Features: {

What does it mean to be accurate?

What does it mean to support speed?

What does it mean to have options?

Option Family 1: Performance family options could mean varying cache and register file sizes and instantiating subcomponents (co-processors) and memory hierarchy.

}

**Scratch Ideas in Work –**

1. Block diagram for ISA execution and data flow
2. Precision consideration
3. Block level to microarchitecture logic as implemented in software – mapping.
4. Tree for OPCODE walk through – Could be part of Block Leve data flow – initial block, probably a fastest way to categorize the “next instruction”.
5. Microarchitecture logic – mapped to blocks – on reverse mapped to functions (as in Rust code functions).
6. Microarchitecture Logic + Block Level Execution Flow + …
7. How about SaaS Model, essentially allowing for execution over cloud for this Simulator Model?
8. How would you run time control this simulator, purely GUI based control or would there be CLI based control supposed? Would having a Python based control embedded with simulator module to control it’s parameter be better?
9. Co-processors are not Functional Units across all standard cores (RISC-V Shakti).
10. What defines the state of the system:
    1. What all the processor sub components be inclusive in defining state of the system?
    2. How will memory hierarchy be modeled in terms of delay statistics – delay(x), delay(x+1), …
11. What is the cycle of instruction in RISC-V? This in order to

* Simulating the performance of the microarchitectural implementation of an ISA is crucial component for design space exploration of next-generation designs.
* Sniper proves a range of flexible simulation options to explore a variety of different homogeneous and heterogeneous multicore architectures, as well as Python based runtime environment that allows for analysis and simulator control.