Lab 4 Report:
Adrian Sam Daliri
219677186
10/23/2023

• **Intro:** This lab is about RISC-V instructions for multiplication, division and floating point calculations

• **Explanation:**

0.1 d1a.asm

This code introduces the mul instruction which allows the multiplication of the numbers in two registers and storing the results in a register. In this case 11 and 12 are multiplied and stored in x7

0.2 d1c.asm

This code receives the dimensions of a rectangle from the user and uses mul to calculate the area of the rectangle and outputs the result to the out window.

0.3 d1d.asm

Introduces the div instruction which allows the division of the numbers in two registers and storing the results in a register. In this case 1000 is divided by 20 and the result is stored in x7.

0.4 d1e.asm

Introduces the rem instruction which provides the remainder of division being performed. In this case 67 is being divided by 16 and the remainder is stored in x7

0.5 d1f.asm

This code prompts the user to provide two positive integers and then uses Euclid's algorithm in order to find the greatest common divisor of the two numbers and output it to the out window. It uses rem to find the remainder and then stores the divisor in x6 and checks if the remainder was zero, if not it loops with the remainder as the new divisor.

0.6 d2a.asm

This code introduces the instruction fld(FP load double) which allows FP numbers in the memory to be loaded to the FP registers. Here the float at d1 in the memory is loaded into register f1 and the float at d2 in the memory is loaded into register f2.

0.7 d2b.asm

This code illustrates the various floating point arithmetic such as fadd.d(for addition), fsub.d(for subtraction), fmul.d(for multiplication), fdiv.d(for division)

0.8 d2c.asm

This code loads 3 floats from the memory into f1, f2 and f3 and then computes the sum of the 3 numbers in two ways. In f4, (v1 + v2) + v3 = 3 is performed and in f5, v1 + (v2 + v3) = 0.

0.9 d2d2.asm

This code is the same as the first code but the precision is changed and since 15 decimal digits is the limit for rounding so in this case rounding doesn't occur.

0.10 d2e.asm

This code introduces fsd(FP store double) instruction which allows float values to be stored in the memory from a register. This example finds the result of (3.0 + 3.0) then divides it by the result of (3.0 * 3.0)

0.11 d2f.asm

This code introduces fmin.d which finds the smaller of two floats and fmax.d which finds the larger of the two floats and each instruction stores the result in a given register. In this example, it compares 1.0 and 2.0.

0.12 d2g.asm

This code introduces the fsqrt.d instructions which find the square root of a float in a given register and then it stores the result in a given register.

0.13 d2h.asm

This code introduces fsgnj.d which sets the sign of the destination register to the sign of the value in register 2(fs2), fsgnjn.d which sets the sign of the destination register to the opposite of the sign of the value in register 2(fs2) and fsgnjx.d which sets the sign of fd to the XOR of the sign bits of fs1 and fs2. In this example -2.0 and 3.0 are used.

0.14 d2i.asm

This code introduces fmv.x.d(FP move double to long) and fmv.d.x(FP move long to double) which allows moving of numbers to and from float registers to integer registers without conversion.

0.15 d2j.asm

Introduces fcvt.l.d(FP convert double to long) and fcvt.d.l (FP convert long to double) instructions which are similar to the previous code but instead of an approximation we receive the converted value after a conversion between the types take place.

0.16 d2k.asm
This code introduces flt.d(FP less than) and fle.d(FP less than or equal) and uses them in a code which assigns x the value b if x<a and assigns x the value a is b<=x. In this code flt.d and fle.d are used to find x<a and b<=x and then comparing the value(which is either 0 or 1) to 0 to decide to branch or not to the other case.

• Conclusion
This code was very tedious and I had to go through all the functions with a new data type but by referencing the lecture notes and the lab manual it was a very good learning experience.