НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені Ігоря СІКОРСЬКОГО»
Навчально-науковий фізико-технічний інститут
Кафедра математичних методів захисту інформації

Звіт до лабораторної №1

за темою:

Розгортання систем Ethereum та криптовалют

Оформлення звіту:
Юрчук Олексій, ФІ-52мн

17 лютого 2026 р.
м. Київ

ЗМІСТ

1 Невеличкий вступ

Технологія блокчейн кардинально змінила наше уявлення про децентралізовані обчислення, фінансові транзакції та бездовірчі системи (trustless systems). З моменту публікації ”Bitcoin whitepaper” [1] by Satoshi Nakamoto в 2008 році, з’явилося безліч платформ блокчейн, кожна з яких має свої архітектурні рішення, процедури розгортання та операційні характеристики.

У своїй лабораторній, я розглядав і порівнював, здебільшого зазначені в завданнях до лабораторної, особливості розгортання п’яти основних систем блокчейну/криптовалюти: **Ethereum** [2, 3], **Bitcoin** [1], **Dash** [4], **NEO** [5] та **Litecoin** [6]. Свій аналіз я спрямував більше на практичні аспекти реалізації, розгортання та конфігурації кожної системи, вивченні їх базової архітектури, механізмів консенсусу, мережевих рівнів та можливості обміну модулями між ними.

Поки в планах наступне: у розділі 2 описати теоретичні основи кожної системи, далі в розділі 3 доволі детально порівняти їх за різними параметрами, розглянути їх взаємозамінність. А далі у розділі 4 планую навести покрокові, як я робитиму, практичне розгортання Ethereum (певно найпростіше, що можна взяти) у середовищі на базі WSL за setup-ами від [7, 8]. І відповідно в кінці підбити підсумки по отриманих результатах.

2 Огляд систем блокчейн і криптовалют

2.1 Bitcoin

Bitcoin – це певно перша і найвідоміша криптовалюта, опублікована в 2008 році анонімним девелопером з псевдонімом Сатоші Накамото. Вона використовує модель **UTXO (Unspent Transaction Output)** для відстеження балансів і механізм консенсусу **Proof-of-Work**, заснований на алгоритмі хешування SHA-256.

Ключові особливості Bitcoin включають в себе:

- **Жодних смарт-контрактів** (в традиційному сенсі) – Bitcoin Script навмисно є non-Turing-complete.
- **Block time** – генерація нового блоку $\approx$ 10 хвилин, **block size** обмежений $\approx$ 1–4 MB (з SegWit).
- **Загальний обсяг** обмежений 21 мільйоном BTC.
- **Networking** – використосує особливий мережевий P2P protocol через TCP (port 8333), розповсюдження інформації на основі gossip-based (”пліткування з сусідами”).

2.2 Ethereum

Ethereum – відкрита розподілена обчислювальна платформа на основі блокчейну, яка підтримує функціональність смарт-контрактів. На відміну від Bitcoin, який був розроблений в першу чергу як цифрова валюта, Ethereum був задуманий як універсальний програмований блокчейн (!). Власна криптовалюта блокчейну Ethereum це **Ether (ETH)**.

Основні архітектурні особливості Ethereum включають:

- **Ethereum Virtual Machine (EVM)** – віртуальна машина, яка виконує смарт-контракти, з повною функціональністю Тюрінга (це такий а-ля емулятор для виконання програм за скінченний час і пам’ять);
- **State model** – модель на основі облікових записів діє як банківська книга, відстежуючи баланс і забезпечує можливість укладання складних смарт-контрактів. (в Bitcoin застосовується UTXO model, де рахунки ефективно управляються смарт-контрактами, і в якій забезпечується вищий рівень конфіденційності та здійснюється паралельна обробка даних);
- **Protocol of Consensus** – первісно застосовувався Proof-of-Work (Ethash), але у вересні 22 зробили трансфер на Proof-of-Stake (Casper/Beacon chain);
- **Smart Contracts** – пишуться мовами Solidity, Vyper та деякими іншими мовами і є EVM-compatible;
- **Gas system** – обчислювальні витрати вимірюються за допомогою так званих тарифів ”gas fees”.

