

Coding Midterm - Practice Questions

It is recommended to build solutions of these on top of Lab1 solution posted in Classes

1. Add Jump Instruction

Jump: j J-Type Instruction PC = JumpAddress Opcode = 2 Hex

Jump instruction is like **beq** instruction except that in jump, branch is always taken. Here take care of how the jump address is calculated.

2. Write Code to calculate IPC (Instruction per cycle) and Data Memory Utilization. Print both values in terminal using cout.

IPC = Number of Valid Instructions / Total Number of cycles

Data Memory Utilization = Total DMEM (read/write) access / Number of valid instructions

Number of valid instructions are total number of actual instructions executed, considering phenomenons like stall and loop inside program.

Data Memory Utilization is number of times data memory is accessed (either a read (lw) or write (sw)) divided by total number of valid instructions. For eg. If there are 2 load word instruction and 1 store word instruction and the total number of valid instructions are 10, then Data Memory Utilization is (2+1)/10 = 0.3

Note: Halt is not considered a valid instruction for this exercise.

3. Modify the code such that if a subu-subu hazard occurs, instead of performing EX-EX forwarding make it stall until the Register File is updated.

Here instead of passing the dependency value to the subsequent instruction from the EX stage, let the instruction get executed first and then use that value directly from the register file for the subsequent instruction.

subu \$1 \$2 \$3	IF	ID	EX	MEM	WB			
Subu \$3 \$4 \$5		IF	ID	ID	ID	EX	MEM	WB