Project Idea: Evaluate performance of LLM for verification of hardware designs.

1. Choose your design. Understand the in-out of it.
2. Generate assertions (SVA) for a given Verilog design.
3. Check if the generated assertions are syntactically and semantically correct (using a model checking tool – SVA assertion checker, SVAUnit, SymbiYosis)
    a. You can create an API to call such model checking tool to check this step automatically
4. Check the consistency of the generated assertions
5. Write the correctness properties on your own. Compare the quality of the generated assertions.
6. Check the completeness of the generated assertions
7. Use proper prompt to improve to improve the quality of the generated assertions.
8. Finally, formally verify the design chosen by you.

Rules:

1. No two groups have same design and LLM. The same design can be taken but to be used with different LLM.
2. You need to document each steps of the above.
3. You cannot take any designs from the existing works (like given two papers).
4. You must understand the design well. Choose control dominated designs.
5. Insert the design name and link to it and LLV that you are using in the shared document. https://docs.google.com/spreadsheets/d/1e95dqPCwhr7-PZOxaAHsSVcHLHGkK2mtaGGJIiXHjw0/edit?usp=sharing
6. If we have impressive results, we can plan to publish the result.

Resources:

1. Learn System Verilog Assertion (SVA)
    a. Chapter 2 from the book "A Roadmap for Formal Property Verification" by Pallab Dasgupta, Springer, 2006.
    b. Any online resource
2. Some recent paper on the similar direction.
    a. https://arxiv.org/pdf/2309.09437.pdf
    b. https://arxiv.org/pdf/2306.14027.pdf
3. Open hardware verification:
    a. https://github.com/ben-marshall/awesome-open-hardware-verification#openhw-group-functional-verification
4. Choose any design from
    a. OpenTitan: https://opentitan.org/book/hw/ip/index.html
    b. https://github.com/ben-marshall/awesome-open-hardware-verification
    c. https://github.com/SpinalHDL/VexRiscv
    d. https://github.com/open-sdr/openwifi
    e. Take your own design (logical/practical design) from any other source
    f. **Suggestion:** Do not take the entire design. Take small driver, interface protocol, some controller behavior, etc. Understanding the design that you are taking is important to progress in this project.