Ethereum – це peer-to-peer надбудова над базою інтернет протоколів TCP/IP [9]. Кожен вузол запускає копію блокчейну та бере участь у валідації та розповсюдженні блоків. Офіційний клієнт **Geth** (Go-Ethereum) реалізує протокол RLC (Remote Procedure Call) Node Discovery Protocol, що базується на DHT (Distributed Hash Table) типу Kademlia для виявлення однорангових вузлів (peers) [9].

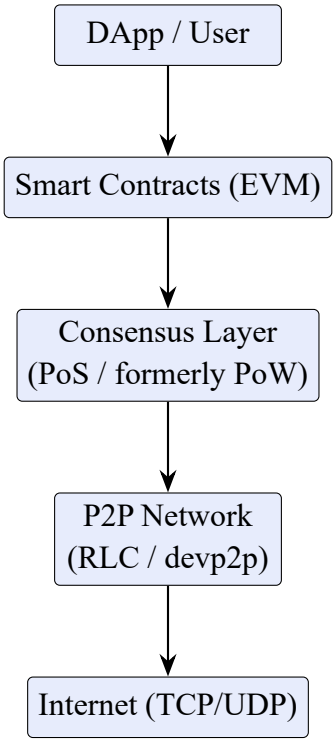


Рис. 1: Архітектура ”шарів” Ethereum-а

2.3 Dash

Dash (originally “Darkcoin”) являє собою fork від кодової бази Bitcoin, але з додатковими функціями конфіденційності та управління. В ньому було запроваджено:

- **Masternodes** – мережа другого рівня (second-tier network) з мотивованими (incentivized) вузлами, які забезпечують InstantSend і CoinJoin (PrivateSend).
- **Алгорит гешування X11** – ланцюжкова послідовність із 11 криптографічних геш-функцій.
- **Система управління фінансами (Treasury system)** – 10% винагороди за блок виділяється на фінансування ідей щодо розвитку (за них голосуються masternodes).
- **Block time** – генерація нового блоку складає ≈ 2.5 хвилин.

2.4 NEO

NEO (колишня назва – AntShares) китайська блокчейн-платформа, яку частенько називають ”китайським ефіром (Ethereum)”. Вона також підтримує виконання смарт-контрактів, але має значні відмінності по своїй архітектурі:

- **Consensus** – делегована візантійська відмовостійкість (delegated Byzantine Fault Tolerance - dBFT), що забезпечує детермінованість виконання.
- **Dual token model** – NEO (токен управління, є неподільним) and GAS (токен-утиліта для виконання контрактів).
- **Multi-language support** – смарт-контракти можуть бути написані на різних мовах програмування, таких як: C#, Python, Java, Go (з подальшим виконанням у віртуальній машині NeoVM).
- **Інтеграція цифрової ідентичності** та орієнтація на сурове дотримання нормативних вимог.

2.5 Litecoin

Litecoin був створений Чарлі Лі в 2011 році як "полегшена" версія Bitcoin. Це один з найперших форків від Bitcoin, який і досі має багато спільного з кодом Bitcoin. Основні особливості:

- **Scrypt hashing** – алгоритм, що вимагає великого обсягу пам'яті (первинно розроблявся для протидії майнінгу ASIC)
- **Block time** – генерування складає ≈ 2.5 хвилин (це у $4\times$ швидше за Bitcoin).
- **Total supply** – 84 мільйони LTC (у $4\times$ перевищує обсяг Bitcoin-а).
- **UTXO model** – така сама структура транзакцій, як у Bitcoin.
- **Test platform** – Часто стає тестовим майданчиком для функцій Bitcoin-а (наприклад, той же SegWit був активований спочатку на Litecoin).

