Zulu Network: Empowering a Scalable and Open Future for Bitcoin

www.zulunetwork.io

Abstract. The Zulu Network represents a ground-breaking advancement in the Bitcoin ecosystem, introducing a two-tiered architecture to expand the capabilities of the Bitcoin Network. The initial and secondary layers are named ZuluPrime and ZuluNexus, respectively. ZuluPrime operates as a Bitcoin Layer 2, prioritizing EVM compatibility and delivering stable and sophisticated DeFi applications and Financial Services to the Bitcoin Network, while facilitating the use of assets such as BTC and BRC20 tokens on different dApps without the need to navigate to other ecosystems. On the other hand, ZuluNexus is positioned as the Layer 2 for ZuluPrime, essentially acting as a Bitcoin Layer 3. It employs a hybrid VM design that is not EVM compatible but supports both UTXO and Account types. This design preserves the capabilities of Bitcoin, while adding extra features, enabling developers to explore new innovative possibilities. ZuluPrime and ZuluNexus are interconnected through a seamless bridge, facilitating swift asset transfers between them. The validity of Zulu's state transitions is verified directly on the Bitcoin Network through a BitVM bridge, introducing Turing-complete programmability to Bitcoin. Zulu also uses a ZK bridge to connect other L1s to ZuluPrime. This innovative approach enhances the overall functionality and compatibility of the Zulu Network, marking a significant step forward in the realm of Decentralized Finance on the Bitcoin blockchain and beyond.

Keywords: Bitcoin, L2, EVM, ZK, BitVM.

1. Introduction

On October 9th, 2023, Bitcoin Researcher and expert in ZK technology Robin Linus released the BitVM whitepaper, sparking significant discussions within the Bitcoin community. While the technical intricacies of BitVM will be introduced in later chapters, its overarching capability can be summarized as the ability to compute anything on the Bitcoin network.

It is noteworthy that, prior to this revelation, several attempts had been made to construct Layer 2 solutions on Bitcoin. However, due to the inherent programmability limitations of Bitcoin itself, these endeavours, including RGB, Stacks, Rootstock, and other Bitcoin Layer 2 projects, encountered challenges in achieving successful implementations.

Although the BitVM whitepaper theoretically demonstrates the feasibility of a BTC L2, there are many areas to optimize, such as the bisection protocol and scripting language. Detailed insights into the specific design and developmental progress can be obtained on Github.

Regardless, BitVM has empowered visionary developers by showcasing the potential for constructing authentic Layer 2 solutions on Bitcoin. Subsequently, numerous Bitcoin Layer 2 projects emerged, with some leveraging Bitcoin as a Finalization Layer based on BitVM, while others utilized Bitcoin as a Data Availability (DA) Layer. A common thread among these projects is the incorporation of the Ethereum Virtual Machine (EVM) as the execution layer for their respective Layer 2 solutions.

The Ethereum Virtual Machine (EVM) offers many advantages when used as secondary execution layer for Bitcoin. It implies that assets on the Bitcoin Network can be employed for applications akin to those within the Ethereum ecosystem. Furthermore, Ethereum's technology stack has undergone extensive testing, ensuring high stability and security. In the context of assets, security stands out as a crucial attribute. However, the narrative logic of this approach has limitations, as users have the option to directly transfer assets from Bitcoin to the Ethereum ecosystem to engage with various applications. Presently, a ZK cross-chain bridge facilitates the connection between the Bitcoin and Ethereum networks.

While BitVM has expanded the narratives around the Bitcoin ecosystem, it's important to note that the blockchain industry demands innovation, and not mere replication. Transplanting the EVM to the second layer of Bitcoin may not be enough to open up the Bitcoin Network, as it is deterministic and has already been demonstrated on Ethereum. The emergence of the Ordinal protocol signals a trend of actively seeking innovation on Bitcoin.

As a Layer 2 solution on Bitcoin, Zulu needs to facilitate the swift integration of new assets and more mature DeFi applications into the Bitcoin Network, through the ZuluPrime platform. Simultaneously, we must create a platform that fosters developer innovation, preserving the UTXO model of Bitcoin and expanding its capabilities to write smart contracts. This expansion should be compatible with the Account type (addresses used on Ethereum), through the ZuluNexus platform.

Both UTXO (used on BTC) and Account (used on ETH) models have distinct advantages and disadvantages. Combining these two types opens significant innovation possibilities for developers. Taking the Ordinals protocol as an illustration, Indexer rules can be directly articulated in smart contract language, resulting in open and transparent regulations. Initiatives like the Aleo project (public chain), the Aztec project (Ethereum L2), and the Ola (Ethereum L2) represent endeavours to blend UTXO and Account types, creating a hybrid asset platform. These innovations will introduce novel use cases in the industry, combining Bitcoin's battle-tested accounts with more user-friendly Ethereum-inspired accounts.

The following is a summary of the technical characteristics of ZuluPrime and ZuluNexus, highlighting the potential of Zulu, which will be elaborated on in subsequent chapters:

ZuluPrime Features

1. EVM Compatibility: Implementing an EVM-compatible Layer 2 based on zkSync's ZK stack.

2. ZK-Based: Generating a stark proof for each block.

3. Verify on L2: Establishing a verifying network in the Layer 2 to quickly validate proofs before submitting it for verification on Bitcoin, prioritizing security and cost efficiency.

