4/3/2022

Go through some code, decide procedure calling conventions.

TODO:

- An unambiguous English description of the syntax and semantics of each instruction (see pages A-51 through A-80 of your book).
- A symbolic description of the behavior of each instruction (like those on the MIPS Green Sheet).

4/6/2022

Adjust the instructions to fit into 16 bits.

Set all instructions to be 0x00XX and data to be 0xFFXX

Replaced $ra with a specific address 0xFFFE

$sp set at 0x1FFF

4/7/2022

Meeting with the professor

Write minimum code for the procedure call

Add the caller of relPrime

Write an assembler

4/8/2022

Finished writing the RTL

Thinking about changing jump or changing sw lw to make them into 4 cycles

4/16/2022

Tried writing the assembler using java

Get file not found error

Plan to try using Python

4/19/2022

Plan not to change the addressing mode, since sign extent is powerful in a way that it does not need the ALU, may add an addressing mode that uses ALU when we switch to pipelining

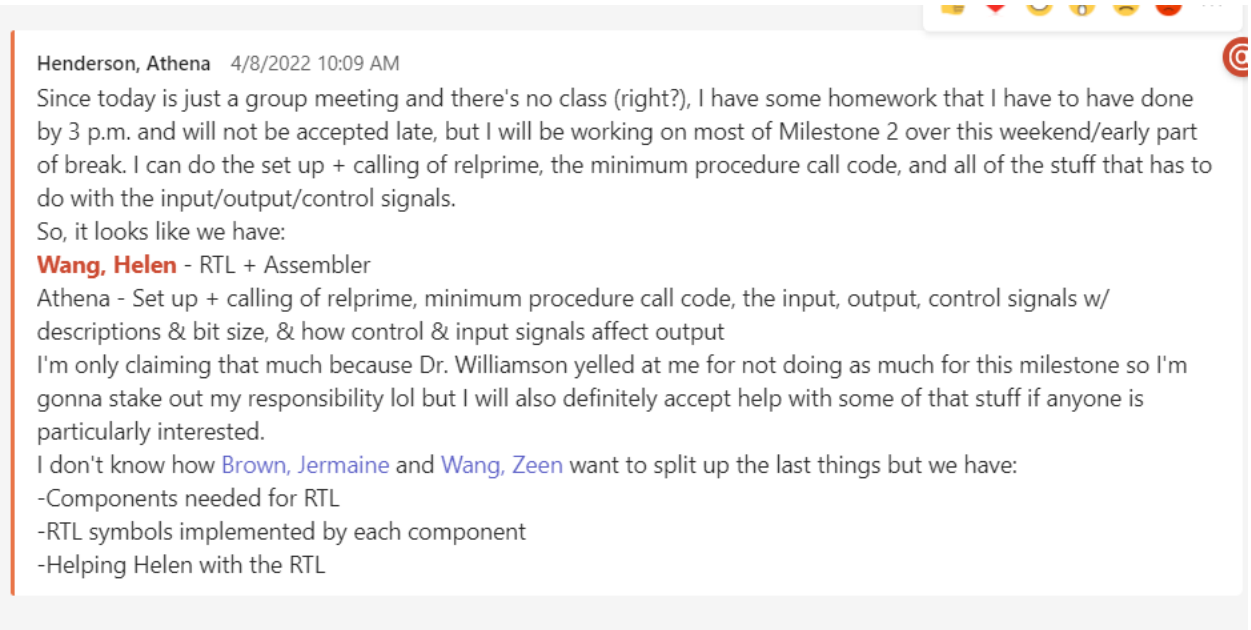
Go through the document of M2 and tell Zeen what to do.

Thinking about having a weekly team meeting time

4/20/2022

Zeen is available Sunday so thinking about 2 pm Sunday.

