02321 Hardware/Software Programmering (E10)

Home assignment 3

Morten Hillebo (s072923)
Kim Rostgaard Christensen (s084283)

Problem 5.4

(a)

8 bits should be sufficient ($2^8 = 256$)

(b)

5 bits are needed.

(c)

The PC-relative address must be 14, as the relative calculation is done after PC increment.

Problem 5.8

In order to increment the number of registers, we would need to increment memory word size (instruction size). This is due to the fact that we now need 5 bits to identify the registers in the add instruction instead of 3.

Problem 5.10

0	0	0	0	1	1	1	1	0	1	0	1	0	1	0	1

This instruction is the BR (Conditional Branch) instruction. It branches to the PC offset specified

0	1	0	0	1	1	1	1	0	1	0	1	0	1	0	1

This is the JSR (Jump to Subroutine) instruction. It saves the PC to R7 and jumps to the offset or location specified, depending on bit[11].

BR does not provide a way back, and has an option to not branch, so to speak. The JSR is unconditional and always jumps, but in contrast to BR provides a way back. BR can be used for programming language's conditional expressions (such as if's).
JSR can be used to provide functions that return (such as functions in C).

Problem 5.22

The first instruction is LEA. It loads address $x3F + x3011 = x3050$ (The incremented PC value)+to R3
The second instruction is LDR. It loads the value stored (x70A4) at the address specified in base register R3 (x3050) to register R4 using an offset of x0.
The last instruction is also LDR. It now loads the value stored at x70A4 to R6, so that it now contains x123B.

As we load values to three register, and there are no such instruction in our set, we assume that we need an instruction that loads the value directly to R6.
For this, we can use the LDI instruction that loads the address of a memory, from memory. So the instruction would be: LDI R6, x3050.

Problem 5.24

Since we have ranges from 0..63 with zero extension, the largest address the instruction can load from is x4050 and the least x4011.

Problem 5.36