Resources:

Github - ERSP Organization - PyRTL Research Mips Repo - OLD Group Repo **Docs** - Proposal in LaTeX **Other** - Python RSA Documentation	**PyRTL** - Main Page **DINO** - Github - Paper **RISC-V** - The RISC-V Instruction Set Manual - Assembly Register Cheat Sheet

Spring

Week 9: 5/25—5/31

Goals:

- Attend all meetings

Accomplishments:

- Added to final poster
- Collected ~more~ Data :D

Friday, May 26 (— hours):

- Group Meeting/Hackathon

Week 8: 5/18—5/24

Goals:

- Attend all meetings

Accomplishments:

- Added to final poster

- Collected Data :D

Friday, May 22 (5 hours):

- Hackathon
    - Read through some references Alvin sent about glitch attacks
    - Updated poster w/ vince's diagram and some glitch info
    - Tried to help tyler debug the secure channel module
    - Started collecting performance and area data!
    - Calculated the cycles for the RSA module we're using modeled in the paper, collaborated with Alvin to figure out the correct way to calculate

Monday, May 18 (— hours):

- Met with Alvin/Tim: Notes
- Met with Aarti: Notes

Week 7: 5/11—5/17

Goals:

- Attend all meetings
- Help Tyler integrate security modules into base processor

Accomplishments:

- Added to final poster

Friday, May 15 (2 hours):

- Met with Alvin, discussed:
    - Poster feedback
    - Implementation details

Tuesday, May 12 (2 hours):

- Worked with Tyler on poster: added content, hammered out some testing methodology details, added to background, added diagram

Monday, May 11 (— hours):

- Felt sick today, wasn't able to attend meetings

Week 6: 5/4—5/10

Goals:

- Attend all meetings
- Make rough draft of final poster

Accomplishments:

- Began poster
- Found potential model for RSA

Tuesday, May 5 (2 hours):

- Worked with Tyler on poster presentation, got formatting and sections figured out, added content

Monday, May 4 (2 hours):

- Found paper that gives an algorithm to describe the number of cycles it takes to run a hardware rsa module: Hardware Modules of the RSA Algorithm
- Met with Alvin/Tim: Notes
- Messaged Mirza about getting poster template/examples
- Met with Aarti

Week 5: 4/27—5/3

Goals:

- Attend all meetings

Accomplishments:

- Attended all meetings

Friday, April 13 (2.75 hours):

- Met with Mirza
- Met with group
- Did reflection #3

Monday, April 13 (2 hours):

- Met with Alvin/Tim: Notes

Week 4: 4/20—4/26

Goals:

- Attend all meetings

Accomplishments:

- Attended all meetings

Tuesday, April 13 (2 hours):

- Met with group
    - Helped vincent debug what was making small programs break
    - Found that the program was running for too many cycles (based on DINO riscv file specifications) and it was causing it to give undefined behaviour or start over and terminate mid process

Monday, April 13 (2 hours):

- Met with Alvin/Tim: Notes

- Met with Aarti: Notes

Week 3: 4/13—4/19

Goals:

- Attend all meetings

Accomplishments:

- Attended all meetings

Friday, April 13 (0.75 hours):

- Met with Mirza Notes

Tuesday, April 13 (2 hours):

- Met with group

Monday, April 13 (2 hours):

- Met with Alvin/Tim: Notes
- Met with Aarti: Notes

Week 2: 4/6—4/12

Goals:

- Work on branch instructions
- Attend all meetings
- Create a new timeline for Spring

Accomplishments:

- Got branch instructions working!
- Attended all meetings

Thursday, April 9 (1 hour):

- Got the rest of the branch instructions working! (BNE, BLT, BGE)

Tuesday, April 7 (my birthday! :D) (2 hours):

- Met with group
  - Got BEQ (if equal, branch) instructions working :)
  - Also successfully used a separate branch on git and merged to master to push my code! Cool stuff — reference

Monday, April 6 (2 hours):

- Met with Alvin/Tim: Notes
- Met with Aarti: Notes

Week 1: 3/30—4/5

Goals:

- Set up meetings
- Divvy up tasks to start working on

Accomplishments:

- Set up all meetings!
- Divided workload to add instruction support

Friday, April 3 (0.5 hour):

- Met with Mirza: Notes

Wednesday, April 1 (1 hour):

- Set up meetings with Aarti, Alvin/Tim, and Mirza

WINTER

Week 9: 3/4—3/11

Goals:

- Get s

Accomplishments:

- Got I

Sunday, March 8 (2.5 hours):

- Fixed stupid pytest error!

