zkLink: An Aggregated Rollup Infrastructure Based on Zero-Knowledge Technology

Abstract

The proliferation of new blockchain networks has resulted in fragmented liquidity and increased complexity in deploying dApps across multiple blockchains. The absence of a platform that facilitates secure trading of cross-chain assets with interoperability would bring significant challenges for the mass adoption of DApps. Developers are confronted with the intricacies of deploying dApps across different chains due to variations in programming languages and tools.

In response to these obstacles, zkLink is actively developing an aggregated rollup infrastructure, secured by zero-knowledge proof and multi-chain state synchronization, to interconnect various Layer 1 (L1) and Layer 2 (L2) ecosystems. By aggregating native crypto assets from separate blockchain networks onto a unified platform, zkLink simplifies the complexities associated with deploying dApps across multiple chains. Additionally, it aims to unify fragmented liquidity across ecosystems and significantly reduce transaction costs.

1 Introduction

With the rapid advancement of the blockchain space, a multi-chain, multi-layer landscape has emerged with the presence of layer 1 blockchains like Ethereum, Solana, Avalanche, and Ethereum layer 2 rollups such as Arbitrum, zkSync, Starknet. Users navigate between different L1 chains and L2 rollups to meet their specific needs, utilizing a diverse array of crypto tokens.

While the proliferation of new chains and rollups presents significant value for crypto users worldwide, it also poses unforeseen challenges of ecosystem fragmentation. For users, this leads to increased costs and security risks for cross-chain transactions, and for developers, a complex multi-chain dApp development environment.

zkLink addresses the above issues by building an aggregated rollup infrastructure to achieve cross-chain liquidity aggregation and simplify multi-chain dApp deployment. Through multi-chain state synchronization, dApps can securely access the liquidity from any connected chains. By leveraging zero-knowledge proof technology, it provides a high throughput, low-cost and secure environment for the dApps of any kind.

Key features of zkLink infrastructure include:

Native Asset Aggregation: Applications using zkLink infra solution will be able to access the native tokens across the connected L1s and L2s, allowing users to trade multi-chain assets on a unified user interface.

Low Fee and High Scalability: zkLink's modular stack provides unparalleled scalability for dApps building on top of our ecosystem. Zero-knowledge proof technology can dramatically reduce execution costs and provide for a blazing-fast user experience.

Minimum Security Assumption: Every transaction on zkLink infra undergoes verification via zero-knowledge proof. Third party asset bridges are not needed for bridging assets onto zkLink infra, thus eliminating cross-chain asset bridging risks. Multi-chain state synchronization is achieved via transmitting sync hash of on-chain transactions, which prevents malicious operation like sending fraud deposit information.

Currently, zkLink serves two product lines:

zkLink Nova: zkLink Nova is an EVM-compatible, aggregated Layer 3 rollup network built on top of Ethereum and its Layer 2 Rollups. DApp developers can deploy Solidity smart contracts on Nova’s open platform and have immediate access to liquidity and native assets from all the integrated networks, such as Ethereum, Arbitrum, zkSync, Linea, and many others. Nova inherits Ethereum security by achieving multi-chain state synchronization via an Ethereum smart contracts, which forwards on-chain transaction sync hashes through the canonical rollup bridges.

zkLink X: zkLink X is an aggregated rollup infrastructure for customized high performance applications with their own sovereignty. zkLink X serves as a middle-ware that abstracts away the complexity of multi-chain deployment, making developers feel like building on a single chain with multi-chain liquidity. Its modular architecture allows developers to customize the key components, for example, DA solution and multi-chain settlement scheme, to meet diverse demands of different use cases. It provides a trading-specific-zkVM that empowers high-throughput, low-cost App Rollup solution for high performance financial applications such as order book DEX.