4. Verify on Bitcoin: Employ BitVM technology to verify Layer 2 proofs on the Bitcoin network.

5. Store on DA Service: Place transaction data on a Data Availability (DA) service, such as Celestia, Eigenlayer, and other DA services.

6. Trust-Minimized Bridge: Ensure secure asset cross-chain transactions between ZuluPrime and Bitcoin, implementing BitVM and 2-way peg technology.

ZuluNexus Features

1. Hybrid Type: Support UTXO and Account asset types.

2. Programmable: Support smart contracts.

3. ZK-Based: Generating a stark proof for each block.

4. Verify on ZuluPrime.

5. Store on DA Service: Place transaction data on a Data Availability (DA) service, such as Celestia, Eigenlayer, and other DA services.

6. Decentralized Bridge: Ensure secure asset cross-chain transactions between ZuluPrime and ZuluNexus.

2. Zulu Network

Zulu Network ("Zulu") stands out as the first Bitcoin Layer 2 (L2) platform to introduce a dual-layer architecture, meticulously considering both stability and innovation. Relying solely on EVM compatibility poses challenges, such as limited ecosystem innovation and application scenarios. On the other hand, exclusively retaining UTXO characteristics carries inherent risk issues. To address these concerns and facilitate the use of fast DeFi protocols on the Bitcoin Network, Zulu has ingeniously proposed a dual-layer architecture design.

This exciting approach aims to support developers in exploring innovative applications based on both the UTXO and Account models. Additionally, it seeks to sustain the growth of new protocols like BRC20 and ARC20, ensuring the continued prosperity of the Bitcoin ecosystem. The dual-layer architecture design is visually represented below:

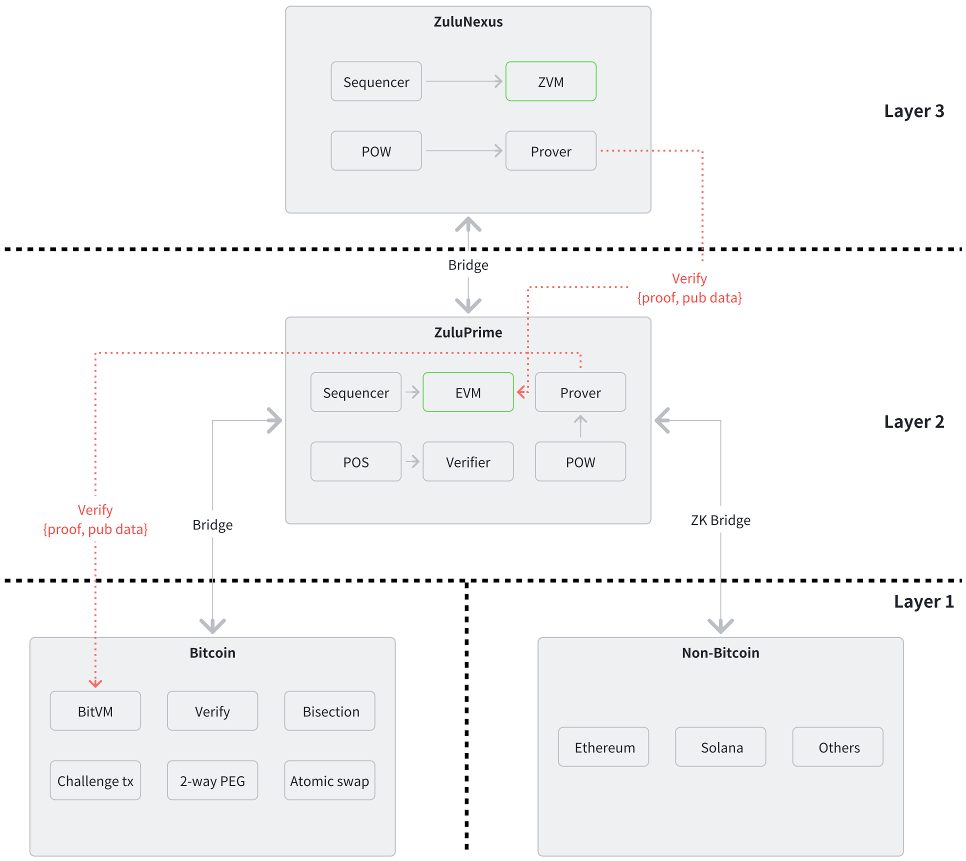

Figure 1: Zulu Architecture

Layer 1:

As an L2 network implemented on the Bitcoin blockchain, Bitcoin assumes the role of a Settlement Layer. However, due to its inherent programmability limitations, running a ZKP verification program directly on Bitcoin is impractical. Nevertheless, leveraging BitVM technology enables the implementation of a trust-minimized solution, facilitating the verification of ZKP on the Bitcoin network. In fact, within the framework of trust minimization, any program executed on BitVM can undergo verification on the Bitcoin blockchain.