Saturday, March 7 (2.5 hours):

- Hackathon
    - Ran into stupid pytest error when trying to run tyler's testing infrastructure for the single cycle ://

Tuesday, March 3 (5.5 hours):

- Aarti meeting
    - Integrate with travis github testing
    - Send google calendar invite for every other week to Tim
    - Talk to Tim/email Tim about security research ideas
    - Keep sending weekly updates

Week 8: 2/25—3/3

Goals:

- Get single instructions running on single cycle cpu

Accomplishments:

- Got I and R type instructions running!!! :D
- Spent way more time on ERSP this week! Made real concrete progress! Ran into a crap ton of confusing problems but kept working!

Tuesday, March 3 (5.5 hours):

- Group meeting in archlab
    - Debugged single cycle, made fixes in control unit, instruction decoder, alu control unit
    - Vince and I got I-type and R-type instructions running!!! :D

Monday, March 2 (5 hours):

- Worked on single cycle simulation
    - Used ethan's simple simulator as a base

Sunday, March 2 (4 hours):

- Worked on single cycle soft version, decided this was futile
- Decided on and built good test instructions for future simulation
- Getting caught up on how everything works, tracing through the inputs/outputs in the top file
- Slowed down by some stupid import errors

Tuesday, Feb 25 (.75 hours):

- One-on-one meeting with Mirza
    - Got wrecked
    - Cried a lot
    - Lead to improvement

Week 7: 2/18—2/25

Wednesday, Feb 19 (.5 hours):

- Mirza meeting
    - START ACTUALLY THINKING ABOUT RESEARCH QUESTIONS
    - Read papers from Alvin

- Look into memory solution
- Talk to george about csr thing again
- **Demand** face time with Tim to talk about research question! (nicely)
- Tell Tim that we feel very unsure about being able to pin down a specific research question from papers and things - plz help
    - If he doesn't have a more specific idea, at least have him around to brainstorm
    - Feelings??? We WANT to engage in the process of brainstorming but we don't know how/don't know what ideas are feasible
    - Point us in a direction??
- PHASE 2 IS COMING - research 'n shit
-

Week 4: 1/27—2/2

Goals:

- Make instruction decoder
- Continue integrating modules
- Do slides for alvin + aarti meetings

Accomplishments:

- Finished first draft of instruction decoder
- Made meeting slide
- Updated proposal timeline

Sunday, Feb 2 (5 hours):

- Pyrtl hackathon in the arch lab w/ vincent, alvin, ethan, george
    - Personal accomplishments: implemented instruction decoder (module in github), corrected an error in the alu control unit, collaborated with vincent
    - Group accomplishments: got a nearly complete draft of single cycle processor, george looked into helping out with the csr

Friday, Jan 31 (.75 hours):

- Met with Aarti
    - 2-minute elevator pitch every meeting
    - Make friends with george

Week 3: 1/20—1/26

Goals:

- Begin instruction decoder
- Implement ALU control unit
- Finish filling GANTT chart + update progress
- Continue integrating modules
- Do slides for alvin + aarti meetings
- Do reflections!!

Accomplishments:

- Finished first draft of ALU control unit!
- Made meeting slide
- Did reflections

Sunday, Jan 26 (.75 hour):

- Filled out reflection google forms

Friday, Jan 24 (.75 hour):

- Met with Aarti
    - Set up weekly meeting progress slides here
    - Sit with alvin and have him help figure out the csr code
    - Find new/more time to meet with tim & come with questions - peep his office hours?

Wednesday, Jan 22 (.75 hour):

- Met with Mirza
    - To test single cycle: manually input a single instruction into a mem-block and run through the processor, see if it works.
    - See if we can avoid csr for now lol
    - There are csr instruction types ! CSRRCI, CSRRS, etc.
        - First look into branch instructions - BEQ, BNE, etc. how they connect to the csr and the csr instructions
    - Csr used in branching - check the result of the previous operation
        - Alu outputs to csr
    - Manually trace instructions
    - Have questions for tim ready by next meeting
    - Keep reading papers
    - **Next mirza meeting:** everyone pick 2, 3 instructions and trace it through OUR single cycle processor, make a circuit diagram for our processor, demo something if we can. Explain purpose of csr in some instances.

