



School: Campus:
Academic Year: Subject Name: Subject Code:
Semester: Program: Branch: Specialization:
Date:

Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment : Dive into Ethereum – Clients and EVM

* Coding Phase: Pseudo Code / Flow Chart / Algorithm

Introduction:

Ethereum Clients

- Software that allows nodes to connect, validate, and interact with the Ethereum network.
- Multiple clients exist (e.g., Geth, Nethermind, Besu, Erigon) — all implement Ethereum’s protocol.
- Ensure decentralization and resilience by avoiding single-client dependency.
- Handle networking, consensus, and execution of transactions.

Ethereum Virtual Machine (EVM)

- The **execution environment** for Ethereum smart contracts.
- Acts as a global, decentralized “computer” that runs untrusted code securely.
- Executes bytecode compiled from smart contract languages (e.g., Solidity, Vyper).
- Maintains deterministic execution — same input always gives the same output across all nodes.
- Provides isolation, so buggy/malicious contracts can’t affect the broader network.

* Softwares used

- 1.Brave browser
- 2.MetaMask Wallet
- 3.Remix IDE
- 4.Sepolia Testnet

* Implementation Phase: Final Output (no error)

This Solidity program is a **simple storage smart contract** that lets users save a number on the blockchain and retrieve it later.

It contains one state variable (storedNumber), a setter function (setNumber) to update the value, and a getter function (getNumber) to read the value.

```

1  // SPDX-License-Identifier: MIT
2  pragma solidity ^0.8.0;
3
4  contract SimpleStorage {
5      uint public storedNumber;
6
7      // Function to set a number
8      function setNumber(uint _num) public { 22514 gas
9          storedNumber = _num;
10     }
11
12     // Function to get the stored number
13     function getNumber() public view returns (uint) { 2453 gas
14         return storedNumber;
15     }
16 }

```

[view on Etherscan](#) [view on Blockscout](#)

✓ [block:9276369 txIndex:14] from: 0x5b3...5c960 to: SimpleStorage.(constructor)
value: 0 wei data: 0x608...e0033 logs: 0 hash: 0x59c...e136a

Debug

DEPLOY & RUN TRANSACTIONS

ENVIRONMENT
Injected Provider - MetaMask

Sepolia (11155111) network

ACCOUNT +
0x5b3...5c960 (0.170988691...)

