Dilithion: A Post-Quantum Cryptocurrency

Version 1.0

October 2025 Launch Date: January 1, 2026

Abstract

Dilithion is a decentralized cryptocurrency designed from the ground up for the post-quantum era. As quantum computers advance toward breaking classical cryptographic systems like ECDSA and RSA, the need for quantum-resistant blockchain technology becomes critical. Dilithion addresses this threat by implementing CRYSTALS-Dilithium, a NIST-standardized post-quantum digital signature scheme, combined with RandomX proof-of-work for ASIC-resistant CPU mining.

This whitepaper presents Dilithion's technical architecture, consensus parameters optimized for large post-quantum signatures, economic model, and roadmap for sustainable decentralized currency in the quantum age.

Key Features:

- **Post-quantum security:** CRY STALS-Dilithium (NIST FIPS 204)
- **ASIC-resistant mining:** RandomX proof-of-work
- **Optimized consensus:** 4-minute blocks for large signature propagation
- Fair distribution: No premine, pure proof-of-work launch
- Fixed supply: 21 million coins
- Launch: January 1, 2026, 00:00:00 UTC

Important Disclosure

Experimental Nature: Dilithion is an experimental cryptocurrency project. This software has NOT been professionally audited and may contain bugs or vulnerabilities. Use at your own risk. AI-Assisted Development: This project was developed with AI assistance (Anthropic's Claude Code). While AI tools enable rapid development and comprehensive documentation, all code requires careful human review and community scrutiny. We believe in full transparency about our development methods. No Guarantees: This project comes with no guarantee of success, security, or value. Users assume all risks. This is not financial advice. Do your own research (DY OR) before participating.

Table of Contents

- [Introduction: The Quantum Threat](#1-introduction-the-quantum-threat)
- [Post-Quantum Cryptography](#2-post-quantum-cryptography)
- [Technical Architecture](#3-technical-architecture)
- [Consensus Mechanism](#4-consensus-mechanism)
- Economic Model](#5-economic-model)
- [Network Security](#6-network-security)
- [Roadmap](#7-roadmap)
- [Conclusion](#8-conclusion)

1. Introduction: The Quantum Threat

1.1 The Problem

Modern cryptocurrency security relies on classical cryptography:

- ECDSA (Bitcoin, Ethereum): Elliptic Curve Digital Signature Algorithm
- RSA: Rivest-Shamir-Adleman encryption
- SHA-256: Secure Hash Algorithm (for mining) Shor's Algorithm (1994) demonstrated that quantum computers can break ECDSA and RSA in polynomial time. While SHA-256 mining receives only a modest speedup (Grover's algorithm), digital signatures are critically vulnerable.

1.2 Timeline to Quantum Threat

Current State (2025):

- IBM: 1,121-qubit quantum computer (Condor)
- Google: Quantum supremacy claimed
- China: Pan-Jianwei's quantum network **Expert Estimates**:
- **2030-2035:** Cryptographically relevant quantum computers (CRQC)
- **Breaking Bitcoin:** Estimated 1,500-3,000 logical qubits required
- **Current trajectory:** Doubling qubits every ~2 years **Conclusion:** Cryptocurrencies must transition to post-quantum cryptography **now** to remain secure over their multi-decade lifespan.

1.3 Existing Cryptocurrency Vulnerability

Cryptocurrency Signature Scheme Quantum Vulnerable?
Migration Plan?
Bitcoin ECDSA ✓ Yes None announced
Ethereum ECDSA ✓ Yes Research phase only
Litecoin ECDSA 💟 Yes None announced

| Monero | EdDSA | ✓ Yes | None announced |

| Dilithion | Dilithium3 | X No | Built-in from genesis |

Critical Issue: Retrofitting existing blockchains with post-quantum cryptography requires:

- Hard fork (community consensus required)
- Wallet migrations (user action required)
- Backward compatibility challenges
- Risk of botched transition **Dilithion's Solution:** Start with post-quantum cryptography from genesis block.

