16-bit registers that also have, and eight 8-bit registers. You cannot use arbitrary registers as you can with other operations, such as addition and subtraction.