2 zkLink Protocol Design

2.1 Protocol Overview

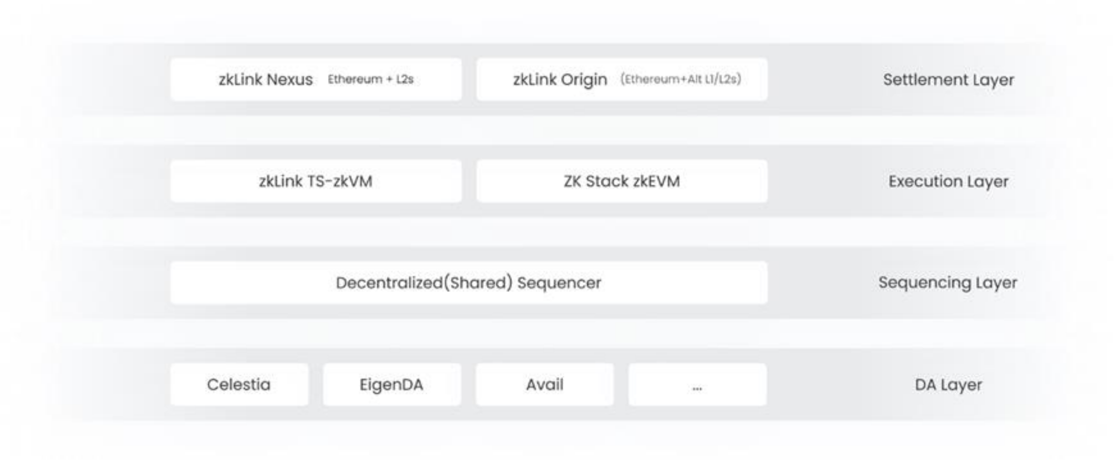

Figure 1: zkLink Protocol Design

zkLink Protocol is an aggregated ZK-Rollup Infrastructure with a modular architecture, allowing for various components or modules to be easily interchanged, upgraded, or added as needed. This modular approach provides flexibility and scalability by combining multiple specialized blockchains and technologies.

zkLink Protocol is composed of four layers: the settlement layer, the execution layer, the sequencing layer, and the DA layer. These four layers are decoupled for customized rollup deployment. The following sections will introduce the protocol from the perspective of the four layers. It is important to note that the core value proposition and most innovations of the zkLink protocol are related to the settlement layer solutions and the execution layer solutions.

2.2 Settlement Layer

A classic ZK-Rollup network typically selects a single chain, i.e, Ethereum as the settlement layer to verify the proofs and settle the transactions. The settlement layer maintains the security and integrity of off-chain transactions.

In comparison to a classic ZK-Rollup architecture, zkLink proposes an aggregated ZK-Rollup, a new type of ZK-Rollup architecture which integrates with multiple blockchains. In order to securely aggregate native token assets across L1s and L2s onto one single platform, users' assets are locked inside the rollup bridge contracts deployed on the connected chains. Additionally, the aggregated ZK-Rollup securely synchronizes multi-chain state via sync hash of on-chain transactions using canonical rollup bridges, preventing malicious operations like fraud deposit.

zkLink offers two types of proprietary settlement solutions, zkLink Nexus and zkLink Origin, each designed to meet different network integration requirements.

Solutions	Supported Networks	Security Assumption
zkLink Nexus	Ethereum and its L2s	Ethereum Equivalent
zkLink Origin	Ethereum, Alt-L1s and L2s	The Sequencer and the Light Oracle Network cannot collude for malicious activities

Table 1: zkLink Nexus vs. Origin Settlement Solutions

2.2.1 Working Principle of An Aggregated ZK-Rollup

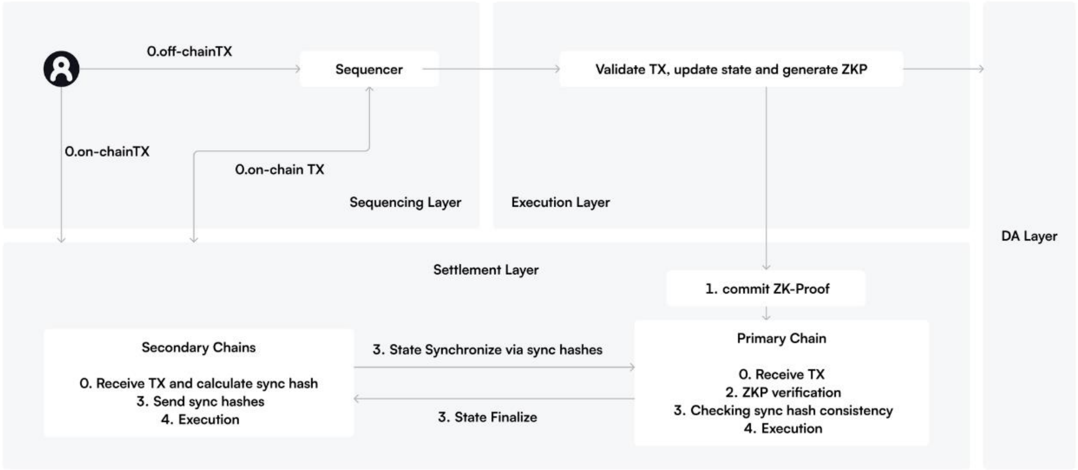

Figure 2: zkLink Settlement Architecture

In the architecture of an aggregated ZK-Rollup, on-chain transactions such as deposits will be relayed to the rollup network in real-time to deliver the best user experience. However, the hard finality of every transaction is achieved by multi-chain settlement, a new settlement paradigm which depends on the result of ZKP verification and multi-chain state synchronization.

In order to optimize the verification cost, only one chain among the connected chains will be designated as the **primary chain**, which is responsible for ZKP verification. While the other chains will act as **secondary chains** that do not need to execute ZKP verification, through multi-chain state synchronization, it is equivalent to completing the verification on all chains.