Current Roles:



Henderson, Athena 4/8/2022 10:09 AM

Since today is just a group meeting and there's no class (right?), I have some homework that I have to have done by 3 p.m. and will not be accepted late, but I will be working on most of Milestone 2 over this weekend/early part of break. I can do the set up + calling of relprime, the minimum procedure call code, and all of the stuff that has to do with the input/output/control signals.
So, it looks like we have:
Wang, Helen - RTL + Assembler
Athena - Set up + calling of relprime, minimum procedure call code, the input, output, control signals w/ descriptions & bit size, & how control & input signals affect output
I'm only claiming that much because Dr. Williamson yelled at me for not doing as much for this milestone so I'm gonna stake out my responsibility lol but I will also definitely accept help with some of that stuff if anyone is particularly interested.
I don't know how Brown, Jermaine and Wang, Zeen want to split up the last things but we have:
-Components needed for RTL
-RTL symbols implemented by each component
-Helping Helen with the RTL

For Zeen:

- A list of the RTL symbols that this component will implement. For example, a generic register component might implement both the 'PC' and 'ALUout' RTL symbols.

- Now that you have RTL, you can use it to design the datapath in the next milestone. However, if your RTL has errors, it will cause your datapath to be incorrect. Go back and verify that your RTL correct.

  - Give a brief overview of your process for double-checking the RTL for errors.

Zeen takes over Assembler

Thinking about adding pseudo instructions and improving RTL. Think better leave it like that since there is a possibility that we will switch to pipelining. Not may possible pseudo instruction, since we will need to have an extra register or store the value of acc on stack then retrieve it, maybe better without it.

Pseudo Instructions:
sw 0(m): load m, sw 0
Store m on stack with offset 0

slt a,b : load a, slt b
Compare a and b, if a<b, put 1 into acc, else put 0

addi m 1: load m, addi 1, save m
Increment m by imm

We do not have a way to address registers other than acc for most instructions. Can only do the 2nd method.

4/22/2022

Half finished with building datapath.

Added a cycle for sw and lw and do the pre-calculation for beq, bne instead since they are used more often

Tried to use the unused bit to switch between the two but beq, bne are AI type, they do not have the 3 extra bits.

If we are changing the source for ALU, we to choose between the op of the mux to be from IR or control. That means adding a mux to the opcode and using another bit from IR to be the op of this new op mux. But we cannot switch back.

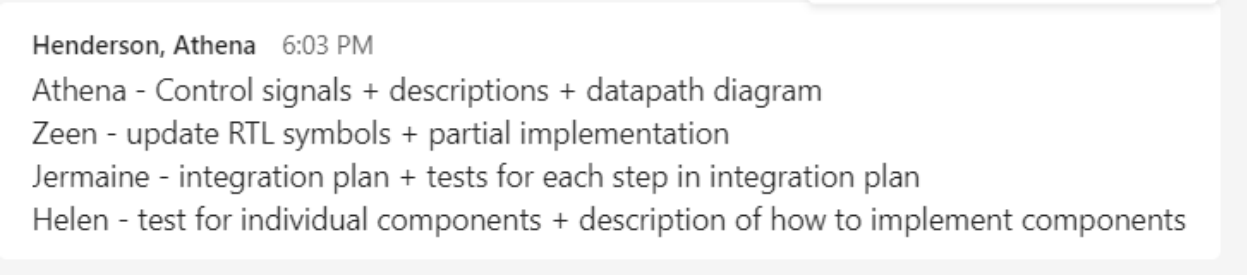
TODO: Add the memory map, increment jump to use 11 bits

TODO: Finish datapath on Sunday

4/24/2022

Finished designing the datapath with Jermaine

Made minor changes to the RTL



Errr.. just noticed that my role should be the test implementation and description, maybe that causes the confusion?

Memory map, update j, jal green sheet and RTL

4/26/2022

Finished updating changes to j jal. Finished adding memory map.

Write some plan of testing the implementation, but the implementation plan itself is not yet written so may need to make changes to it.

The test I write basically focused on testing all the possible inputs of the control signals. Tests for flags and clock are also added for those that have these.

Decide to set memory to read and write on rising edge since in jal we want it to store the current PC before PC changes on falling edge. Does not have a cycle where memory do both read and write so it is okay to put read and write on the same edge.