+ Create Smart Account

GAS LIMIT
Estimated Gas

Custom 3000000

VALUE
0 Wei

CONTRACT
SimpleStorage - simple.sol

view source: pragma

Deploy

Publish to IPFS

At Address Load contract from Address

Transactions recorded 1

Deployed Contracts

SIMPLESTORAGE AT 0x608...11

Balances 9 ETH

withdrawNumber withdraw_num

getNumber

storedNumber

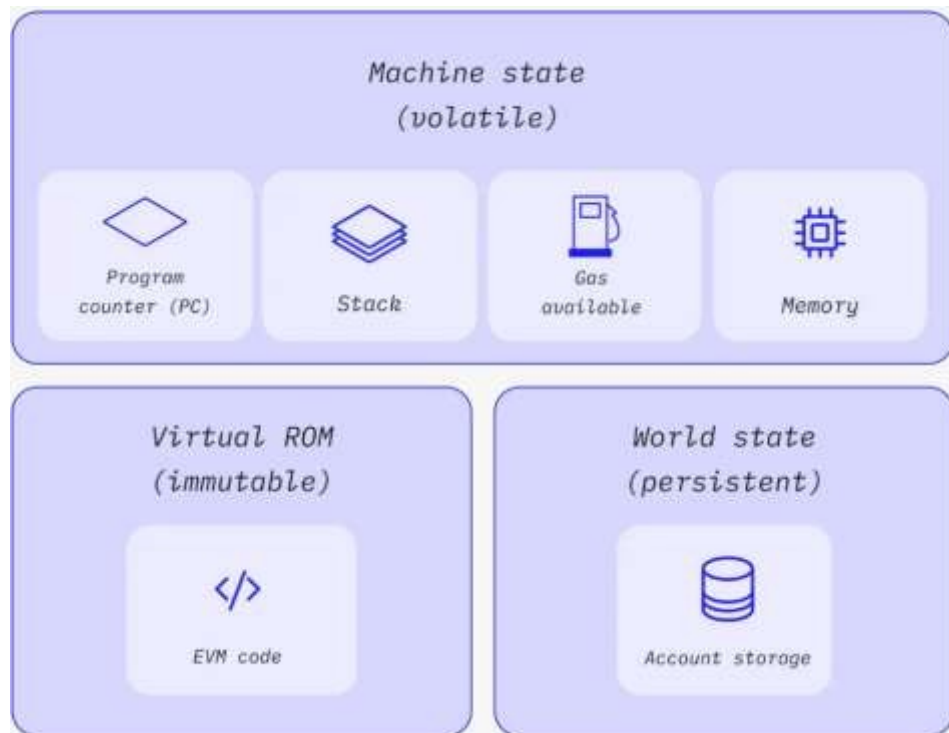
Low level interactions

CALLDATA

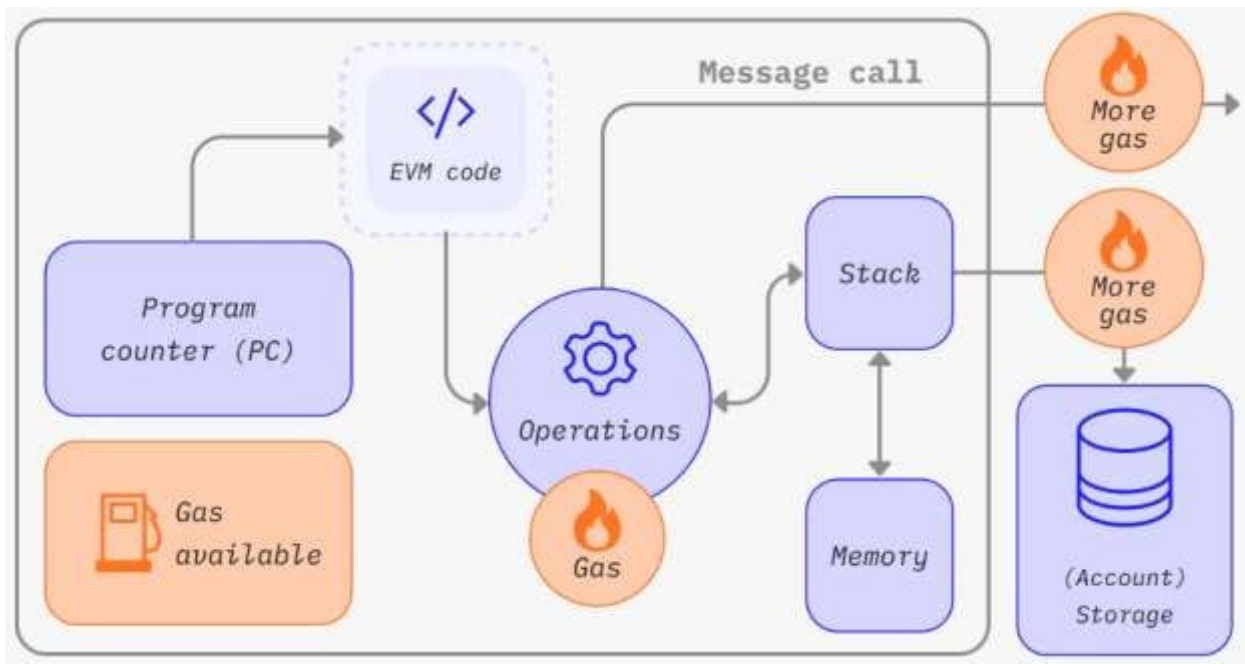
Transfer

* Implementation Phase: Final Output (no error)

Ethereum Virtual Machine (EVM)



EVM instructions



* Observations:

- The Ethereum client was installed and configured successfully.
- The client synchronized with the Ethereum blockchain (testnet/mainnet as configured).
- Account creation and wallet address generation were successful.
- The client responded correctly to JSON-RPC/CLI commands.
- Transactions were executed and verified without errors.
- Logs/output confirmed the proper working of the Ethereum client.

The experiment was successfully carried out. The Ethereum client was installed, configured, and executed without errors. It synchronized with the blockchain, allowed account creation, and responded to commands correctly. This confirms that Ethereum clients are essential for interacting with the Ethereum network, validating transactions, and executing smart contracts through the EVM.

* Conclusion

The experiment was successfully carried out. The Ethereum client was installed, configured, and executed without errors. It synchronized with the blockchain, allowed account creation, and responded to commands correctly. This confirms that Ethereum clients are essential for interacting with the Ethereum network, validating transactions, and executing smart contracts through the EVM.

ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10		
Planning and Execution/ Practical Simulation/ Programming	10		
Result and Interpretation	10		
Record of Applied and Action Learning	10		
Viva	10		
Total	50		

Signature of the Student:

Name :

Regn. No. :

*As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.