CSE_BUBBLE

Assignment 7

Manasvi Jain - 210581
Saugat Kannojia - 210943

[PDS1] Decide the registers and their usage protocol.

Sol. As per the problem statement, our processor only has 32 registers, all being 32 bits wide, to fetch, store and decode all the instructions. We will use some registers for data storage and some for instructions storage.
Our registers are as follows-
R0: Zero register
R1-R5: Used to store parameters - from a0 to a4
R6-R7: Used to store results - v0,v1
R8-15: Temporary Registers - from t0 to t7
R16-23: Saved results - from s0 to s7
R24-31: Stores contents stored for later use

[PDS2] Decide upon the size of instruction and data memory in VEDA.

Sol. We are making the VEDA memory of size 32 with every register of width 32 bits which will be storing all the data. We will be making instruction memory of size 32, storing the machine code generated by converting the MIPS code.

[PDS3] Design the instruction layout for R-, I- and J-type instructions and their respective encoding methodologies.

Sol. Our VEDA memory is composed of 32 registers which can be accessed using 5-bit addresses.

1. R-Type Instructions-
   In this type of instruction, we have a specific opcode and function combination for the specific instruction we want to do. We access 3 registers using their addresses to store data.

6 bits opcode	5 bits rs	5 bits rt	5 bits rd	5 bits shamt	6 bits funct

These are the following opcodes and their function codes:-

Instruction	Opcode	Function Code
add	000000	100000
addu	000000	100001
sub	000000	100010
subu	000000	100011
and	000000	100100
or	000000	100101
sll	000000	000000
srl	000000	000010
slt	000000	101010

2. I-Type Instructions-
   In this type of instruction, we have a specific opcode for the specific instruction we want to do. We access 2 registers using their addresses to store data.

6 bits opcode	5 bits rs	5 bits rd	16 bits constant

These are the following opcodes:-

Instruction	Opcode
addi	001000
addiu	001001

andi	001100
ori	001101
lw	100011
sw	101011
beq	000100
bne	000101
bgt	000111
bgte	000001
ble	000011
bleq	000110
slti	001010

3. J-Type Instructions-
   In this type of instruction, we have a specific opcode for the specific instruction we want to do.
   We access only 1 register using their address to store data for jr instruction.

opcode 000010	5 bits register	0000000000	funct 001000

j and jal only requires the target label.

6 bits opcode	26 bits target

j has opcode 010100, and jal has opcode 010101.

How to run the processor-

1. Go to instruction memory and provide your machine code into the memory by typing the instructions manually.
2. Go to the testbench and run it for a sufficient amount of time and change the output in whatever format you want by extracting elements of the data memory.
3. Display the data memory or view the gtkwave for the output.