I was talking with@alberto about our shielded resource machine report, and we agreed on the value of discussing here the outline structure of the document we are envisioning.

Protocol Details

Using the ZCash specs as a blueprint, I suggest it would be best to have a abstract protocol which is back-end agnostic and contains descriptions of:

- Cryptographic schemes: describe in detail the cryptographic schemes used across all protocol.
- Key Components and Derivation
- Nullifier Derivation
- Compliance circuit
- Resource Logic: I am not sure on how to expand this section. I would start by writing examples, for instance the ones present on the Taiga repo are worth writing down.
- Balance and Binding Signature
- Constants
- Encodings

These section of the report could also come with Cairo code snippets. I am aware that the content above partially overlaps with the Resource Machine specs, however it is best for the sake of ease in writing to ignore such overlaps during the fist version of the document. We can discuss later about what should be kept and where. It could also be useful to split this section in two (one more abstract and one more concrete), however I do not think it is necessary for the time being.

Back-end Specifics

Afterwards, sections specific to each used back-end (Cairo, Risc0, Halo2):

- Back-end specification:
- Finite fields
- Curves
- Hash function parameters
- Finite fields
- Curves
- Hash function parameters
- Benchmarks

In practice

I would write a rather vertical first draft concerning the cairo resource machine implementation, which would help visualizing the abstractions better.

A question we should address is: do we need to write different compliance circuits (in juvix) for different back-ends? I am aware that Alberto is writing a Rust implementation of the compliance circuit in Risc0, and I wonder if we could reuse the compliance circuit written in Juvix for Cairo,

@xuyang @vveiln @cwgoes @Lukasz