Note 13.5: RISC-V Datapath Extra Points

Kevin Moy

1 CPU Clocking

Now that we have finished our CPU design, we need to consider CPU clocking. Remember we can adjust the CPU clock period to give more time for each instruction- but we want to minimize this time as well.

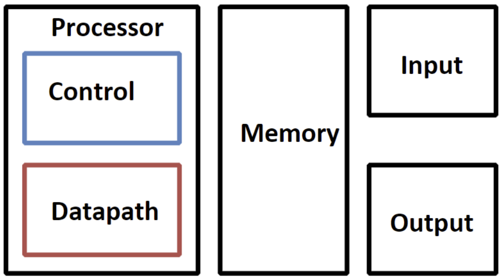
There are generally five stages of an instruction:

1. Instruction Fetch (IF)
2. Decode/Register Read (ID)
3. Execute Instruction (EX)
4. Read/Write from (data) Memory (MEM)
5. Register Write (WB)

Ideally, we want one instruction executed per clock cycle. For an instruction to complete, it must go through all 5 of these stages, so the clock period must be long enough (and thus clock frequency must be low enough) such that an instruction has enough time.

2 Processor Design Principles

Below is a general layout for our processor design:



We've essentially finished the datapath and control part (well, datapath at least).

2.1 Five Stages

There are generally five stages in designing a processor:

1. Analyze instruction set -¿ establish datapath requirements
2. Select datapath components and establish clock methodolgy (max clock period, pipelining, etc.)
3. Build data path.
4. Analyze implementations of each instruction.
5. Assemble control logic

Project 3B focuses primarily on the control logic- we've covered essentially the entire datapath already.

Again, remember that control signals are the ”brain's decisions” for the CPU. It'll tell the ALU to shift logical left, or it'll tell data memory to write to a return register. Control signals are pretty much always paired with MUXs because they are like choice bits. Your control signals, of course, are entirely dependent on your datapath itself, which will change based on your ISA (RISC-V datapaths will differ heavily from MIPS datapaths, for example).

3 Summary

We covered a LOT of material in Note 13. First, we built our *universal* data path (account for all instructions) by first implementing R-type instructions and making cumulative modifications for the other instruction formats along the way. Remember we want all instructions to execute in a *single* clock cycle- i.e. single-cycle CPU.

There are five phases of executing an instruction: IF, ID, EX, MEM, and WB- make sure you know what these stages mean! Note not 5 all stages will necessarily be active for every single instruction (only load instructions do, actually).

The controller, or control logic block, serves as the "brain" of the CPU. The datapath is really just hardware implemented to follow control logic instructions. Use the architect-construction worker analogy! With this abstract, it is actually very easily implement new instructions: just modify our control logic in a few ways to account for new instructions- we don't really need to change our datapath here!