Simultaneously, all native assets are initially issued on the base layer. All those assets can be seamlessly moved to Zulu through different bridges. Zulu can integrate with existing bridges, such as Orbiter or Polyheldra, enabling the trustless transfer of assets from non-Bitcoin chains to Zulu. However, achieving trustless cross-chain bridges for assets on Bitcoin is challenging due to the absence of support for smart contracts on the Bitcoin blockchain. Nevertheless, trust-minimized bridges can still be realized through the BitVM + 2-way peg mechanism.

Layer 2:

The emergence of the Ordinals protocol has enabled the issuance of new tokens on Bitcoin, adhering to the BRC20 token standard. However, these tokens, constrained by their current capabilities, are limited to transfer operations. Consequently, there is an urgent need to integrate DeFi applications into Bitcoin's L2, allowing these Bitcoin-based assets to leverage decentralized applications (dApps). Given the close association of these applications with user assets, prioritizing the security and reliability of these dApps is crucial to minimizing potential asset losses for users.

In response to this imperative, Zulu will introduce an Ethereum Virtual Machine (EVM) into his Layer 2, referred to as ZuluPrime. The ZuluPrime layer will be constructed on the ZK stack of zkSync, a well-established, high-performance, and EVM-compatible Zero-Knowledge EVM (ZKEVM) solution. This EVM compatibility facilitates the seamless migration and deployment of the Ethereum ecosystem onto ZuluPrime. Once Bitcoin assets are bridged to ZuluPrime, users can fully access and benefit from these mature dApps.

Acknowledging the computational demands for Zero-Knowledge (ZK) proof operations on ZuluPrime, a Proof-of-Work (PoW) mechanism has been introduced to ensure ample computing power on the network. Unlike traditional PoW, ZuluPrime's PoW primarily focuses on calculations related to the core operators of the ZK algorithm. Considering the potential high costs associated with submitting proof for each block to the Bitcoin network, a certain number of proofs will be aggregated under normal circumstances to reduce the frequency of verification submissions to the Bitcoin Network.

However, there is a challenge when the validity of the proof has not been verified before submission to the Bitcoin Network, and the state on ZuluPrime remains unfinalized. To address this, ZuluPrime will establish a Proof-of-Stake (PoS)-based verification network to promptly verify the proof for each block. This verification occurs before the proofs are submitted to Bitcoin, instilling trust in the PoS-based verification network.

Layer 3:

As discussed in the preceding chapter, Zulu holds the belief that merely supporting the Ethereum Virtual Machine (EVM) onto the Bitcoin ecosystem does not constitute the entirety of the ecosystem. This is because users have the straightforward option of transferring assets from Bitcoin to Ethereum, achieving a comparable outcome. We have identified the Ordinals protocol and Atomical protocol as noteworthy and innovative initiatives within the native asset realm of the Bitcoin Network. As the designated Layer 2 (L2) solution for Bitcoin, it is crucial for Zulu to safeguard this fertile ground for innovation and create a platform that is more user-friendly for developers.

ZuluNexus has been conceived for precisely this purpose. Positioned on top of ZuluPrime, it functions as both the Layer 2 (L2) of ZuluPrime and Layer 3 (L3) of Bitcoin. The strategic advantage of this arrangement is that, as an innovative Layer, ZuluNexus exclusively interacts with ZuluPrime. The innovation within ZuluNexus is rooted in its Virtual Machine (VM) design. In contrast to ZuluPrime's EVM, the VM implemented in ZuluNexus is an entirely new construct, unifying BTC and ETH asset types (UTXO & Account). Simultaneously, it extends the capabilities of the UTXO type, enabling the authoring of smart contracts based on the UTXO model. Zulu incorporates insights from the designs of Aleo, Aztec, and Ola in this realm, building upon and expanding their foundations.

2.1 ZuluPrime (L2)

ZuluPrime is positioned as the Layer 2 (L2) solution on the Bitcoin network, presenting itself as a Zero-Knowledge EVM (ZKEVM) solution constructed on the ZK stack of zkSync.

Transactions conducted on ZuluPrime undergo bundling into blocks before execution. The Sequencer plays a crucial role in providing the world state both before and after the execution of all transactions within the current block. In the event of doubt, any individual can act as a Challenger, challenging the execution of the current block. Theoretically, these transactions require re-execution on Layer 1 (L1) to ensure confirmation at the same security level as Layer 1. However, due to Bitcoin's inability to execute smart contracts (which may originate from various smart contract calls), ensuring L2 security becomes challenging.

BitVM introduces programmability to the Bitcoin network, operating on the premise of minimum trust (1 of N level, ensuring a sufficiently large N). This, to some extent, addresses the problem described above of replaying transactions on Bitcoin. However, upon meticulous evaluation, certain challenges persist; notably, the necessity to deploy all smart contracts on BitVM. From the standpoint of verifying the validity of transaction execution, this presents an inherently inefficient solution in terms of resources, scalability, and data privacy.