3 Порівняльний аналіз архітектурних рішень та процедур розгортання

3.1 Механізми консенсусу

Механізм консенсусу є основним модулем, який визначає, як мережа блокчейну досягає згоди щодо стану реєстру. Кожен з названих нами п'яти блокчейнів використовують принципово різні підходи:

Блокчейн	Алгоритм гешування	Протокол	Остаточність
Ethereum	Gasper (PoS)	Proof-of-Stake	Probabilistic $\rightarrow$ deterministic (2 epochs)
Bitcoin	SHA-256	Proof-of-Work	Probabilistic ($\sim$6 confirmations blocks)
Dash	X11 + Masternodes	Hybrid: PoW + PoS	Instant (via ChainLocks)
NEO	dBFT 2.0	Delegated BFT	Deterministic (1 block, no forks)
Litecoin	Scrypt	Proof-of-Work	Probabilistic ($\sim$6 confirmations blocks)

Таблиця 1: Порівняння протоколів консенсусу

3.2 Архітектура мережі та P2P-взаємодія

Всі п'ять систем працюють як однорангові децентралізовані (peer-to-peer) мережі, побудовані поверх мережі Інтернет. Однак їхні протоколи виявлення однорангових вузлів і комунікації значно відрізняються.

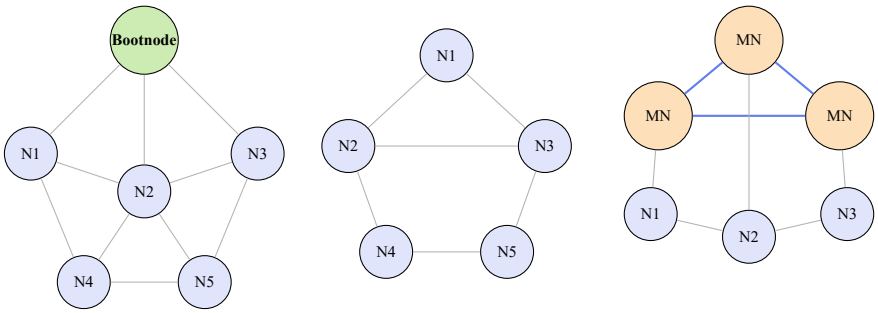
Ethereum використовує набір протоколів devp2p, який включає протокол виявлення вузлів RLPx на основі модифікованого DHT Kademlia. Кожен вузол ідентифікується за допомогою ідентифікатора `enode` (хеш SHA3 його відкритого ключа). Реалізація Kademlia використовує метрику відстані на основі XOR з 256 сегментами, кожен з яких містить до 16 записів. Виявлення однорангових вузлів (peers) використовує чотири типи повідомлень UDP: `ping`, `pong`, `findnode` та `neighbors`. Передача даних відбувається через зашифровані TCP-з'єднання з використанням транспортного протоколу RLPx із використанням шифрування ECIES (Elliptic Curve Integrated Encryption Scheme).

Bitcoin і Litecoin використовують простіший P2P-протокол на основі обміну інформацією (gossip-based). Masternodes виявляють однорангові вузли за допомогою DNS-посівів і поширення повідомлень

`addr`. Протокол працює через TCP (Bitcoin на порту 8333, Litecoin на порту 9333). Структурованого DHT немає; натомість вузли підтримують базу даних `addrman` (address manager) відомих одноранговых вузлів.

Dash розширює протокол P2P Bitcoin окремим masternode layer. Masternodes утворюють накладну мережу другого рівня з детермінованим порядком, що дозволяє використовувати такі функції, як InstantSend quorums і CoinJoin mixing rounds.

NEO використовує структурований механізм виявлення однорангових вузлів із seed nodes. Консенсус dBFT вимагає меншого набору вузлів консенсусу (наразі 7 в основній мережі), які ”спілкуються” через спеціальний канал консенсусу.