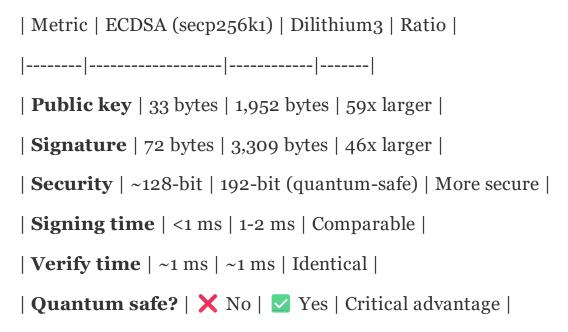
2. Post-Quantum Cryptography

2.1 CRYSTALS-Dilithium

Selection Process:

•	NIST Post-Quantum Cryptography Standardization (2016-2024)				
•	82 initial submissions				
•	Multiple rounds of evaluation				
•	Winner: CRY STALS-Dilithium (2022)				
•	Standardized: FIPS 204 (August 2024) Why Dilithium?				
•	Security: Based on hard lattice problems (Module-LWE, Module-SIS)				
•	Performance: Fast signing and verification				
•	Standardization: Official NIST standard				
•	Analysis: Years of public cryptanalysis, no serious breaks				
•	Versatility: Three security levels (Dilithium2, 3, 5) Dilithion uses Dilithium3:				
•	Security level: NIST Level 3 (equivalent to AES-192)				
•	Public key size: 1,952 bytes				
•	Signature size: 3,309 bytes				
•	Signing speed: ~1-2 milliseconds				
•	Verification speed: ~1 millisecond				

2.2 Comparison to Classical Cryptography



Trade-off: Dilithion transactions are ~15x larger than Bitcoin transactions, but provide quantum resistance.

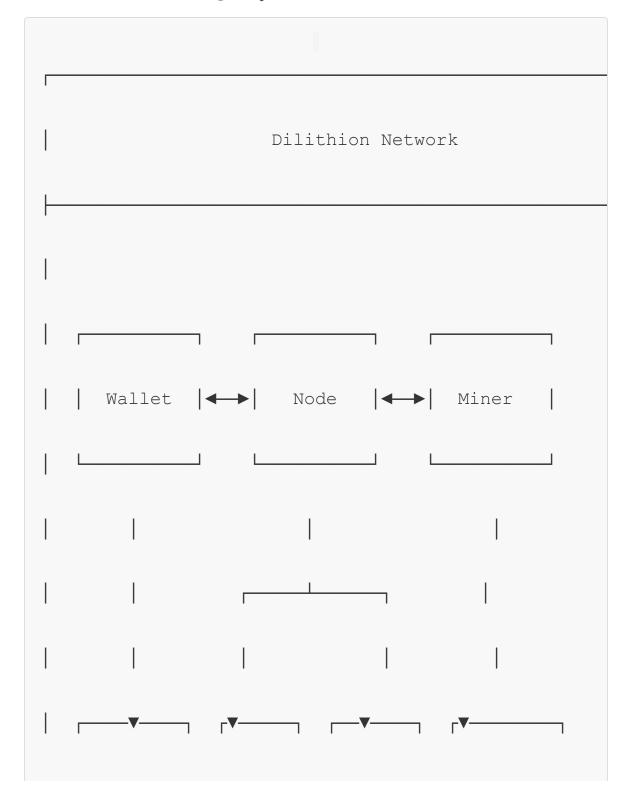
2.3 SHA-3 Hashing

Dilithion uses **SHA-3** (**Keccak**) throughout:

- Address generation: SHA3-256
 Transaction IDs: SHA3-256
- Merkle trees: SHA3-256
- Wallet encryption: SHA3-512 with PBKDF2 Why SHA-3?
- Quantum-resistant (Grover's algorithm provides only quadratic speedup)
- NIST standard (FIPS 202)
- Different construction than SHA-2 (defense in depth)
- Well-analyzed and trusted