The working principle of a classic ZK-Rollup typically contains three stages: the Commit stage, Prove stage, and Execute stage. zkLink's aggregated ZK-Rollup architecture introduces an additional Synchronization stage following the Prove stage. The Synchronization stage validates the consistency of all on-chain transactions in a transaction batch verified by ZKP.

The 4 stages are briefly described as below:

1. Commit: The sequencer submits the zk-proof and transaction batch to the verifier contract on the primary chain.

2. Prove: The zkLink contract on the primary chain verifies the validity of the zk-proof.

3. Synchronization: The transaction sync hashes of secondary chains are forwarded to the primary chain via secure message channels. The primary chain verifies if sync hashes are consistent with the on-chain transactions previously relayed by the sequencer. Upon the verification of both ZKP and on-chain transaction consistency, the transaction batch can be finalized and the batch root will be sent to secondary chains for execution.

4. Settlement Execute: After receiving the batch root of the transactions including the Merkle Proof for fund withdrawal, fund withdrawals will be approved and executed.

2.2.2 Nexus: Settlement on Ethereum and its L2s

The zkLink Nexus settlement solution only connects to Ethereum and its L2s. Nexus introduces a new paradigm of multi-chain synchronization that maintains inherited Ethereum security of L2s.

In the stage of multi-chain synchronization, transaction sync hashes from secondary chains are forwarded to the primary chain via Ethereum L2s' canonical message bridge. Upon the successful verification of both ZKP and on-chain transaction consistency, batch root will be sent to secondary chains via canonical message bridge for execution.

As the correctness of the states and transactions of L2s is finalized on Ethereum via validity proof (for ZK-Rollup L2s) and fraud proof (for Optimistic-Rollup L2s), the additional multi-chain state synchronization is finalized via Ethereum. Thus, the zkLink Nexus L3 rollup inherits the security of Ethereum.

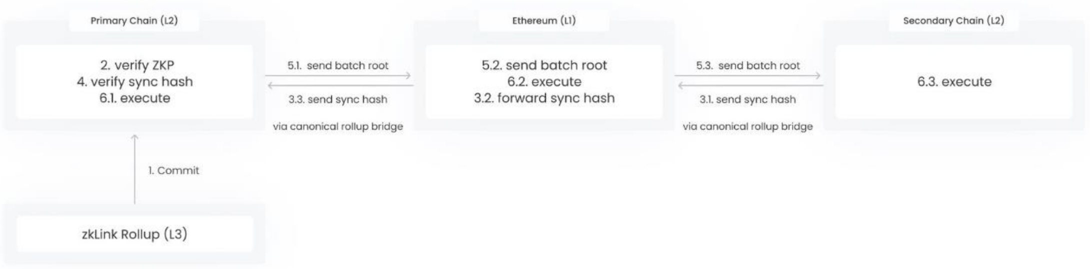

Figure 3: zkLink Nexus Settlement Process

2.2.3 Origin: Settlement on Ethereum, Alt-L1s and L2s

In contrast to Nexus, zkLink Origin's architecture allows for the integration with Alt-L1s like Solana and Avalanche, in addition to Ethereum and its L2s.

A zkLink Origin rollup settles the transactions and transition of states on networks connected, as long as one of the networks supports zk-SNARKs proof verification. To establish a fast and secure communication mechanism between the alt-L1s and Ethereum, zkLink introduces a Light Oracle Network for cross-chain message transfer.

The security assumption of zkLink Origin is that the rollup sequencers, responsible for batching transactions, cannot collude with all nodes of the Light Oracle Network for malicious activities.

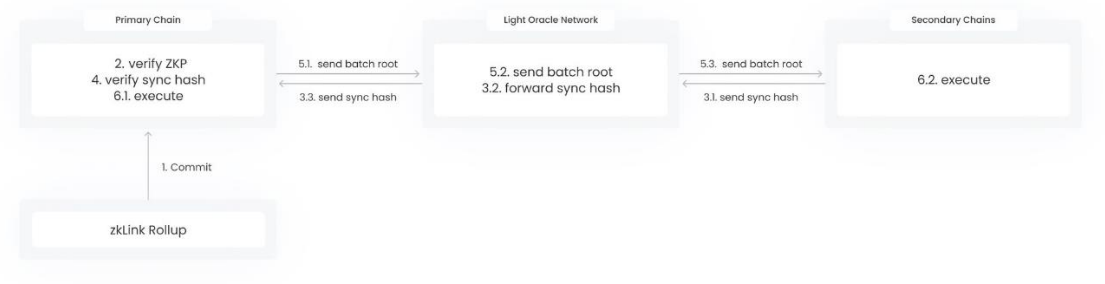

Figure 4: zkLink Origin Settlement Process

2.3 Execution Layer

Execution entails executing transactions that update the state correctly. Thus, execution must ensure that only valid transactions are executed, *i.e.*, transactions that result in valid state machine transitions.