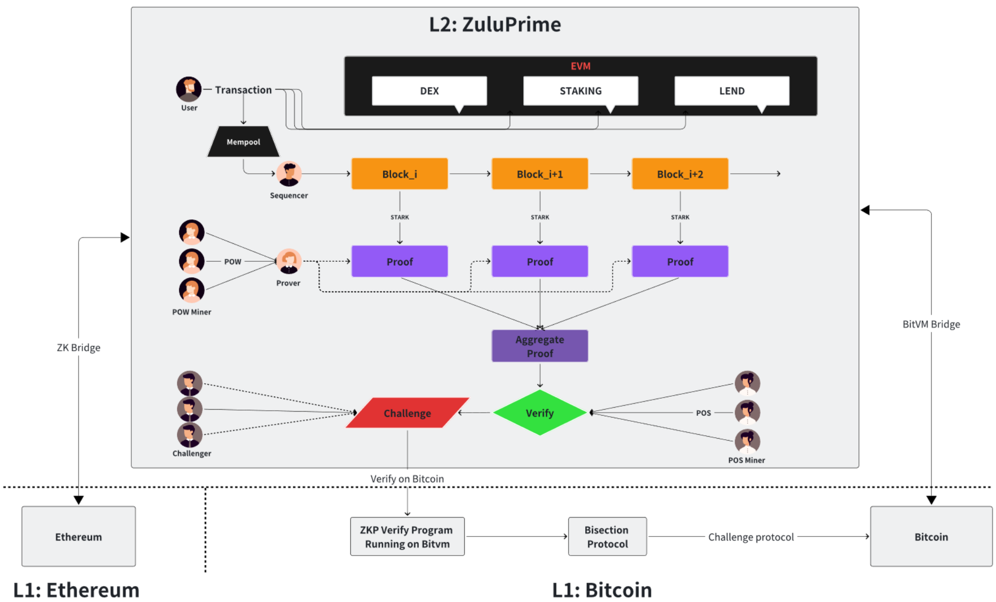

Fig 2. ZuluPrime Architecture

To navigate past those challenges, we introduce the concept of Zero-Knowledge Proof (ZK). Essentially, a proof is generated for the execution of all transactions, and successful verification indicates the validity of the transaction execution. This method, from a verification standpoint, mitigates the impact of higher-level transaction types, as the ZKP verification program consistently runs on BitVM, irrespective of the transaction logic. The only variation between different blocks lies in the input of the program.

To maintain uniformity in the execution environment, the verification network bases the execution environment of the ZKP verification program on BitVM. Subsequently, Challengers have the ability to challenge the results of its execution.

ZuluPrime has a fast block generation speed, so it needs substantial computing power from different provers to generate proofs for blocks. Simultaneously, ordinary user engagement verify networks are also essential on the network. Given the low computing power requirements of the ZKP verification program, these networks facilitate swift verification of proofs, thereby ensuring system security. ZuluPrime's overarching objective is to enable users to access the verify network on their mobile phones, lowering the barrier for user participation.

2.2 ZuluNexus (L3)

ZuluNexus is designated as Layer 3 (L3) within the Bitcoin ecosystem and will be directly deployed on ZuluPrime, interacting seamlessly with it. However, the verification of its state changes will involve both the Verify network and the Bitcoin network, in conjunction with the proof provided by ZuluPrime. The architectural diagram of ZuluNexus is depicted below:

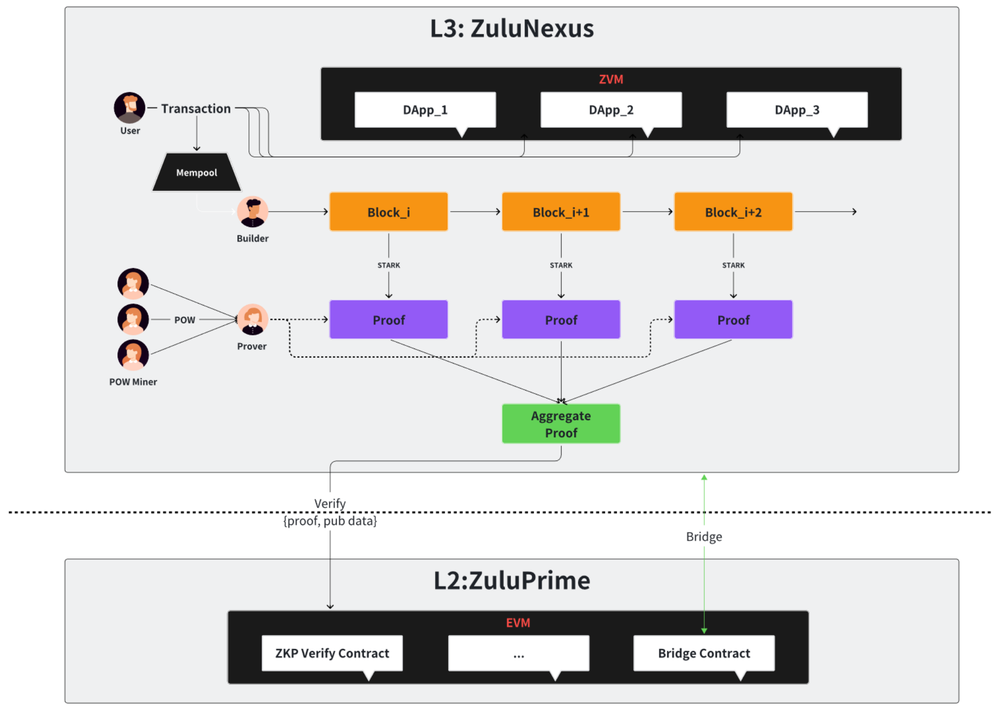

Figure 3. ZuluNexus Architecture

Since the state transition of ZuluNexus will be verified on ZuluPrime, there is no necessity to introduce an additional verification network on ZuluNexus. Furthermore, the interval between proof submissions on ZuluNexus and ZuluPrime can be brief, thanks to lower average transaction costs. As illustrated in Figure 3, the primary distinction between ZuluNexus and ZuluPrime stems from the design of their Virtual Machines (VMs).

ZuluNexus is constructed on a novel VM that differs from the Ethereum Virtual Machine (EVM). This VM maintains two distinct world state trees—one for the Account type and the other for the UTXO type state. In practice, a UTXO commitment tree may also be upheld. This approach is informed by the experience of recent years, where the Account type has demonstrated greater compatibility with smart contracts, evident in the success of Ethereum. Simultaneously, the UTXO type is better suited for private transactions, given that transactions based on the UTXO type are executed locally by users and subsequently verified by network nodes. Many privacy-focused projects, including Zcash, Aleo, Aztec, Ola, etc., are implemented based on the UTXO type.

Consequently, both UTXO and Account types boast distinct advantages in specific scenarios. The goal of ZuluNexus is to combine these two types, creating a programmable platform based on mixed account types. This framework enables developers to continually expand upon more innovative functionalities based on the foundation of UTXO types.

3. Asset Bridging From Bitcoin

As Bitcoin's Layer 2 (L2) solution, the Zulu system must facilitate the seamless transfer of assets between Bitcoin and Zulu. Presently, all well-conceived cross-chain bridge designs incorporate the BitVM framework, exemplified by Robin's BitVM Bridge and ZK Base's ZKByte design. Zulu's cross-chain bridge for Bitcoin will be implemented utilizing the BitVM bridge, ensuring a trust-minimized security level.

The BitVM bridge can maintain a 1:1 ratio between the value of zBTC published on Zulu and the value of BTC locked on Bitcoin through a 2-way PEG mechanism. However, a challenge arises when crossing zBTC based on the BitVM bridge—its value remains fixed, equivalent to the value of BTC locked previously. Ordinarily, due to certain transaction behaviors, assets crossed back by regular users may not have a fixed value. To address this scenario, the system supports ordinary users in transferring assets between Bitcoin and Zulu through the mechanism of Atomic swaps. ZBTC crossed via the BitVM bridge can function as a liquidity provider for Atomic swaps in such situations.

3.1 Peg IN/OUT BTC on Zulu

Pegging assets in and out of Zulu involves locking a specific amount of BTC assets on the Bitcoin network and subsequently issuing wrapped BTC and zBTC on the Zulu platform. This process is subject to several requirements:

1. Locked BTC assets must remain non-transferable. There are centralization risks associated with both single and multiple signing methods.
2. Zulu must incorporate a Bitcoin SPV Client to verify the confirmation of specific transactions on the Bitcoin network.

3. Zulu's SPV Client must recognize the existence of Bitcoin and verify the occurrence of specific transactions on the Zulu platform. This functionality will be implemented based on BitVM.

The complete process of Pegging IN/OUT is visually illustrated below:

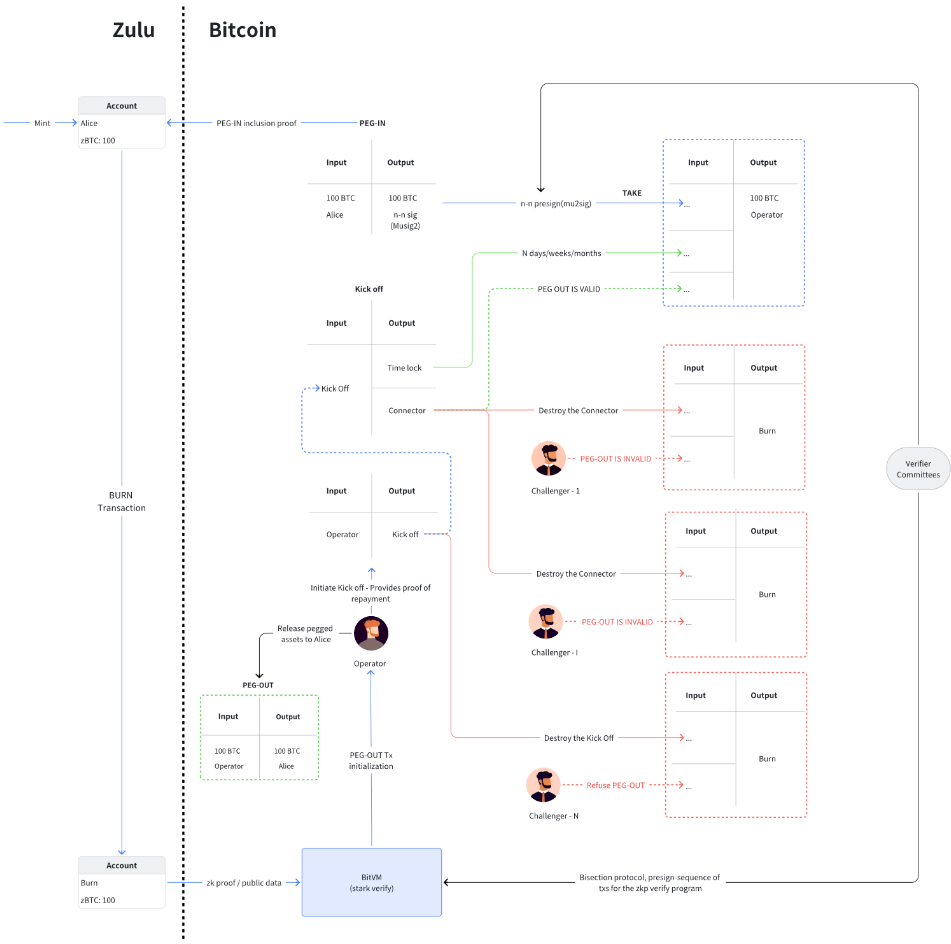

Fig 4. Peg IN/OUT Process

1. Alice initiates a Peg-In process with 100 BTC while simultaneously endorsing an n-n transaction, utilizing the Mulsig2 signature algorithm, where n represents the number of challengers in the network.
2. The signed transaction outlines the inputs and outputs, ensuring that the Peg-In output UTXO can no longer be spent in any other form.
3. At this point, Alice receives 100 zBTC on Zulu as a result of the Peg-In.
4. When Alice decides to Peg-Out, she initializes a Burn transaction on Zulu.
5. Validate the legitimacy of the Burn transaction on BitVM. During this phase, the Operator transfers 100 BTC from their account to Alice. The Operator can be any one of the n participants.
6. Upon completion of the transfer, the Operator initiates a kick-off transaction, signifying the conclusion of the proxy expenditure. At this point, the Operator aims

to trigger the PEG-OUT transaction, facilitating the transfer of their Peg-In BTC to Alice.

7. This request undergoes a challenge period of 6 months. If any verifier detects any irregularities in the Operator's actions, they can consume the connector output. This action prevents the Operator from initiating the PEG-OUT transaction the regular way.

3.2 Atomic Swap

As mentioned earlier, PEG IN/OUT operations are constrained to assets of the same value. However, there are instances where users may wish to engage in transactions involving more or fewer assets, making PEG IN/OUT unsuitable for such scenarios. Atomic swaps offer a solution to this problem. The fundamental concept involves users transferring BTC assets to liquidity providers, who, in turn, transfer an equivalent value of zBTC to users on Zulu. There are several requirements.

1. Ensure the atomicity of transactions, guaranteeing that transactions either succeed or fail simultaneously.
2. Implement an SPV Client to provide proof of transaction occurrence.
3. Establish an Incentive Mechanism and penalty mechanism to safeguard the earnings of liquidity providers and deter any potential misconduct.

The flowchart illustrating the entire Atomic Swap process is depicted below:

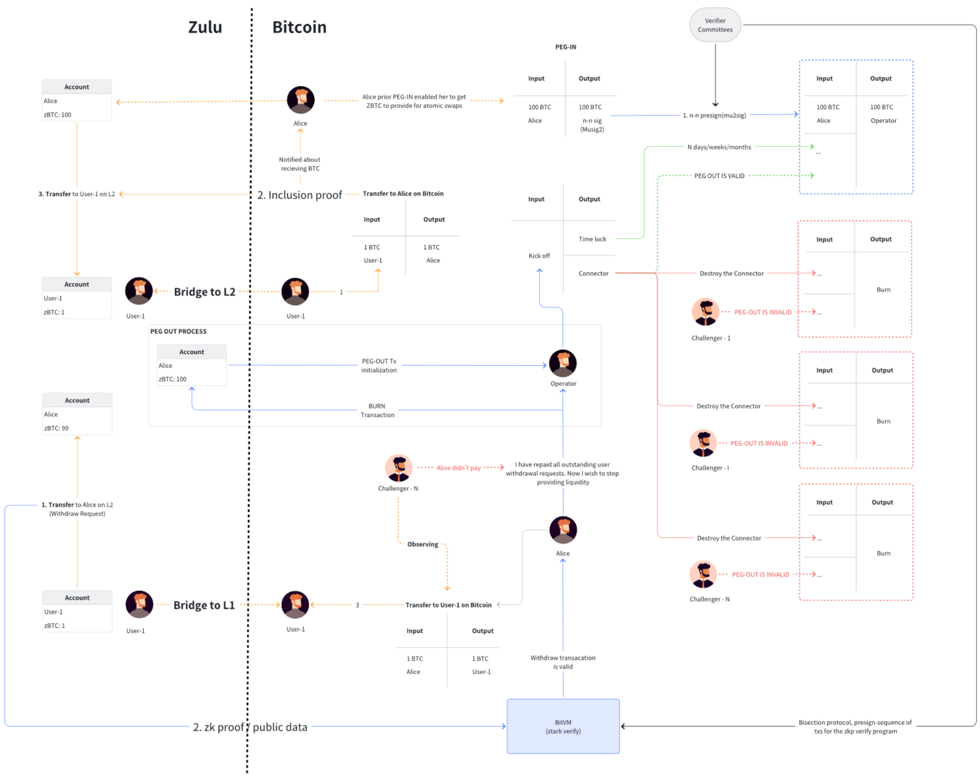

Fig 5. Atomic Swap Process

1. Alice acquired zBTC on Zulu through PEG IN/OUT and subsequently staked 100 BTC on Bitcoin, transitioning Alice's role into being a liquidity provider.
2. Regular users aiming to switch between Bitcoin and Zulu or vice versa are required to furnish an SPV proof, validating that the trading user has transferred the corresponding assets to Alice.
3. Alice reviews and verifies the transaction, proceeding to transfer the assets from her account to the user.
4. When Alice decides to withdraw as a liquidity provider, she must substantiate her honesty, demonstrating that she has not engaged in any malicious activities. Subsequently, she attempts to trigger the PEG OUT transaction to reclaim the pledged BTC.
5. The PEG OUT transaction is subject to a specific challenge time window. If a verifier identifies any malicious behavior by the Operator, they can consume the connector output, preventing Alice from retrieving the pledged assets.
6. In the role of a liquidity provider, Zulu will offer incentives in the form of native tokens.

4. Verify on Bitcoin

Regardless of the bridge or L2 design, the use of BitVM is imperative for verifying certain state proofs. Consequently, the crux of the verification process on Bitcoin lies in the BitVM design. BitVM is envisioned as a Turing-complete virtual machine equipped with its own collection of opcodes, facilitating the composition of various program logics. In crafting the Instruction Set Architecture, the design adheres to the following key characteristics:

1. Support for Turing completeness: This necessitates the ability to handle arbitrary computations.
2. Simplified Instruction Set Architecture: Custom BitVM opcodes must be implemented using the opcodes currently supported by Bitcoin.

In the design of BitVM, Zulu will uphold these principles, drawing inspiration from similar designs found in projects like Starkware's Cairo VM and Ola's OlaVM. The Zulu team aims to build upon these foundations. The introduction of the Bisection mechanism allows both provers and challengers to contest the execution trajectory based on BitVM's Instruction Set Architecture. It involves identifying disputed opcodes and subsequently verifying the corresponding input, output, and current instructions on Bitcoin.

Concerning the challenge mechanism's design, following a concept similar to Robin's, Zulu will challenge all modules of BitVM. The figure below illustrates this design concept from Robin's framework (https://github.com/BitVM/BitVM/blob/main/docs/bitVM_graph_v2.svg).

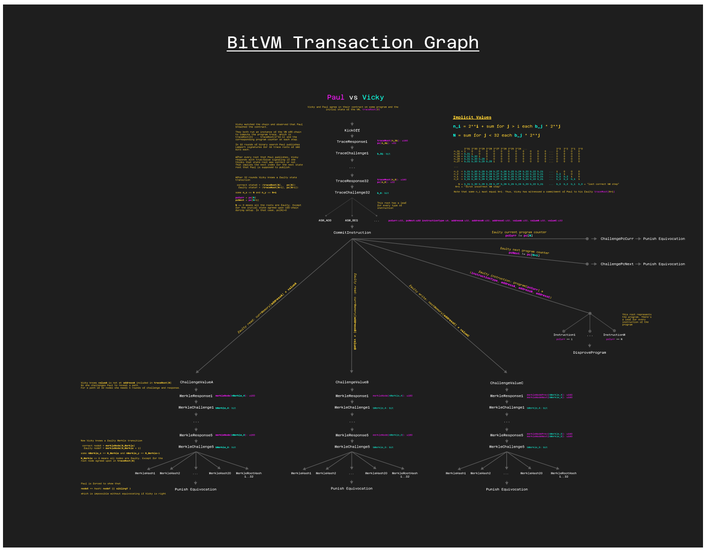

Fig 6. BitVM Graph

As depicted in the illustration:

1. Paul and Vicky jointly undertake the challenge of a program. In Zulu, this challenging program involves ZKP verification, utilizing the STARK algorithm for the ZKP algorithm.
2. Both Paul and Vicky individually execute this program within the BitVM environment, meticulously recording a sequence of changes in the BitVM root.
3. Commencing from the initial root, with each execution, a new root is derived until all instructions are executed.
4. Given Vicky's identification of Paul's misconduct, there is a high likelihood that the root values obtained by Paul and Vicky will differ.
5. Through a continuous binary search approach, the goal is to persistently identify the instruction corresponding to the differing roots.
6. Since BitVM instructions encompass extensive information, including the instruction type, operands, and contextual information both upstream and downstream, it is imperative to challenge each of these aspects separately.