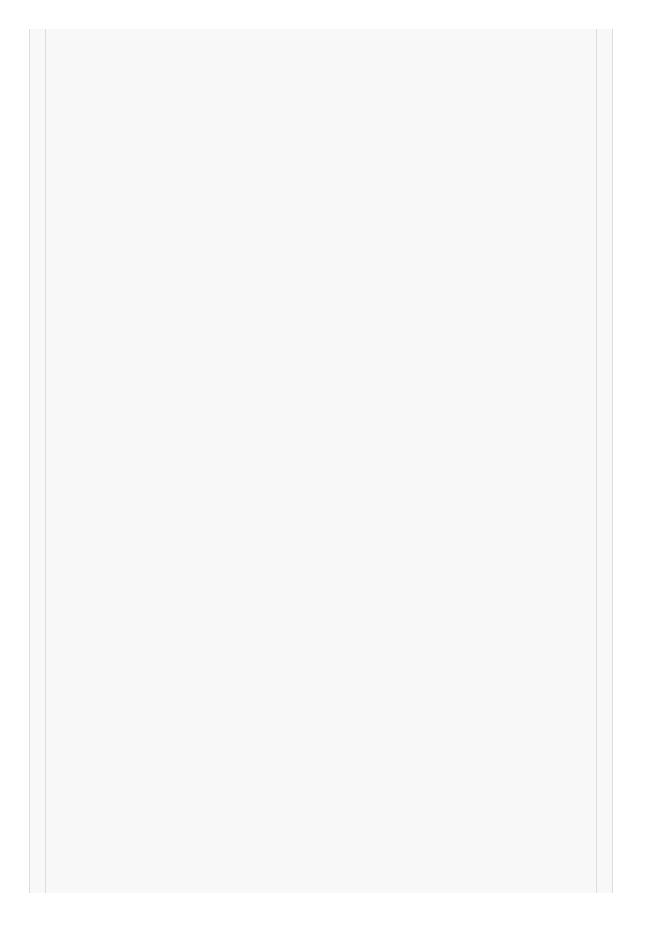
3. Technical Architecture

3.1 System Overview



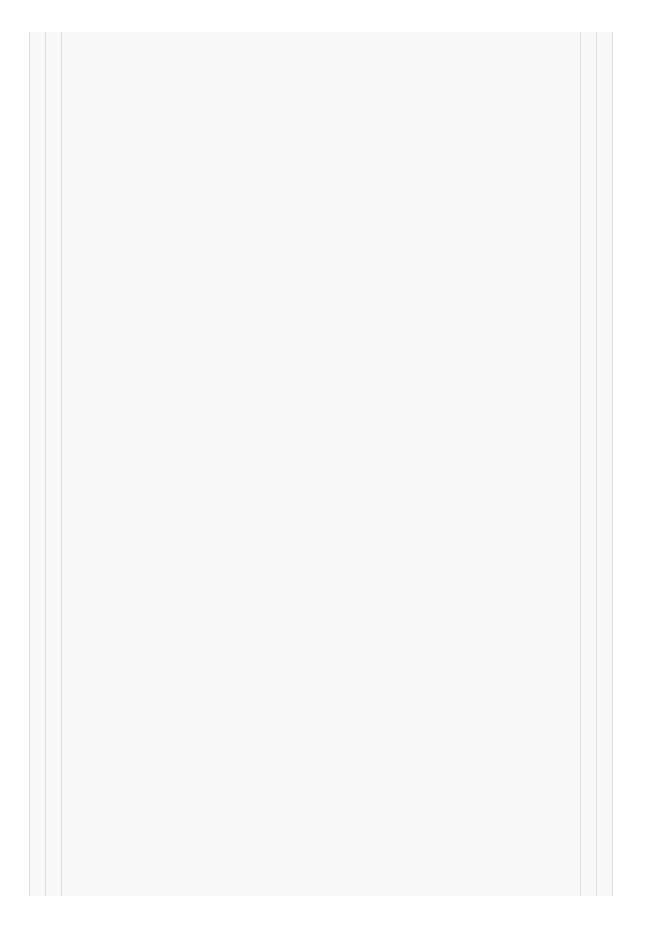
	Dilithium	SHA-3	LevelDB	RandomX
	Sigs	Hash	DB	PoW
L				

