Introduction to Computer Science
Jacobs University Bremen
Dr. Jürgen Schönwälder

Module: CH-232
Date: 2022-11-11
Due: 2022-11-18

ICS 2022 Problem Sheet #10

Problem 10.1: *assembler programming* (1+2+3+1 = 7 points)

The following program has been written for the simple central processing unit introduced in class. The table below shows the initial content of the 16 memory cells. The first column denotes the memory address and the second column shows the memory content in hexadecimal notation.

Cell	Hex	Binary	Assembler	Description
0	2f			
1	6a			
2	4f			
3	21			
4	71			
5	41			
6	a9			
7	d0			
8	e0			
9	6f			
10	01			
11	02			
12	03			
13	04			
14	05			
15	06			

a) Convert the machine code from hexadecimal notation into binary notation.

b) Write down the assembler code for the machine code. Add meaningful descriptions.

c) The program leaves a result in memory cell 15 when it halts. What is the value? Explain how the program works.

d) What happens if the value stored in memory cell 9 is changed to 0x70 before execution starts? Explain.

Problem 10.2: *integer multiplication in risc-v rv32i assembler* (2+1 = 3 points)

The 32-bit RISC-V base integer instruction set (rv32i) does not support multiplication and division operations. To deal with this, a compiler may call a function when a multiplication is needed. For example, `gcc` expects that a function `__mulsi3(unsigned int a, unsigned int b)` is provided to multiply two integers. A multiplication can be carried out by repeated additions and shifts:

```
unsigned int __mulsi3 (unsigned int a, unsigned int b)
{
    unsigned int r = 0;

    while (a) {
        if (a & 1) {
            r += b;
        }
        a >>= 1;
        b <<= 1;
    }
    return r;
}
```

a) Translate the above C code into equivalent RISC-V rv32i assembler code. Comment the assembler code to explain how the calculation proceeds. Note that the arguments are passed via the registers a0 (x10) and a1 (x11) and that the result is returned in a0 (x10).

b) Does the function need a function call prologue and a function epilogue? Explain why or why not.

You are invited to use emulsiV to develop and test your assembler code.