2.3.1 TS-zkVM for Trading-Specific dApps

zkLink Trading-Specific-zkVM (TS-zkVM) is based on tailored core ZK circuits and Risc0 zkVM extension ZK circuits. The TS-zkVM offers high-throughput and low-cost execution for App Rollup developers. It supports various order book product features, including but not limited to spot trading, derivatives trading, NFT trading, etc.

2.3.2 zkEVM for Universal dApps

In zkLink's modular architecture, developers have the flexibility to build their own customized rollups based on open-sourced zkEVM frameworks like ZK Stack and Polygon CDK. Based on zkLink aggregated rollup infrastructure, dApp developers can easily build Solidity written app-rollups that have the access to the aggregated liquidity from multiple blockchains and rollups.

Based on ZK Stack, zkLink has officially built zkLink Nova (refer to section 3.1), a permissionless, aggregated zkEVM Layer 3 rollup for universal dApps. dApps can have access to the aggregated liquidity from Ethereum and its L2 networks, while still inheriting the security from Ethereum.

2.4 Sequencing Layer

The sequencing layer is a pivotal component in rollup systems, primarily responsible for receiving user transactions, sequencing the transactions and bundling them into batches. These batches are then committed to the settlement layer. Additionally, in scenarios where the system employs an external DA layer, the sequencer also ensures efficient transmission of transaction data to the DA layer.

Similar to other rollups, zkLink starts with a centralized sequencer model. While this approach offers certain development efficiencies, it also presents challenges and risks, such as potential single point of failure, transaction censorship and issues around miner extractable value (MEV), affecting network fairness and transparency.

To address these concerns, the zkLink protocol aims to incorporate decentralized sequencer solutions. These solutions, including platforms like Espresso, Astria, and Fairblock, aim to mitigate centralization risks by processing and validating transactions across a distributed node network. This strategy will not only boost network security and transparency but also strive to offer a more secure, equitable, and efficient rollup solution to its users.

2.5 DA Layer

Data Availability (DA) entails making the transaction data available. A DA layer is a critical component for rollups to ensure the capability to reconstruct the rollup states when the rollup service halts unexpectedly.

Firstly, zkLink by default supports the primary chain as the DA layer.

Secondly, zkLink supports Validium mode, which involves an external DA solution. zkLink will integrate various third party modular DA solutions, such as Celestia, EigenDA, Avail, etc., to meet diverse demands from developers.

Furthermore, zkLink DAO will have the option to organize a Data Availability Committee (DAC). This DAC serves as another option for developers, providing an additional choice for data availability assurance.

3 zkLink Nova: The Industry’s First Aggregated Layer 3 Rollup Network

3.1 zkLink Nova Overview

zkLink Nova is the industry's first aggregated Layer 3 zkEVM Rollup network built on top of Ethereum and Ethereum Layer 2 rollups (L2s). zkLink Nova, based on zkEVM of ZK Stack, is an EVM-compatible, open platform for simple and fast smart contract development of any kind. zkLink Nova's platform allows for scattered assets across Ethereum Layer 2s to be aggregated for interoperable transactions. zkLink Nova is secured by zero-knowledge proof technology, charges extremely low gas costs, offers fast finality, and inherits its security from Ethereum based on the zkLink Nexus solution.

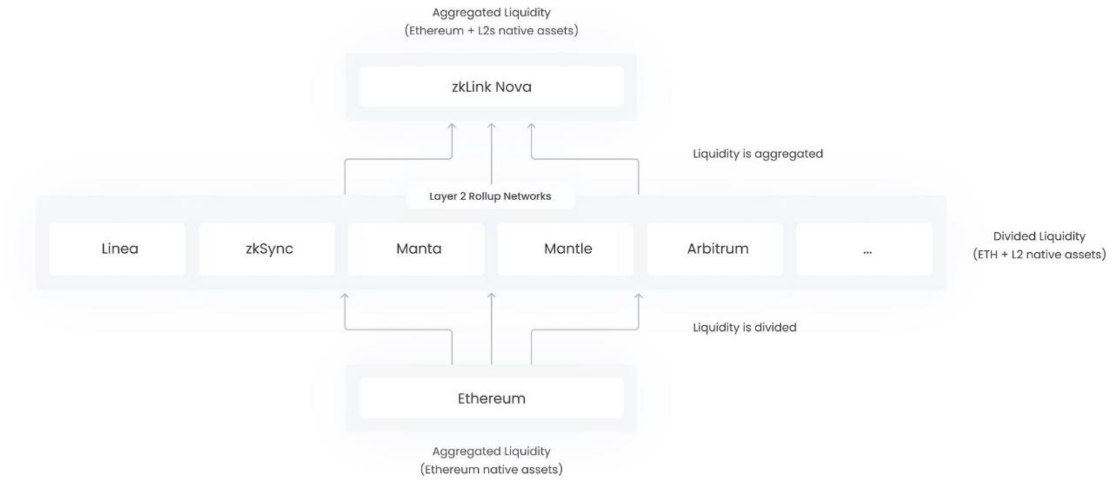

Figure 5: zkLink Nova Aggregation Illustration

3.2 zkLink Nova Key Features

3.2.1 Native Asset Aggregation

Users can deposit assets from the Ethereum Layer 1 as well as Ethereum Layer 2s directly to zkLink Nova. These assets will be locked inside the canonical rollup bridge contracts hosted on the source chains before entering the zkLink Nova network. This feature allows for applications on Nova to have access to all the connected Layer 2s' native tokens, thereby allowing users to trade their multi-chain assets with interoperability.