3.2 Transaction Structure



```
cpp
class CTransaction {
  int32 t nVersion;
                     // Transac
  uint32 t nLockTime; // Lock ti
};
class CTxIn {
  COutPoint prevout; // Previou
  std::vector scriptSig; // Dilithium signat
  };
class CTxOut {
```

```
// Amount
    CAmount nValue;
    std::vector scriptPubKey; // Dilithium pub
};
```



```
Typical Transaction Sizes:
             1-input, 1-output: ~3,864 bytes
             2-input, 2-output: ~9,598 bytes
               Average: ~5,000-7,000 bytes
                 Comparison to Bitcoin:
               Bitcoin typical: ~250 bytes

    Dilithion is ~15x larger (trade-off for quantum security)

     3.3 Currency Units and Denominations
                    Base Unit: DIL
                      Symbol: DIL
              Total Supply: 21,000,000 DIL
                   Decimal Places: 8
                  Smallest Unit: ions
                1 \text{ DIL} = 100,000,000 ions
• Named after "Dilith-ion" - fitting the post-quantum theme
• Similar to how Bitcoin uses "satoshis" (named after Satoshi
                  Denomination Table:
  | Unit Name | Value in ions | Value in DIL
  |-----|-----|
 | ion | 1 | 0.00000001 DIL | Smallest unit
  | kiloion | 1,000 | 0.00001 DIL | Thousand
```

| megaion | 1,000,000 | 0.01 DIL | Million

| **DIL** | 100,000,000 | 1 DIL | Base currency

Why "ions"?

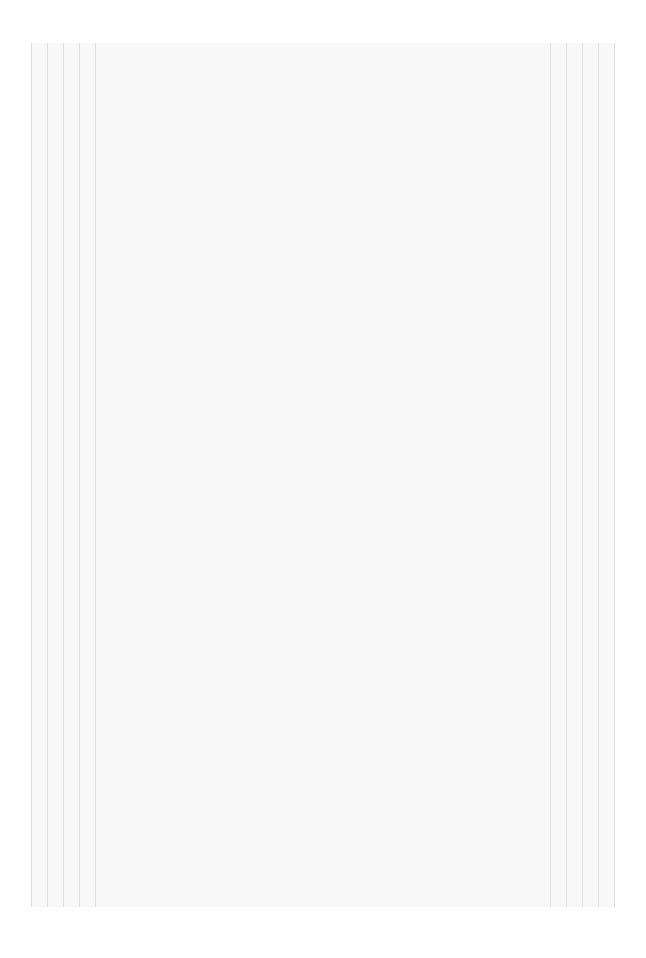
- Consistent with Dilithion branding
- Quantum/scientific theme (from "Dilithium")
- Short, memorable, easy to type
- Avoids confusion with other cryptocurrencies
- Represents the smallest "quantum" of value

Examples:

- Minimum transaction fee: 50,000 ions (0.0005 DIL)
- Typical transaction: 100,000-300,000 ions (0.001-0.003 DIL)
- Block reward (initial): 5,000,000,000 ions (50 DIL)

3.4 Block Structure

```
cpp
class CBlockHeader {
    int32_t nVersion;
                                     // B
    uint256 hashPrevBlock;
                                     // P
                                     // M
    uint256 hashMerkleRoot;
    uint32_t nTime;
                                     // B
    uint32 t nBits;
                                     // D
                                     // R
    uint32 t nNonce;
};
class CBlock {
                                     // B
    CBlockHeader header;
    std::vector vtx; // Transactions
};
```



