# Homework 4

Andrei Tumbar

02-24-2021

## 6.10

Convert the following instructions into machine language.

```
add $t0, $s0, $s1
lw $t0, 0x20($t7)
addi $s0, $0, -10

000000|10000|10001|01000|00000|100000
100011|01111|01000|0000000000100000
001000|00000|10000|1111111111110110
```

## 6.12

### Which instructions from Exercise 6.10 are I-type instructions?

The `lw` and `addi` instructions are I-type instructions.

### Sign-extend the 16-bit immediate of each instruction from part (a) so that it becomes a 32-bit number.

```
(16-bit)         (32-bit)
0x0020       -> 0x00000020
-10 (0xff06) -> 0xffffff06
```

## 6.14

```
(hex) 0x20080000
(bin) 001000|00000|01000|0000000000000000
(MIPS) addi $t0, $0, 0x0000

(hex) 0x20090001
(bin) 001000|00000|01001|0000000000000001
(MIPS) addi $t1, $0, 0x0001

(hex) 0x0089502A
(bin) 000000|00100|01001|01010|00000|101010
(MIPS) slt $t2, $a0, $t1

(hex) 0x15400003
(bin) 000101|01010|00000|0000000000000011
(MIPS) bne $t2, $0, 0x0003

(hex) 0x01094020
(bin) 000000|01000|01001|01000|00000|100000
(MIPS) add $t0, $t0, $t1
```

```
(hex) 0x21290002
(bin) 001000|01001|01001|0000000000000010
(MIPS) addi $t1, $t1, 0x0002

(hex) 0x08100002
(bin) 000010|00,0001,0000,0000,0000,0000,0010
(addr) 0000|0000|0100|0000|0000|0000|0000|1000
(MIPS) j 0x0100002 (0x00400008)

(hex) 0x01001020
(bin) 000000|01000|00000|00010|00000|100000
(MIPS) add $v0, $t0, $0

(hex) 0x03E00008
(bin) 000000|11111|00000|00000|00000|001000
(MIPS) jr $ra
```

```
int f(int n)
{
    int t0 = 0;

    for (int t1 = 1; t1 < n; t1 += 2)
    {
        t0 += t1;
    }

    return t0;
}
```

This program will sum up all of the odd numbers that are less than the input `n` and return this value.

### Question 6.1

Write assembly code to swap two registers.

```
; Swap registers $t0 and $t1
xor $t0, $t0, $t1
xor $t1, $t1, $t0
xor $t0, $t0, $t1
```

### Final Question

Show all relevant data and control values on Harris figure 7.11 for commands

```
add  $t0, $s0, $s1
addi $s0, $0, -10
```

Table 1: Relevant data values

| Instruction | A1 | A2 | A3 | SignImm |
|---|---|---|---|---|
| add $t0, $s0, $s1 | 10000 | 10001 | 01000 | N/A |
| addi $s0, $0, -10 | 00000 | N/A | 01000 | 0xfffffff6 |

Table 2: Relevant control values

| Instruction | MemToReg | MemWrite | Branch | ALUSrc | RegDst | RegWrite |
|---|---|---|---|---|---|---|
| add $t0, $s0, $s1 | 0 | 0 | 0 | 0 | 0 | 1 |
| addi $s0, $0, -10 | 0 | 0 | 0 | 1 | 0 | 1 |