Hardware Trojan Detection via Chisel HDL

Compare different implementations of a component for the *PULPino* system

- While we cannot use TLM/RTL models from others, we can create our own TLM/RTL models (using *Chisel HDL*) for components of the *PULPino* system.
- The output of these different implementations is given to a majority gate (i.e., use majority voting).
- Components that yield a different output from the majority of the components have hardware trojans.
- Use proof-carrying code techniques to determine which hardware component went bad, and when and how (i.e.., sequence of input patterns) did it go bad.

Hardware Trojan Detection via Chisel HDL - OLD slide

Compare different implementations of a component for the *PULPino* system

- Use techniques from fault tolerance for detecting hardware trojans.
- While we cannot use TLM/RTL models from others, we can create our own TLM/RTL models (using *SystemC*, *Verilog*, and/or *Chisel HDL*) for components of the *PULPino* system.
- The output of these different implementations is given to a majority gate (i.e., use majority voting).
- Components that yield a different output from the majority of the components have hardware trojans.

Use the aforementioned software redundancy technique to determine which hardware component went bad, and when and how (i.e.., sequence of input patterns) did it go bad.