3.2.2 Supporting Universal dApps

zkLink Nova is EVM-compatible by utilizing the zkEVM of ZK Stack. DApp developers can deploy any kind of Solidity smart contracts such as DEXs, Lending, GameFi, SocialFi and more on Nova's open platform. These dApps have immediate access to liquidity and native assets from all the integrated networks, such as Arbitrum, zkSync, Linea, and many others.

3.2.3 Stack Agnostic to the Supported Networks

zkLink Nova is stack agnostic, meaning it can connect to rollups of different stack choices, including ZK Rollups, Optimistic Rollups, and any additional stacks they're built upon. While this approach sacrifices the atomic interoperability of cross-rollup transactions, it offers the broadest liquidity that can be aggregated from the entire Ethereum ecosystem.

3.2.4 Low Fees and High Scalability

zkLink Nova's modular stack provides maximal scalability for dApps building on top of the ecosystem. ZK Stack is used by zkLink Nova as the execution layer, which dramatically reduces execution costs and provides a blazing-fast user experience. In Validium mode, an external DA solution will further reduce the data portion of transaction costs on the network for end users.

3.2.5 Ethereum Equivalent Security

zkLink Nova inherits Ethereum security by directly settling on Ethereum through the zkLink Nexus settlement layer solution. Every transaction on Nova first undergoes verification via zero-knowledge proof followed by multi-chain state synchronization via the canonical rollup bridge. Sync hashes forwarded via canonical rollup bridges are used to check if the data commitment for ZKP is consistent with the transactions on all host chains. This prevents security risk of a malicious node operator submitting false on-chain transactions.

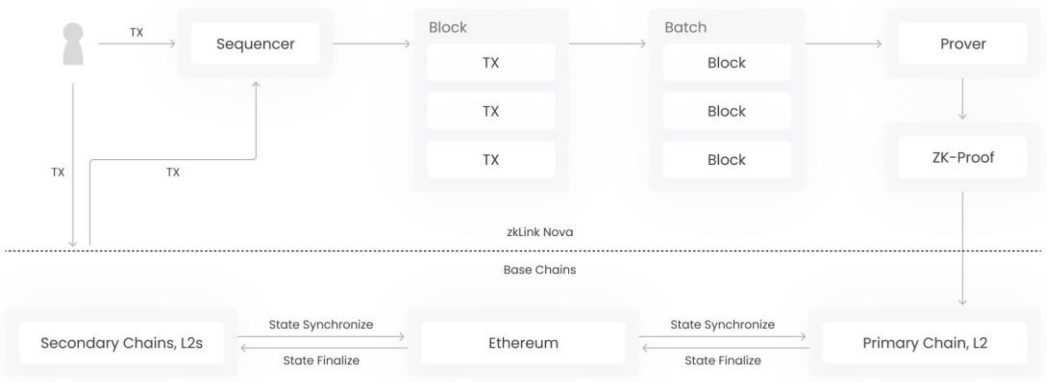

Figure 6: zkLink Nova Transaction Life Cycle

4 zkLink X: The App-Specific Aggregated Rollup Infrastructure

4.1 zkLink X Overview

zkLink X provides a modular infrastructure for aggregated App-Rollups with their own sovereignty. Applications using zkLink X App-Rollup solution will be able to access the native tokens across the connected L1s and L2s, allowing users to trade multi-chain assets on a unified user interface. Cross-chain asset bridges are not needed in the process, thus avoiding cross-chain asset bridging risks and bridging fees.

zkLink X decouples the four layers of the rollup framework and provides a fast and customized rollup deployment solution with SDKs and APIs. Developers have the freedom to choose which building blocks they need. zkLink X focuses on the development of settlement layer (Nexus and Origin solutions) and execution layer (TS-zkVM) solutions, and will integrate third party modular solutions for DA layer and sequencing layer, allowing developers to customize the key components to meet diverse demands of different use cases.

-**Network Integration and Settlement Layer Solution**. Developers can choose which chains the App Rollup can access to, including but not limited to: ETH, BNB Chain, Avalanche, Polygon PoS, Solana, zkSync, Starknet, Scroll, Polygon zkEVM, Linea, Taiko, Arbitrum, Optimism, Base, etc.