Tuesday, Jan 21 (4 hours):

- Met with group for 2 hours
- Finished first draft of ALU control unit! :D
    - Finally figured out funct3 and funct7 (which was a block for a while)
    - Looked through vincent's testing infrastructure to begin testing the module
- Researched how to implement instruction decoder
    - Read some patterson and hennessy (ch.4)
    - Looked through DINO implementation
    - Compared instructions in the RV32I base instruction set to better understand the differences (pg. 104 of The RISC-V Instruction Set Manual)

Monday, Jan 20 (2 hours):

- Began mapping out and implementing ALU control
- Used DINO diagram and RV32I instruction set to trace out functionality conceptually

Week 2: 1/13—1/19

Goals:

- Begin instruction decoder
- Begin ALU control unit
- Set up/fill GANTT chart
- Do slides for Alvin meeting

Accomplishments:

- ALU control research
- Set up GANTT chart

Friday, Jan 17 (1 hour):

- Met with Aarti
    - Revise timeline
    - Remember progress slide for next week!

Thursday, Jan 16 (1 hour):

- Looking through DINO implementation of alu control, trying to figure out inputs/outputs of the module

Tuesday, Jan 14 (3 hours):

- 2 hour group meeting
    - Set up GANTT chart
    - Redistributed workload/implementation of parts
    - Worked on implementations
    - Set plan for beginning integration of modules
- Met with Alvin: notes

Week 1: 1/6—1/12

Goals:

- Finish setting up meeting times
- Meet with Mirza, Aarti
- Consolidate ALU implementation in pyrtl
- Begin instruction decoder module implementation

Accomplishments:

- Set up all meeting times

Wednesday, Jan 8 (1 hour):

- Mirza meeting notes
    - Revise timeline
        - Account for pipelined cpu
        - More time for security research
    - Use github *projects* <- ask aarti about these
    - Use GAN chart to visualize progress
    - Make ersp week tuesday to tuesday, logs due then
    - Remember to do reflections every other week, about 2 paragraphs each
    - Revise proposal (especially timeline) and give to tim via mirza
    - Reach out to Dylan, ask Alvin/Tim about who to ask about pyrtl
- New self schedule: 30 – 60 minutes working on pyrtl
    - Half jupyter wkbks until they are completed
    - Half instruction decoder + testing until completed, then next component (or subassembly testing)

FALL

Week 8: 11/22—11/28

Goals:

- Create and give final presentation

Accomplishments:

-

Wednesday, Dec 4 (hours):

- Questions
    - Security
        - Are they implementing the SMT translator and the solver?

Tuesday, Dec 3 (1.25 hours):

- Mirza meeting notes
    - Try to reuse DINO's compiler
    - Do a more thorough **lit review**
        - Start the survey paper
    - Find countermeasures that are feasible AND interesting
    - Intel secure module -> focus our processor around this ?? ask Tim after digging into this
    - Taint flow tracking -> read this paper
    - Set reasonable expectations, don't expect to publish and don't put too much pressure on ourselves
    - Add taxonomy of attacks + identify possible attacks for us to look at in proposal
    - Revision of proposal not required, but we could if we want to
    - We should all take 3 units next quarter

Week 7: 11/22—11/28

Goals:

- Not get a concussion :)

Accomplishments:

- Got a concussion :)))))))

Sunday, Nov 24 (.5 hours):

- Edited proposal, fixed some images

Week 7: 11/15—11/21

Goals:

- Finish peer review-ready draft of proposal
- Finish pyrtl wkbk examples
- Read (at least) chapter 4 of Patterson and Hennessy
- Install DINO & try out a lab assignment

Accomplishments:

- Wrote a LOT for the proposal
- Read some more research papers

Sunday/Monday, Nov 17/18 (6.5 hours):

- Skimmed/read through some more papers & resources
    - Papers:
        - Combining Clock and Voltage Noise Countermeasures against Power Side-Channel Analysis by Jacqueline Lagasse, Christopher Bartoli, Wayne Burleson
        - A Logic Level Design Methodology for a Secure DPA Resistant ASIC or FPGA Implementation by Kris Tiri and Ingrid Verbauwhede
    - Resources:
        - RISC-V Base ISA lecture slides

        - Timing Attack - Wiki
        - Side-Channel Attacks - Wiki
        - Fault injection general info?
        - An example of failed constant time algorithm
- Worked on proposal forEVER. Who needs sleep? Not I. :))
    - IMPORTANT NOTE: I contributed so much to this draft of the proposal because I wanted to make up for not having the time to contribute as much to the very first version. This is not an indication of my groupmates slacking
    - Part 1
        - Added more reference papers/research
        - Explained concepts more in depth
            - RISC-V
            - ISA
            - PyRTL, etc.
        - Added graphic to explain better
    - Part 2
        - Outlined overall process
        - Broke it down into steps of implementing the processor in pyrtl, and multiple phases of testing
        - Added another graphic for comprehension of what a CPU even is lol
    - Version of proposal after my additions to first draft and some editing from vincent: Proposal

Week 6: 11/8—11/14

Goals:

- Finish first draft of proposal
- Do 2-3 more pyrtl examples
- Read 1-2 chapters of Patterson and Hennessy
- Install DINO & try out a lab assignment

Accomplishments:

- Practiced some pyrtl
- Organized the current trajectory of our project and divvied up workloads w/ groupmates

Thursday, Nov 14 (1 hour):

- Did most of a pyrtl example
- Briefly looked into pytest
- Arch meeting with Alvin
    - Notes

Wednesday, Nov 13 (3 hours):

- Met with group to go over processor components, divvy up work (below under HW), and decide on questions to bring up in tomorrow's meeting
- HW:
    - Read Ch.4 in Patterson and Hennessy
    - ALU
        - Compile best solutions into one ALU
        - Make ALU Control Unit
        - Make testing functions
- Lecture Notes:
    - Don't expect prior knowledge of topics (ex: side-channel attacks)

Week 5: 11/1—11/7

Goals:

- Make an ALU by thursday meeting
- Read more of Patterson and Hennessy
- Code up some pyrtl examples

Accomplishments:

- Made basic ALU
- Did 3 pyrtl workbook examples :)

Thursday, Nov 7 (4 hours):

- Made a little ALU! :D
  - Code
- Completed second and third pyrtl workbook examples found here
  - Notes
- Met with Alvin and Tim
  - Meeting was much more productive and on-track than previous meetings :)

Wednesday, Nov 6 (3 hours):

- Successfully installed pyrtl (FINALLY haha)
- Completed first pyrtl workbook example found here
- Lecture discussion notes:
  - High level diagram of CPU components
    - Divide & conquer work
  - Template on how to implement each component (class?)
  - Better communication
  - Break big problems into small, manageable tasks
  - For security: adding simple small components which tackle security issues (ex: noise generator, pointer arithmetic module, etc.)
  - Pick 1-2 important questions and make them clear at the start of the meeting
    - Feedback on ALU code and how to test our implementation(s)
    - Best practice for component modules?
  - Ask about pyrtl tutorial day?
  - We want to be able to draw the resulting circuit from our code

Monday, Nov 4 (1 hour):

- Lecture Notes about Paper:

    - Fitts' Law is an equation which has been very successful in modeling the time it takes to select an object

Sunday, Nov 3 (3.5 hours):

- Met with arch group to work together on our understanding of pyrtl
    - Began implementation of a *very* simple ALU
    - Created github repo

Week 4: 10/25—10/31

Goals:

- Browse through DINO repo, attempt some lab assignments
- Look through PyRTL examples and try to practice some
- Start reading Computer Organization and Design RISC-V Edition: The Hardware Software Interface by **Patterson and Hennesey**

Accomplishments:

- Wrote brief reflection, and filled out google form
- Identified research problems

Wednesday, Oct 30 (0.5 hours):

- Lecture notes:
    - Use latec for monday's assignment - make open/over? leaf account
    - Read paper on how to write a research paper
- Began lit search part 1

Monday, Oct 28 (2 hours):

- Lecture notes:
    - Flow chart <- linkify

- Literature search into hardware security
- Go through literature for different reasons
    - Get a sense for general field
    - How did other researchers set up their experiments?
- Where do we find these papers??
    - Look in the introduction of a paper for connected works. If not here, look through rest of paper

- Reflection about ERSP so far:
    - What do you like most about the work you have been doing?
        - I genuinely find our research topic very interesting so I really enjoy learning about it, especially when I feel that I understand a concept at a new level. I also really like my groupmates, they're fun to work with and seem just as interested in the work we will be doing.
    - What do you like least?
        - Since our group just had our first official mentor meeting last week, I feel like we're behind and I don't know how that will work out with the ERSP lecture/course schedule which is a bit stressful. I guess the uncertainty of it all is stressful in general. Alvin and Tim told us their plan for the project and it seems like a LOT. It's hard to tell if we will get anywhere close to the goals they are envisioning for this project.
    - What is the biggest concern or question you have?
        - I'm a bit concerned that the project is too large to be manageable, but I do think Tim and Alvin will be understanding of our ability or lack thereof to adhere to their loose timeline or overall goals.
- General research problems and technical challenges:
    - How can we optimize the security/performance tradeoffs in processor design?
    - What are the current security vulnerabilities in a RISC-V processor?
        - Specific to our group: what are the vulnerabilities in the DINO implementation
    - How do we implement the DINO CPU in PyRTL?
        - How do we organize our code to effectively do this?
    - How do we approach the problems of hardware security as computer scientists?