Block Properties:

Target time: 4 minutes (240 seconds)

Max size: 4 MB (soft limit, adjustable)

Typical size: ~500 KB - 2 MB

Hash algorithm: RandomX (for mining)

Header hash: SHA-3-256

4. Consensus Mechanism

4.1 RandomX Proof-of-Work

Design Goals:

- ASIC-resistant (keep mining decentralized)
- CPU-optimized (accessible to everyone)
- Memory-hard (prevent brute force)

RandomX Characteristics:

- Memory requirement: 2 GB (dataset)
- Algorithm: Random code execution
- Hash rate: ~60-80 H/s per CPU core (consumer hardwa
- ASIC resistance: High (designed to utilize general-

Why RandomX?

- Proven: Used by Monero since 2019
- Fair: Anyone with a CPU can mine
- Decentralized: Prevents mining centralization
- Secure: Well-analyzed, no shortcuts found

4.2 Block Time: 4 Minutes

Decision Rationale:

Original proposal: 2 minutes (5x fast

Final decision: 4 minutes (2.5x faster than Bitcoin

Why 4 minutes is optimal:

- Large Signature Propagation
 - Dilithium signatures: 3,309 byte
 - Typical block: 10-50 transaction
 - Global network needs time to pro
 - 4 minutes reduces orphan rate by

Blockchain Growth

2-minute blocks: 720 blocks/day = \sim 76

4-minute blocks: 360 blocks/day = ~36

Balanced Confirmation Time Bitcoin: 10 min/block × 6 confirmations Dilithion: 4 min/block × 3 confirmations Litecoin: 2.5 min/block × 6 confirmations Better Emission Schedule 2-min: 62.6% mined in Year 1 (too aggressive) 4-min: 31.3% mined in Year 1 (balanced distributi

Global Mining Fairness

- Network latency (200-400ms globally) becomes smaller % of block time
- Miners worldwide have equal opportunity

4.3 Difficulty Adjustment

Algorithm: Similar to Bitcoin's difficulty adjustment

```
срр
// Adjust difficulty every 2016 blocks
const int64 t DIFFICULTY ADJUSTMENT INTERVAL = 2016;
const int64 t BLOCK TARGET SPACING = 240; // 4 minut
// Target timespan: 2016 blocks × 4 minutes = 5.6 da
const int64 t TARGET TIMESPAN = DIFFICULTY ADJUSTMEN
// Difficulty adjustment formula:
new difficulty = old difficulty * (actual time / tar
// With bounds:
new difficulty = clamp (new difficulty, old difficult
```

Properties:

Adjusts every ~5.6 days

Maximum change: 4x per adjustment

Prevents difficulty manipulation attacks

Responsive to hash rate changes

4.4 Timestamp Validation

Rules:

Block time must not be more than 2 hours in the future
 Block time must be greater than median-time-past (last 11 blocks)

Prevents:

Time manipulation attacks

Difficulty adjustment gaming

Chain reorganization exploits

5. Economic Model

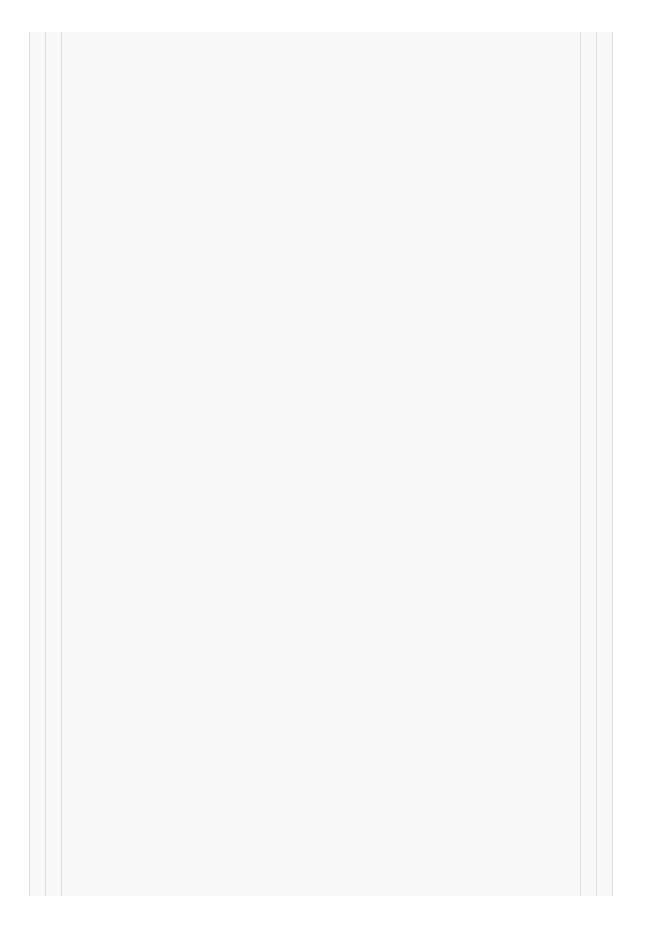
5.1 Supply Schedule

Total Supply: 21,000,000 DIL (fixed cap)

Initial Reward: 50 DIL per block

Block Time: 4 minutes (240 seconds)

Halving: Every 210,000 blocks (~1.6 ye



5.2 Emission Schedule

| 0 | 0 - 209,999 | 50 DIL | 1.60 years | 1

| Halving | Block Range | Reward | Duration

| 1 | 210k - 419,999 | 25 DIL | 1.60 years

| 2 | 420k - 629,999 | 12.5 DIL | 1.60 year

| 3 | 630k - 839,999 | 6.25 DIL | 1.60 year

|4+|840k+|<6.25 DIL|~8 years|~1,31

Year-by-Year Emission:

• Year 1: 6,570,000 DIL (31.3% of total supply)

Year 2: 5,250,000 DIL (25.0%)

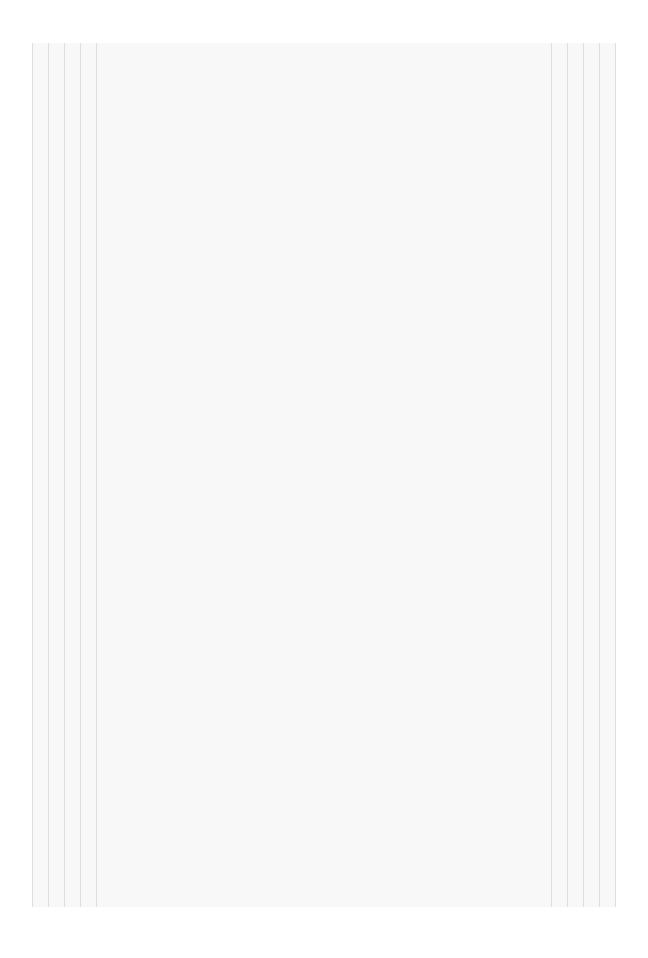
Year 3: 3,285,000 DIL (15.6%)

