USPTO Provisional Patent Application

TITLE OF THE INVENTION

Canister-Based CPU Architecture for Deterministic Distributed Computation on Blockchain Networks

FIELD OF THE INVENTION

The invention relates to computer architecture systems, distributed computing, and blockchain-based execution substrates. More particularly, the invention describes a new type of processor architecture in which isolated blockchain execution units (“canisters”) collectively implement the functional components of a central processing unit (CPU), including an arithmetic logic unit (ALU), registers, memory, control logic, and instruction dispatch mechanisms.

BACKGROUND OF THE INVENTION

Technical problem:

Modern processors rely on tightly coupled transistor circuits operating within a single physical die. Distributed systems exist, but none replicate the core abstractions of a CPU — ALU, bus, registers, control unit — using independent, mutually isolated computation units running in a deterministic virtual machine on a blockchain.

Prior Art:

A review of existing technologies reveals no CPU implemented using blockchain canisters, no ALU built from deterministic isolated runtimes, no distributed register system mapped to stable blockchain memory, and no instruction pipeline using chain-level message routing.

SUMMARY OF THE INVENTION

This invention introduces a Canister-Based CPU Architecture, a novel computational model where each fundamental CPU component is implemented using blockchain execution units known as canisters. These units communicate exclusively through deterministic blockchain message routing.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 — Overall Architecture

FIG. 2 — ALU Execution Flow

FIG. 3 — Register Memory Model

DETAILED DESCRIPTION OF THE INVENTION

1. Canister Execution Units

Each canister is an isolated deterministic virtual machine instance.

2. ALU Canisters

ALU logic is implemented as WebAssembly-based computational nodes.

3. Register Canisters

Registers stored in stable memory with load/store message interfaces.

4. Memory Canisters

Block-addressable and key-value storage structures implemented as separate canisters.

5. Control Unit Canister

Decodes instructions, dispatches operations, manages branching.

6. Message Routing as CPU Bus

Blockchain message routing acts as the interconnect.

7. Instruction Set Architecture (ISA)

ADD, LOAD, STORE, CMP, BRANCH instructions executed via inter-canister messaging.

8. Clocking and Determinism

Clock cycles defined by deterministic blockchain message rounds.

9. Scaling

Horizontal scaling achieved by adding ALU, register, and memory canisters.

EXAMPLE EMBODIMENT

Example: ADD Operation

1. CU decodes ADD R1, R2 -> R3

2. CU requests values

3. ALU computes

4. Result stored back in register

CLAIMS (optional)

1. A processor architecture implemented using blockchain canisters.

2. A method for executing processor instructions by routing messages between canisters.

3. A distributed ALU composed of blockchain canisters.