-**Execution Environment**: TS-zkVM, zkEVM

-**Decentralized Sequencer**: Espresso, Astria, Fairblock, etc.

-**Modular DA Solutions**: In addition to Ethereum, developers can choose Celestia, EigenDA, Avail, DAC organized by zkLink, etc.

4.2 Trading Specific zkVM for High Performance DEX

The TS-zkVM constructed by zkLink is a high-efficiency ZKP execution environment specifically designed for high-performance financial products such as CLOB DEXs. TS-zkVM extends and integrates with the risc0 zkVM outside of the precompiled core circuit.

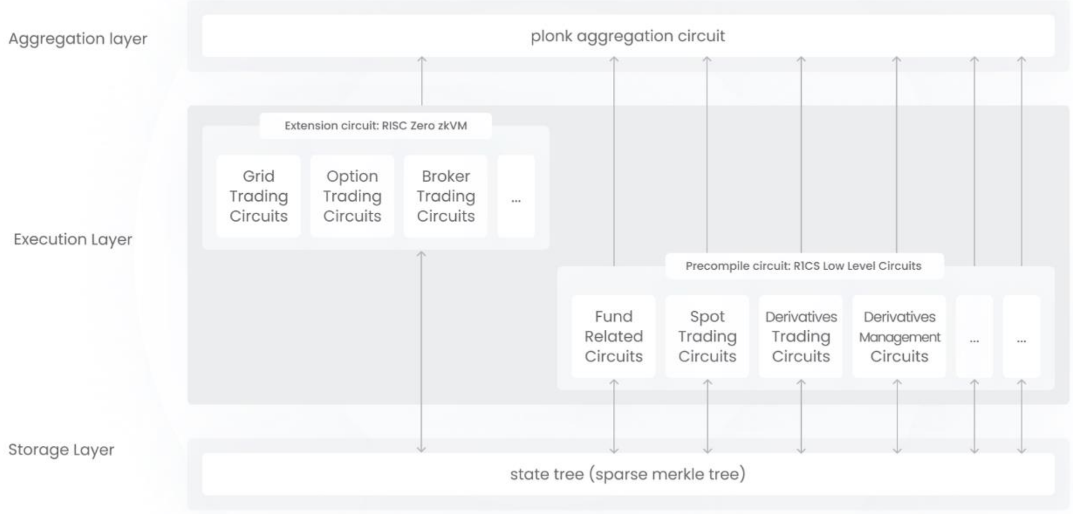
Aggregation layer
plonk aggregation circuit
Execution Layer
Extension circuit: RISC Zero zkVM
Grid Trading Circuits
Option Trading Circuits
Broker Trading Circuits
...
Precompile circuit: RICS Low Level Circuits
Fund Related Circuits
Spot Trading Circuits
Derivatives Trading Circuits
Derivatives Management Circuits
...
...
Storage Layer
state tree (sparse merkle tree)

Figure 7: TS-zkVM Architecture

The above figure shows a high-level overview of the TS-zkVM architecture, which is divided into three sub-layers.

4.2.1 Storage sub-layer of TS-zkVM

The storage of zkLink TS-zkVM utilizes a Sparse Merkle Tree (SMT) to store the state. This data structure provides the system with an efficient and secure way to maintain and verify changes in the state. Compared to the SMT in the EVM, the customized and optimized SMT of zkLink TS-zkVM is more streamlined and better adapted to the needs of high-frequency financial transaction scenarios. With in-depth customization, the SMT has been specifically tailored to suit particular business needs, thereby achieving exceptional zero-knowledge proof performance.

4.2.2 Execution sub-layer of TS-zkVM

The execution sub-layer of TS-zkVM is divided into two parts, each designed for different performance and scalability requirements:

- **Precompiled circuit**: Requires high performance and includes various transaction-related circuits, such as:
    - Fund Related Circuits (Deposit, Withdrawal, Transfer)
    - Spot Trading Circuits
    - Derivatives Trading Circuits
    - Derivatives Management Circuits (Liquidation, ADL, Funding)
    - Oracle Verification Circuits
    - Authorization Circuits (Passkey Verification, Social Login Verification)

  These circuits are specifically optimized to handle corresponding financial transactions, ensuring efficiency and high response speed when the system processes transactions. Each sub-circuit corresponds to different types of transaction operations or financial instruments and handles the most performance-intensive tasks.

- **Extension circuit:** The extension circuit is built using risc0 zkVM on the same Sparse Merkle Tree (SMT), and the proofs generated by the zkVM are subsequently verified by the plonk aggregation circuit in the aggregation layer. Developers will be able to perform custom operations using risc0 zkVM, such as:
    - Grid Trading Circuits
    - Option Trading Circuits
    - Broker Trading Circuits

  An SDK based on risc0 zkVM will be provided, which includes common operations on the TS-zkVM SMT, such as: verifying a Merkle Tree Proof for a specific account, checking if the balance change of an account complies with certain patterns, etc.