Year 5: 89.1% mined

Year 13: 99%+ mined

5.3 Comparison to Bitcoin | Metric | Bitcoin | Dilithion | Ratio | |-----| | Total Supply | 21M BTC | 21M DIL | 1:1 | | Initial Reward | 50 BTC | 50 DIL | 1:1 | | Block Time | 10 min | 4 min | 2.5x faster | Halving Period | 210,000 blocks | 210,000 | First Halving | ~4 years | ~1.6 years | 2 | 99% Mined | ~32 years | ~12.8 years | 2.5 | Year 1 Emission | 12.5% | 31.3% | 2.5x fa Conclusion: Dilithion's emission is exactly 2.5x faster tha 5.4 Transaction Fees Fee Model (Option A):

```
срр
// Consensus parameters
MIN_TX_FEE = 50,000 ions // 0.0
FEE_PER_BYTE = 25 ions // 25
MIN RELAY TX FEE = 100,000 ions // 0.0
// Fee calculation
fee = MIN TX FEE + (transaction size byt
```



Typical Transact	ion Fees:				
Transaction Type	Size Fee (DIL)				
1-in, 1-out 3,86	4 bytes 0.00147				
1-in, 2-out 5,81	6 bytes 0.00195				
2-in, 2-out 9,59	8 bytes 0.00290				
Design Goa	ls:				
	• Affordable: Fees remain negligible (<\$0.003 per tra				
• Spam protection: 3x higher t					
Miner incentives: Provides mSustainable: Scales with tr	_				
Long-term Fee	Market:				
• Short-term: Fixed fee model	(simple, predictable)				
• Year 1-2: Monitor usage patt	• Year 1-2: Monitor usage patterns and fee adequacy				
• Year 2+: Implement dynamic f	ee market (EIP-1559 sty				
5.5 Inflation	on Rate				
Year Supply Star	t Annual Emissi				

| 1 | 0 | 6,570,000 | 6,570,000 | N/A | 2 | 6,570,000 | 5,250,000 | 11,820, | 3 | 11,820,000 | 3,285,000 | 15,105 | 4 | 15,105,000 | 1,965,000 | 17,070 | 5 | 17,070,000 | 1,642,500 | 18,712 | 10 | ~20,200,000 | ~205,000 | ~20,4 | 20 | ~20,900,000 | ~12,800 | ~20,91 Observation: Inflation drops to single digits by Ye 6. Network Security

6.1 Attack Vector Analysis

6.1.1 51% Attack

Definition: Attacker controls >50% of network hash

Dilithion Defenses:

RandomX CPU Mining

- No ASICs available (ASIC-resista
- Attacker must acquire thousands
- Very expensive and detectable

Confirmation Requirements

Small tx (<\$100): 3 confirmations

Medium tx (\$1K): 6 confirmations

Large tx (\$10K+): 10 confirmations

Exchange deposits: 20+ confirmation

Economic Disincentive

Attack cost: \$20,000-\$50,000 (hardware)

Attack profit: \$1,000-\$5,000 (one-time, if

Consequence: Coin price crashes, attacker's

Result: Attacker loses money

Risk Level: LOW to MEDIUM (economically impractical)

6.1.2 Double-Spend Attack

Mitigation:

Requires 51% attack to succeed

Exchanges wait for multiple confirmations

Cost exceeds potential gain

Risk Level: LOW (same as 51% attack)

6.1.3 Sybil Attack

Definition: Attacker creates many fake network nodes

Dilithion Defenses:

Mining power matters, not node count

Nodes don't receive rewards (no incentive to fake)

Peer quality scoring (future enhancement)

Risk Level: LOW (ineffective attack vector)

6.1.4 Eclipse Attack

Definition: Isolate a node from the honest network

Mitigation:

Multiple seed nodes (DNS + hardcoded)

Peer diversity requirements

Automatic peer discovery

Risk Level: LOW (standard Bitcoin-style defenses)

6.1.5 Quantum Computer Attack

Definition: Use quantum computer to break cryptography

Dilithion Defense:

Signatures: Quantum-resistant (Dilithium3)

Hashing: Quantum-resistant (SHA-3, only Grover speedup)