5. Data Availability

As an L2 network, Zulu must provide users with the means to securely retrieve their assets from L1 by validating their assets through historical transactions, even in instances where L2 services are inaccessible. This precautionary measure is in place to prevent the locking of users' assets in L2 during periods when L2 services are unavailable. The optimal solution involves storing all transactions on the Layer1 verification network. However, drawing from the developmental history of Ethereum, we learn that while placing transaction data on the Layer1 network achieves the highest level of security, an excess of transaction data can overwhelm the network. Consequently, Ethereum itself addressed this Data Availability (DA) issue through the upgrade of EIP4844.

In the context of the Bitcoin network, storing all transactions directly on Bitcoin is not considered an ideal approach. Zulu's strategy involves placing actual transaction data on networks specifically dedicated to DA, such as Celestia, EigenLayer, among others.

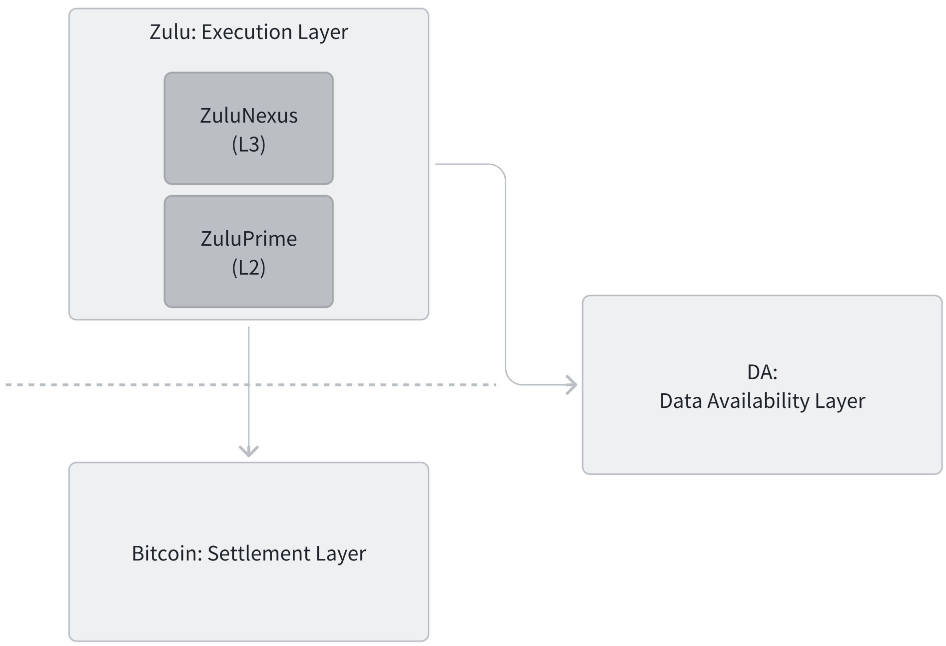

Fig 7. Zulu Modular Design

Zulu follows a modular design, a technical direction widely embraced in the blockchain industry that plays a crucial role in enhancing the operational efficiency of the system.

6. Acknowledgment

Zulu is building a genuine ecosystem within the Bitcoin industry, allowing existing Bitcoin assets to engage with sophisticated DeFi applications and Financial Services, while preserving the original security, reliability, and capabilities of the Bitcoin Network.

The Zulu Network provides developers with a space for ongoing and sustained innovation. Zulu does not favour any specific technical direction, but rather is committed to integrating the strengths of various approaches to foster true prosperity within the Bitcoin industry.

The project expresses sincere gratitude to those who have made significant contributions towards the future of Bitcoin. Zulu would not be able to build such complex and sophisticated solutions without the help of the technology pioneers that came before us. We want to acknowledge the zkSync team, Aleo team, Ola team, and individuals like Robin Linus for their open-source spirit. This spirit represents the core ethos that propels the continual development of the Bitcoin community and the broader blockchain industry.

Without such dedication, Zulu would not have a base to build upon. The project pledges to uphold this spirit, pressing forward, leveraging the knowledge of industry pioneers, and contributing meaningfully to the Bitcoin ecosystem, the Zero Knowledge field, and Blockchain technology.

References

[1] BitVM whitepaper: https://bitvm.org/bitvm.pdf

[2] Aleo whitepaper: https://eprint.iacr.org/2018/962.pdf

[3] Cairo VM whitepaper: https://eprint.iacr.org/2021/1063

[4] OlaVM whitepaper: https://github.com/Sin7Y/olavm-whitepaper-v2/tree/master

[5] ZK Stack of ZKSync: https://docs.zksync.io/zk-stack

[6] BitVM Github：https://github.com/BitVM/BitVM/tree/main/docs

[7] ZKByte：https://medium.com/zkswap/brc-layer2-design-43dfe54b9448

[8] EIP4844: https://eips.ethereum.org/EIPS/eip-4844

[9] ethStark: https://eprint.iacr.org/2021/582.pdf

[10] plonky2: https://github.com/0xPolygonZero/plonky2/blob/main/plonky2/plonky2.pdf

[11] Connect output example: https://github.com/supertestnet/loan-shark/