4.2.3 ZK Proof Aggregation sub-layer of TS-zkVM

The aggregation sub-layer is responsible for aggregating the different transaction proofs and producing a proof to verify the correctness of the entire batch of transactions, which achieves increased efficiency in proof verification and reduced on-chain costs for transactions.

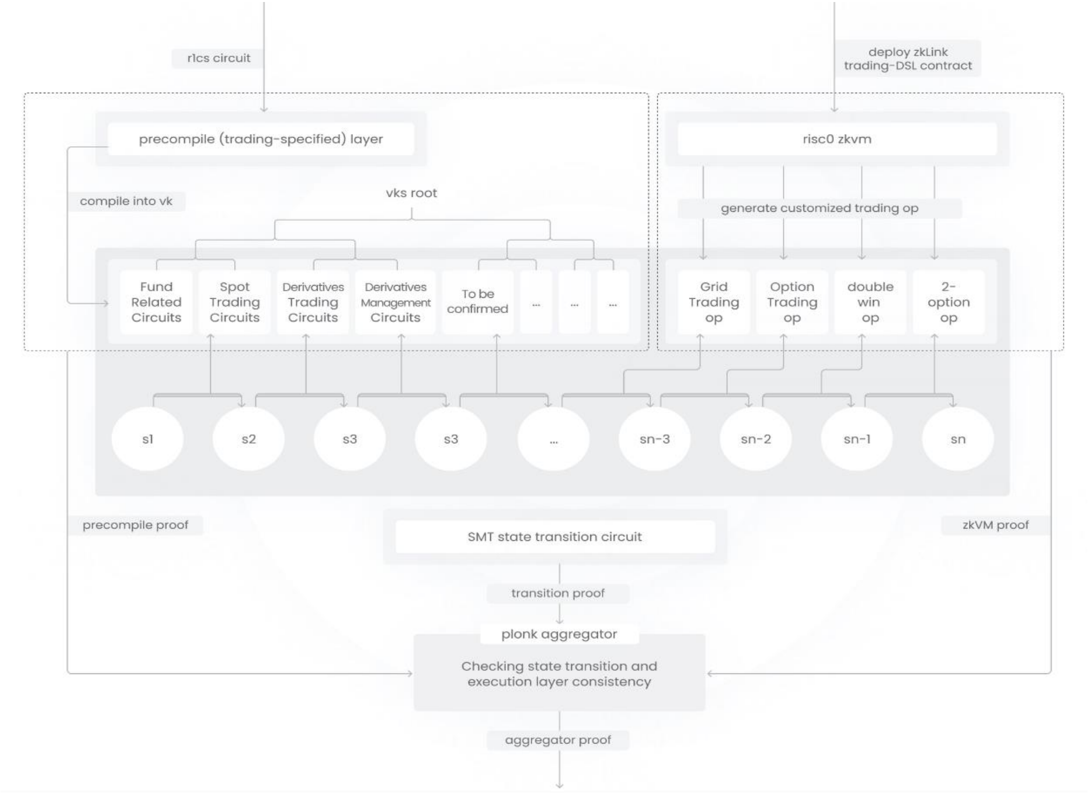

Figure 8: Aggregation Detail of TS-zkVM

Aggregation Details

The aggregation sub-layer is composed of the following modules:

- **Precompile Module**: This module includes trading-specified precompiled circuits, which need to be compiled into verification keys (vk). This provides the system with a flexible way to generate customized circuits for specific types of transactions and integrate them into the entire verification system.
- **SMT State Transition Circuit**: The state transition circuit uses an SMT to record states, with changes in the root representing all users' ledger changes, ensuring the correctness and consistency of the system state. Each state transition (S1, S2, S3, ..., Sn) is constrained by the circuit to generate a transition proof.
- **zkVM**: This section demonstrates how to deploy zkLink Trading-DSL contracts using risc0 zkVM to generate custom trading operations (op). This mechanism allows users to customize trading logic and ensures that these logics are correctly executed and verified.
- **PLONK Aggregator**: The PLONK aggregator is responsible for checking the consistency of state transitions and the execution layer. It integrates proofs from

different sources and generates an aggregated proof. This process enhances verification efficiency and reduces the costs required for on-chain verification.

- **Proofs**: The architecture includes various types of proofs:
  - **Precompile Proof**: Ensures the correctness of the precompile layer.
  - **Transition Proof**: Ensures the correctness of state transitions.
  - **zkVM Proof**: Ensures the correctness of custom trading operations.