Ethereum (Kademlia) **Bitcoin/LTC (Gossip)** **Dash (Two-tier)**

Рис. 2: Топології мереж P2P для різних блокчейнів

3.3 Структури даних та управління станами

Блокчейн	Станова модель	Структура даних	Сховище
Ethereum	Account-based	Modified Merkle-Patricia Trie	LevelDB / Pebble
Bitcoin	UTXO	Merkle Tree	LevelDB
Dash	UTXO (extended)	Merkle Tree + MN lists	LevelDB / RocksDB
NEO	Account-based	Merkle Tree + state root	LevelDB
Litecoin	UTXO	Merkle Tree	LevelDB

Таблиця 2: Станові моделі і структури даних в блокчейнах

Фундаментальна різниця між account-based системами (Ethereum, NEO) та UTXO (Bitcoin, Dash, Litecoin) проявляється в в процесі розгортання та взаємозамінності модулів. У системах UTXO кожна транзакція споживає попередні виходи та створює нові. У account-based системах глобальний стан явно відстежує залишки на рахунках та сховище контрактів.

3.4 Можливості смарт-контрактів

Блокчейн	Віртувальне середовище	Мови програмування	Здібності блокчейну
Ethereum	EVM	Solidity, Vyper	Turing-complete
Bitcoin	Bitcoin Script	Script opcodes	Non-Turing-complete
Dash	—	—	No native smart contracts
NEO	NeoVM	C#, Python, Java, Go	Turing-complete
Litecoin	Bitcoin Script	Script opcodes	Non-Turing-complete

Таблиця 3: Порівняння можливостей блокчейнів щодо смарт-контрактів

3.5 Deployment Parameters and Requirements

В наступній таблиці хочу навести основні параметри розгортання для запуску повного вузла блокчейну:

Блокчейн	RAM (min)	ROM (min)	Default Port	Primary Client
Ethereum	8 GB	∼1 TB+	30303	Geth / Nethermind
Bitcoin	2 GB	∼600 GB	8333	Bitcoin Core
Dash	4 GB	∼40 GB	9999	Dash Core
NEO	4 GB	∼20 GB	10333	neo-cli (C#)
Litecoin	2 GB	∼120 GB	9333	Litecoin Core

Таблиця 4: Параметри розгортання mainnet у 2024–2025 роках

3.6 Конфігурація генезис-блоку

Будь який блокчейн починається з так званого **генезис-блоку (genesis block)**. У приватних/тестових розгортаннях його конфігурація визначає початкові параметри мережі. Нижче наведено приклад для Ethereum (`genesis.json`):

```
    {
    "config": {
    "chainId": 12345,
    "homesteadBlock": 0,
    "eip150Block": 0,
    "eip155Block": 0,
    "eip158Block": 0,
    "byzantiumBlock": 0,
    "constantinopleBlock": 0,
    "petersburgBlock": 0,
    "istanbulBlock": 0,
    "berlinBlock": 0,
    "londonBlock": 0
    },
```

```
"alloc": {},
"coinbase": "0x0000000000000000000000000000000000000000",
"difficulty": "0x400",
"extraData": "0x00",
"gasLimit": "0x8000000",
"nonce": "0x0000000000000042",
"mixhash": "0x0000000000000000000000000000000000000000000000000000000
↪  00000000",
"parentHash": "0x0000000000000000000000000000000000000000000000000000000
↪  00000000000",
"timestamp": "0x0"
}
```

Основними важливими полями є:

- `chainId` – унікальний ідентифікатор, що відрізняє створювану приватну мережу від основної мережі (chainId=1).
- `difficulty` – контролює складність майнінгової головоломки (mining puzzle complexity); нижчі значення дозволяють швидший майнінг у тестових середовищах.
- `gasLimit` – максимальний дозволений обсяг гасу на блок, що встановлює верхню межу для виконання контракту.
- `alloc` – дозволяє ініціалізувати (робити pre-funding) рахунків валютою Ether при їх створенні.
- Поля `*Block` вказують, під яким номером блоку активуються різні EIP (пропозиції щодо вдосконалення Ethereum) та хард-форки. Встановлення всіх значень на 0 дозволяє активувати всі функції з самого початку.

Для порівняння, генезисний блок Bitcoin hard-coded в клієнтському програмному забезпеченні і не може бути налаштований таким же чином. Тому при розгортанні тестової мережі Bitcoin або мережі regtest параметри модифікуються під час компіляції за допомогою констант або прапорців конфігурації (`-regtest`, `-testnet`).

