Formally verifying Valida zkVM

May 2024 - Valida

Formal Land

Past projects and Valida

1. L1 of Tezos
2. coq-of-rust
3. coq-of-python
4. coq-of-solidity
5. Valida zkVM

L1 Tezos

- Formal verification of the core of Tezos
- 80% of files with some proofs
- Interpreter, storage, backward-compatibility
- https://formal-land.gitlab.io/coq-tezos-of-ocaml/

coq-of-rust

- Import Rust code to the proof assistant Coq
- https://github.com/formal-land/coq-of-rust

coq-of-python

- New project, ongoing
- For the EVM specification
- https://github.com/formal-land/coq-of-python
- Combined with coq-of-rust to verify the Revm version of the EVM

coq-of-solidity

- Just starting
- Provides a verification tool for Solidity with Coq
- Reusing the same techniques as coq-of-rust

Verifying Valida zkVM

Our proposition

1

FORMALIZE THE CODE

- Import Valida to Coq with coq-of-rust
- Process the output so that it is suitable for formal verification

2

SHOW SOUNDNESS

- Define in Coq the RISC version you use
- Verify the arithmetization of all the operations!
- The Risc semantics should match the Valida implementation

Thanks