Interactions Between Modules

- Application Specific Circuits are eventually compiled into **verification keys**, and these verification keys are stored in the root node of the vk tree.
- zkVM allows for the deployment of **Trading-DSL** contracts, generating custom trading operations.
- Precompiled circuits, risc0 zkVM, and SMT State Transition Circuits are interconnected, ensuring the correctness of each state change.
- All proofs are collected by the Aggregator and verified in the aggregation circuit — to check if the transition set commitment in the transition proof is included in the **(Algorithm id, Input, Output)** tuple vector of the precompile proof and zkVM proof, a membership proof verification is also needed in the aggregation circuit. This ultimately produces an Aggregation Proof for on-chain verification.

5 Tokenomics

ZKL will be a standard ERC20 token issued on the Ethereum Mainnet. The total supply of ZKL is capped at 1 billion and is non-inflationary. ZKL will also be made available on the zkLink Nova network.

ZKL serves as the native utility token and governance token for the zkLink protocol.

Utility wise, ZKL helps developers unlock the access to the App-Rollup infra service of zkLink X and pay for the zero-knowledge proof computational resource.

ZKL is the governance token of zkLink DAO, which governs the development of the protocol. For zkLink Nova, in the future, ZKL will present the right to participate in Nova's decentralized sequencing network.

5.1 Payment

App Rollups served by the zkLink X solution will need to pay a license fee in ZKL to the zkLink DAO (the DAO). The DAO will provide support for startups with a certain amount of ZKL, for covering license fees in the beginning. In return, the App Rollup will share a portion of its future revenue or tokens with the DAO.

ZKL is the payment token for the zero-knowledge proof generation service provided by provers. Provers bid in zkLink's proof auction market and complete the proving tasks to earn ZKL tokens. Provers will need to stake a specific amount of ZKL to join the proof market. The provers that fail to complete a task or exceed the time requirement will be punished. This dynamic mechanism fosters competition and motivates provers to continuously improve their services and contribute to the entire zkLink ecosystem.

5.2 Governance

ZKL is the governance token of zkLink protocol. Holders will be able to stake ZKL to get veZKL based on staking amounts and the remaining lock-up period. veZKL holders will exercise governance rights over the direction and development of zkLink DAO.

Today, zkLink Nova's revenue comes directly from the centralized sequencer, accruing to the zkLink DAO vault. In the future, ZKL grants the right to participate in zkLink Nova's decentralized sequencing network.

5.3 Optional Gas Token of zkLink Nova Network

The native gas token of zkLink Nova is ETH. However, users will have the option to pay the gas cost in ZKL at a discount rate when conducting transactions on Nova.

6 Conclusion

This paper presents the conceptual design and system structure of zkLink protocol, a decentralized aggregated rollup infrastructure protocol based on zero knowledge technology.

zkLink addresses the problems of liquidity fragmentation, lack of interoperability, high trading cost, and complexity of multi-chain deployment via a novel aggregated rollup framework using zk-SNARKs technology with multi-chain state synchronization.

Based on zkLink aggregated rollup infrastructure, zkLink Nova is built to be the industry's first aggregated Layer 3 zkEVM Rollup network for general purpose dApps, enabling developers and users the access to the aggregated liquidity from the entire Ethereum ecosystem, including Ethereum, ZK-Rollups and Optimistic Rollups.

For customized high-performance applications requiring their own sovereignty, zkLink X serves as a middle-ware infrastructure that abstract away the complexity of multi-chain deployment, making developers feel like building on a single chain with multi-chain liquidity with SDK and APIs. Its modular architecture allows developers to customize the key components to meet diverse demands in different use cases. The TS-zkVM provides a high-efficiency execution environment specifically designed for high performance and low-cost financial products such as CLOB DEX, NFT trading etc., contributing to the mass adoption of self-custodial, user-friendly DeFi products.

Glossary

ZKP

Zero-knowledge Proof

L1

A layer 1 blockchain such as Ethereum, Solana, Avalanche, BNB Chain, etc.

L2

A Ethereum layer 2 rollup network such as zkSync, Starknet, Arbitrum, Optimism, etc.

L3

A rollup network built upon L2.

TS-zkVM

A trading-specific execution runtime based on tailored low level core ZK circuits and open-source zkVM extension circuits, developed by zkLink.

zkLink Nexus

A multi-rollup Layer3 ZK-Rollup architecture proposed by zkLink, which connects Ethereum L2s.

zkLink Origin

A multi-chain ZK-Rollup architecture proposed by zkLink, which connects various L1s and L2s.

zkLink App Rollup

An application-specific rollup network based on the rollup infra solution provided by zkLink, deployed by developers that need high throughput and low transaction cost, for example, order book exchanges.

References

1. https://docs.celestia.org/developers/build-modular
2. https://docs.espressosys.com/sequencer/espresso-sequencer-architecture/readme
3. https://docs.polygon.technology/cdk/
4. https://docs.zklink.io/
5. https://docs.zk.link/
6. https://era.zksync.io/zk-stack
7. https://layerzero.network/publications/LayerZero_Whitepaper_V2.0.pdf