Ve370 Big RC

Unrecorded part

Chapter 1

Suggestions: keep the concepts in mind. Practice about performance related calculations.

Chapter 2

Chapter 3

Procedure of function calling: remember how to pass parameters, remember how to and whether to store values of registers into the stack, remember how to pass parameters back, how to release stack, and how to jump back to the caller.

Stack goes down, so open stack == add negative number, release stack == add positive number

Beware the difference between 32bit (P1) and 64 bit(lecture, exam)

Chapter 4

Notice the difference between types of instructions.

Opcode and funct3 serves to tell the computer which instruction is which.

Beware the boundaries of instructions since bits are limited

Go over the slides for the "Assembler ..." part

Chapter 5.6.7

Go over the slide or Ve270 if needed.

make sure you understand each process.

Do some exercises on arithmetic of floating point numbers, mul and div if necessary

Know how to use the extensions like mulh to detect overflow and other situations

For float point numbers, range and precision is important.

Chapter 8

Single stage is a basic implementation of realizing the basic instruction set.

Take the example of:

add x10, x11, x12

The first step: know this is a "add" function, the source is x11 and x12, and the destination is x10

(IF stage in pipeline)

The second stage: Fetch the values in x11 and x12

The third stage: x11 + x12

The fourth stage: x10 = result

Notice that this is a instruction not relevant to Mem. Since there are memory related instructions, a basic procedure of implementation will be:

Fetch instruction-> Access RF -> ALU -> MEM -> RF again

So the basic single stage processor looks like the one on the blackboard

Controls are added so that the computer knows which instruction is which, and how to operate.