Main should start at 0x0000

ALU op should be 3 since it does add, sub, and, or, slt

Finished the Complex FSM for M4

TODO: For M4, finish implementation of test bench, need to contact Zeen to see if he can meet during Thursday Friday class time.

TODO: Ask Zeen to use the assembler to get the new machine code that has address of main starting at 0x0000

4/29/2022

Plan to set up detail task during work time in class

No one came at 10:00

4/30/2022

Meet at 1 pm. Split the work.

Since all the parts are dependent on each other, maybe better to do the work together in a meeting. But others seem to prefer to split the works into relatively independent part and do it on their own.

DONE: write tests for reg

DONE: integration part PC

TODO: test PC

TODO: test 1 bit mux?

DONE: try to figure out problem with the waveform generator

TODO: push our files at 5 pm Tuesday

5/2/2022

Fixed issues with the waveform

Finished tb for reg, reg has no problem

Implemented PC using the block diagram file, but have problem testing

5/3/2022

Decide to re-implement PC using Verilog. Found that actually need 2 bit mux for that. Waiting for Jermain to finish the 2 bit mux.

Plan to write test for 1 bit mux.

TODO: push our files at 5 pm

Cannot solve the issue, seems to be a problem with PC, tb of reg run normally.

Comment out everything apart from the setup in tb, still crash. Suspect there is a loop in the connection of PC, don't see any error.

Maybe try letting someone else run the test bench

No time to do the test for 1 bit mux

5/6/2022

Added test for mux1b1

Error testing PC, find that it is because I assume that 00 is input A but Jermain implemented it as D instead. Texted Jermain about whether he want to change his design to match the patter of other mux.

We really should come up with a general design before going and implementing it separately.

Updated the document of PC and mux1b1

TODO: Wait for Jermain's reply, edit PC if needed

TODO: Wait for other's reply and schedule a meeting during the weekend.

5/8/2022

Secluded meeting at 1 pm. Athena's mom came so she leaved a message saying that she will be late. Jermain not replied yet at 1:28. Zeen said that he will meet us via teams. Called Zeen via Teams and Wechat at 1:29, no reply.

Zeen called back at around 1:30, Jermain called at 1:33, he decided to meet on teams today, so I guess I'm the only one who come in person.

We will meet on Monday class time to make sure M4 is working.

We have 2 main tasks for M5, implementing and testing control, implementing and testing the whole datapath. 2 people will do the control, 2 people will do the datapath.

Decide the split on Monday.

We've talked trough our next step, text Athena that she doesn't need to come anymore.

I'm not really available next week due to take home exam, I'm really really worried about how the team will go.

5/9/2022

Split work, Jermaine and I will be doing the implementation and testing of datapath.

Zeen and Athena will be implementing and testing control

Changed the description that our save and load will do I/O when rt is 0x1000

Texted Zeen to add this feature to memory

5/10/2022

TODO: Solve pc test issue

Looked through our implementation files for tb and implementation:

- Memory test still writing by Zeen

- 16bit reg finished both
- Alu finished both
- Alu subsystem implemented no test banch
- PC finished both
- Mux1b1 finished both
- Mux1b16 need testing
- Mux 2b16 finished both
- Mux 3b16 no test file
- Se finished both
- Shift left finished both
- Ze finished both
- Wires subsystem missing both

5/11/2022

Wires subsystem no test file

Mux 3b16 no test file

Solve the problem of 10 bit memory

Control has .v file, no test bench

Write test implementation plan for some instructions, waiting for all subsystems and control to be implemented and tested.

Commit and push journal and document.

5/13/2022

Finished writing the test implementation plan for datapath.

Schedule meeting at Sunday 1 pm, waiting reply from Jermain and Zeen

Asked Jermaine to connect the subsystems

Write tb of mux1b16, but cannot run it due to syntax error in other files.

TODO: figure the error out with the control team and then run the test bentch

5/16/2022

Added a IRWrite bit that is set to 1 only in the fetch stage

Need Athena to update bubble diagram

Need Athena to update datapath:

Add IRWrite above IR

Delete the MemRead