        - What can we improve so that cybersecurity can be made simpler?
    - What countermeasures can we implement to prevent or shut down physical attacks?
    - Group log of problem consensus

Week 3: 10/18—10/24

Goals:

- Meet with Alvin for first time
- Research confusing topics in DINO research paper
- Answer questions on Griswold's 2-page form
- Research a specific point of confusion in the DINO paper and teach it to my group mates

Accomplishments:

- Collected resources on confusing topics in DINO paper
- Met with Alvin and Tim for our first meeting!
- Did group teaching activity

Thursday, Oct 24 (2 hours):

- Met with Alvin and Tim for our first meeting!
    - ~Notes~ -synthesized key points at top

Wednesday, Oct 23 (1.25 hours):

- Did group teaching activity
    - Taught about instruction types (resources and synthesis of understanding below on 10/22)
        - My slides
    - Learned about:
        - Pipelines (Tyler) - slides
        - Branch prediction (Vincent) - slides
        - Hazards (Ethan) - slides

Tuesday, Oct 22 (4 hours):

- Researched topic from DINO paper: **Instruction types**
- Helpful resources:
  - General RISC-V Educational Materials
  - Oakland University Lecture
  - Opcodes and Operands - Youtube
  - Machine Code Instructions - Youtube
  - Base ISA - Youtube
    - slides
  - Privileged Instructions
  - Immediate Encoding Variants via my dude Cesar on stack overflow
- Synthesis: Each instruction in an RV32I CPU is 32 bits long and contains opcode (operation code which defined the operation to be performed on the data), and operands either in the form of immediate data that is passed in or register addresses in memory of where to retrieve the data, and the register where you store the result of the operation. If there is a funct3 or funct7 in the instruction, you combine it with the opcode to get the full operation. RV32I supports R, I, S, U, B, and J instruction types which are easily visualized in this image.

Monday, Oct 21 (1.5 hours):

- Gave a short presentation on our takeaways from DINO paper - Slides
- Researched confusing topics and reread/took notes on The Davis In-Order (DINO) CPU: A Teaching Focused RISC V CPU Design
  - Notes

Week 2: 10/11—10/17

Goals:

- Set up meeting time with group
- Read + annotate + take notes on RISC-V research paper

Accomplishments:

- Set arch group meeting time
- Got through part 1 of research paper assignment :)

Tuesday, Oct 15 (3 hours):

- Read and annotated The Davis In-Order (DINO) CPU: A Teaching Focused RISC V CPU Design by Jason Lowe-Power and Christopher Nitta
    - Took notes & answered questions: Notes

Monday, Oct 14 (0.75 hours):

- Read How to Read an Engineering Research Paper by William Griswold, UCSD
- Picked Thursdays 3:30–4:40 beginning Oct. 24 as arch group meeting time

Week 1: 10/04—10/10

Goals:

- Set up research log
- Record ERSP preliminary thoughts
- Reflect on research logs

Accomplishments:

- Set up log.
- Wrote log reflection: below

Wednesday, Oct 9 (1.25 hours):

- Set up log. I chose to use Google docs for my first log because it is the platform I've used in school for years so I am very familiar with it.
- **Wrote log reflection :**
- This is what I am most excited about in ERSP:
    - I am very excited to work in a team environment on an interesting project where I'm learning new things at every step of the way. My high school had engineering classes that allowed for

that sort of dynamic and I've really missed it. I'm also interested in learning what research is really like. I had an internship at a cybersecurity software company over the summer, and it made me reconsider whether or not I want to go into industry after college so I'm hoping this program could help me narrow down on a career path.

- This is what I am most nervous about in ERSP:
  - I'm worried that the workload required for ERSP on top of my other classes will become too overwhelming. I do struggle with time management and that's something I will be working to improve quite a bit this quarter (and this whole year) as my academics and other commitments become more demanding. I'm also worried that I won't be able to contribute enough to my research group. I think that other people in my group are probably coming into this program with a lot more knowledge of CS in general and I don't want to be unhelpful because I am behind at the start.
- Reflection on example logs:
  - A couple of them had a "compilation of useful resources" at the top which I think would be helpful for a long project like this when you need specific links/etc. on a regular basis.
  - I also liked that Miranda Parker's log included links/downloadable text files of extra notes for a particular event or part of the project - I think it makes the log more clean-looking while also allowing you to include extra information
  - I think the logs will definitely be useful for us to stay on track, keep our goals in mind, and keep our thoughts organized. They also allow the prof./grad students involved to see our contributions in a dictated way (as opposed to discussions in meetings), where they can also access any relevant links/resources if needed.
  - The students who wrote the example logs did seem to meet most of their goals, it's hard to tell without reading through each one completely but they did seem to improve their manner of, and thoroughness of documentation as the year progressed.