3.7 Створення блоків і час їх генерації

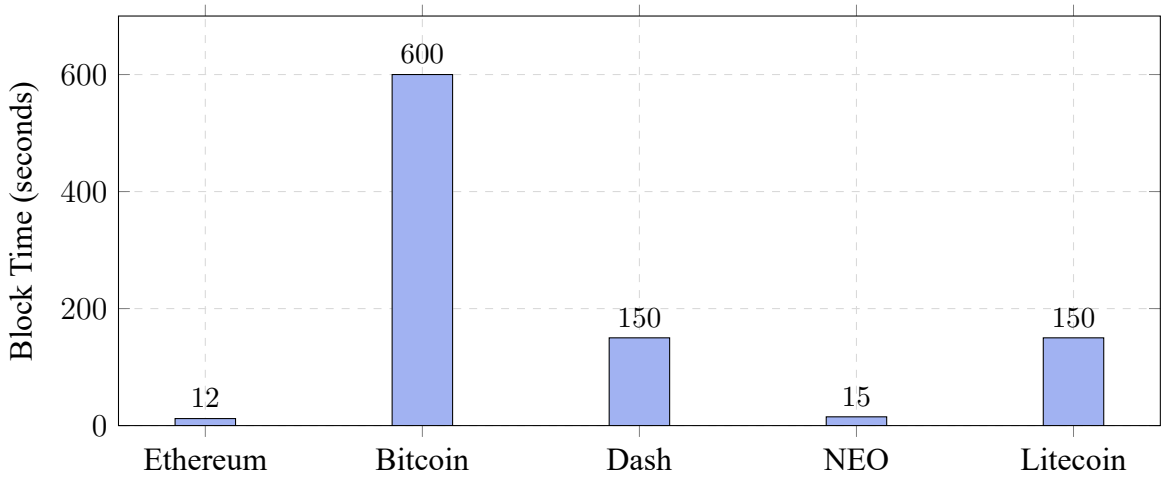


Рис. 3: Середній час генерації блоків (в секундах)

3.8 Пропускна здатність

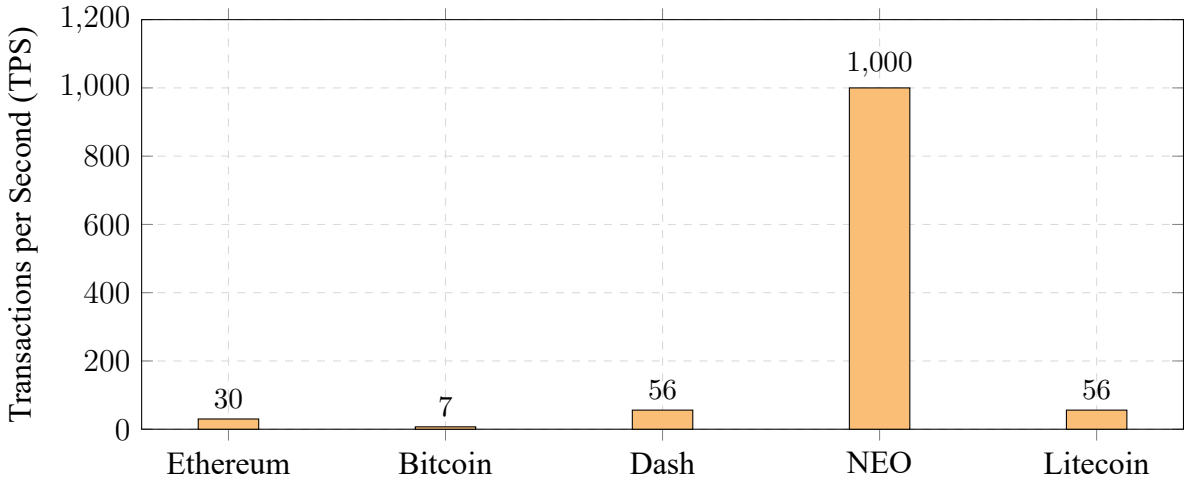


Рис. 4: Приблизна пропускна здатність транзакцій базового рівня (TPS) для різних блокчейнів