COAL Lab-09

Arithmetic and Branch Instruction Assembly Code

Name: Saad Nisar Butt

Reg. no: cs211246

Class: BSCS-3C-1

Literature Review:

Most integer computational instructions operate on XLEN bits of values held in the integer register file. Integer computational instructions are either encoded as register-immediate operations using the I-type format or as register-register operations using the R-type format. The destination is register rd for both register-immediate and register-register instructions. No integer computational instructions cause arithmetic exceptions.

The instruction format for register-immediate instruction is shown below:

31 20	19 15	14 12	11 7	6 0
imm[11:0]	rs1	funct3	rd	opcode
12	5	3	5	7
I-immediate[11:0]	src	ADDI/SLTI[U]	dest	OP-IMM
I-immediate[11:0]	src	ANDI/ORI/XORI	dest	OP-IMM

31 25	24 20	19 15	14 12	11 7	6 0
imm[11:5]	imm[4:0]	rs1	funct3	rd	opcode
7	5	5	3	5	7
0000000	shamt[4:0]	src	SLLI	dest	OP-IMM
0000000	shamt[4:0]	src	SRLI	dest	OP-IMM
0100000	shamt[4:0]	src	SRAI	dest	OP-IMM

The instruction format for register-register instruction is shown below:

31 25	24 20	19 15	14 12	11 7	6 0
funct7	rs2	rs1	funct3	rd	opcode
7	5	5	3	5	7
0000000	src2	src1	ADD/SLT/SLTU	dest	OP
0000000	src2	src1	AND/OR/XOR	dest	OP
0000000	src2	src1	SLL/SRL	dest	OP
0100000	src2	src1	SUB/SRA	dest	OP

The instruction format for branch instruction is shown below:

31	30 25	24 20	19 15	14 12	11 8	7	6 0
imm[12]	imm[10:5]	rs2	rs1	funct3	imm[4:1]	imm[11]	opcode
1	6	5	5	3	4	1	7
offset[12,10:5]		src2	src1	BEQ/BNE	offset[11,4:1]		BRANCH
offset[12,10:5]		src2	src1	BLT[U]	offset[11,4:1]		BRANCH
offset[12,10:5]		src2	src1	BGE[U]	offset[11,4:1]		BRANCH

In Lab Tasks

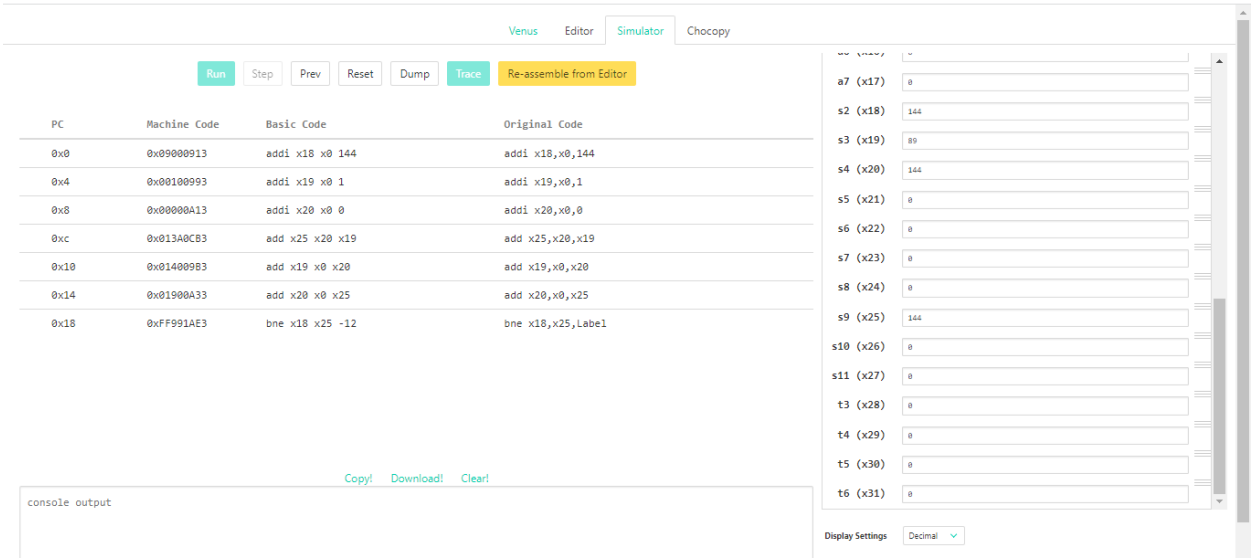
Task:

1. Write down a simple assembly program to calculate the Fibonacci series till 144. Stimulate the code on Venus.

Code:

```
addi x18,x0,144

addi x20,x0,0

addi x19,x0,1


Label:

add x25,x20,x19

add x19,x0,x20

add x20,x0,x25

bne x18,x25,Label
```

Output:



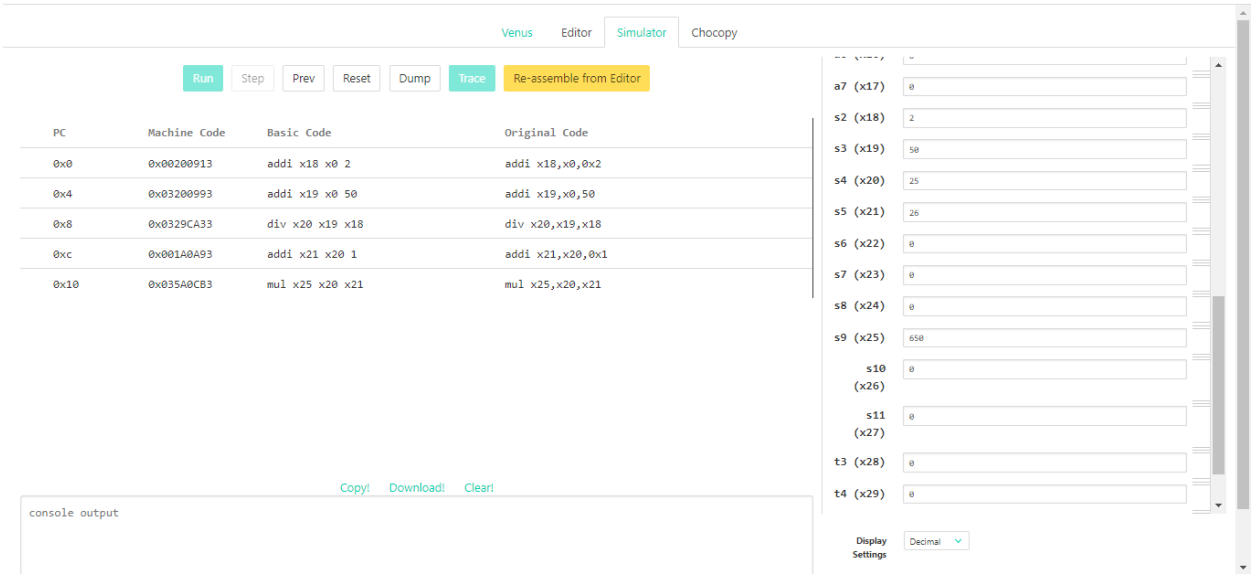
Task:

2. Write down a simple assembly program to calculate the sum of even numbers between the range of 1-50. Stimulate the code on Venus.

Code:

```
addi x18,x0,0x2

addi x19,x0,50

div x20,x19,x18

addi x21,x20,0x1

mul x25,x20,x21
```

Output:


Venus
Editor
Simulator
Chocopy
Run
Step
Prev
Reset
Dump
Trace
Re-assemble from Editor
PC
Machine Code
Basic Code
Original Code
0x0
0x00200913
addi x18 x0 2
addi x18,x0,0x2
0x4
0x03200993
addi x19 x0 50
addi x19,x0,50
0x8
0x0329CA33
div x20 x19 x18
div x20,x19,x18
0xc
0x001A0A93
addi x21 x20 1
addi x21,x20,0x1
0x10
0x035A0CB3
mul x25 x20 x21
mul x25,x20,x21
Copy!
Download!
Clear!
console output
a7 (x17)
s2 (x18)
s3 (x19)
s4 (x20)
s5 (x21)
s6 (x22)
s7 (x23)
s8 (x24)
s9 (x25)
s10 (x26)
s11 (x27)
t3 (x28)
t4 (x29)
Display Settings
Decimal