Mining: Quantum computers provide minimal advantage (Grover = 2x sp

Verdict: ☑ Dilithion is quantum-safe (primary design goal)

6.2 Wallet Security

Features:

AES-256-CBC Encryption

- Industry-standard wallet encryption
- PBKDF2-SHA3 key derivation (100,000 rounds)
- Two-tier architecture (master key + encrypte

Lock/Unlock Mechanism

- Automatic lock after timeout
- Secure memory wiping
- Password strength requirements

Backup & Recovery

- Binary wallet file format (DILWLT01)
- Encrypted backups
- Future: HD wallet with 24-word seed phrase

Best Practices:

- Always encrypt wallet with strong passphrase
- Regular backups to multiple locations
- Store backups encrypted
- Use cold storage for large amounts

6.3 Network Monitoring

Planned Infrastructure:

Seed Nodes: 3-5 globally distributed nodes

DNS Seeds: Automatic peer discovery

Block Explorer: Public blockchain viewer

Hash Rate Monitor: Real-time network statistics

7. Roadmap

7.1 Genesis Launch (January 1, 2026)

Launch Specifications:

Genesis timestamp: January 1, 2026, 00:00:00 UTC

Initial difficulty: Bitcoin-equivalent (0x1d00ffff)

First halving: Block 210,000 (~July 2027)

Network: Mainnet with seed nodes

Launch Readiness:

- ✓ Core node implementation complete
 - ☑ Wallet functionality complete
 - ✓ Mining integration complete
 - Consensus parameters finalized
 - ✓ Security features implemented
 - ✓ Testing complete

7.2 Month 1-2 (Launch Infrastructure)

Priority Features:

Desktop GUI Wallet

- User-friendly interface
- One-click mining
- Visual transaction history
- Windows, macOS, Linux support

Website Launch

- Countdown timer
- Live network dashboard
- Getting started guide
- Documentation

Block Explorer

- View blocks and transactions
- Search functionality

- Network statistics
- API for developers

Mining Pool Software

- Stratum protocol implementation
- Pool operator toolkit
- Fair reward distribution

7.3 Month 2-3 (Ecosystem Growth)

Key Milestones:

- HD Wallet Implementation (HIGH PRIORITY)
- 24-word seed phrase recovery
- BIP32/BIP39 adapted for Dilithium
- Infinite address generation from single seed
- Impact: Prevents coin loss, major UX improve

Mobile Wallets

- iOS app

- Android app
- QR code scanning
- Push notifications
- SPV-style lightweight verification

Exchange Listings

- Engage major exchanges (Binance, Coinbase, F
- Provide integration documentation
- Listing applications submitted

Dynamic Fee Market

- Fee estimation API
- Market-driven pricing
- Mempool analytics

7.4 Month 6+ (Advanced Features)

Long-term Enhancements: Payment Integration

- Merchant tools
- Point-of-sale systems
- E-commerce plugins

Hardware Wallet Support

- Research custom PQC hardware wallet
- Ledger/Trezor collaboration exploration
- Secure key storage solutions

Layer 2 Scaling

- Lightning Network research (adapted for PQC)
- Payment channels
- Atomic swaps

Signature Aggregation

- Research academic developments
- Implement if available (75-85% size reduction

- Significant transaction size improvement

7.5 Year 2+ (Ecosystem Maturity)

Vision:

DeFi Integration

- Decentralized exchanges
- Lending protocols
- Liquidity pools

Smart Contracts (Research)

- Post-quantum compatible VM
- Turing-complete capabilities
- Security-first design

Privacy Features (Optional)

- Ring signatures or similar
- Optional privacy transactions

- Balance transparency vs. privacy

Cross-chain Bridges

- Connect to other blockchains
- Atomic swaps
- Interoperability protocols

8. Conclusion

8.1 Why Dilithion Matters

The Quantum Threat is Real:

- Timeline: 5-10 years to cryptographically relevant quantum computer
 - Existing cryptocurrencies are vulnerable
- Transition will be difficult and contentious

Action needed now

Dilithion's Solution:

- Built quantum-safe from genesis
 - No migration required
- Users protected from day one
- Proven cryptography (NIST standard)

8.2 Technical Excellence Optimized for Post-Quantum Era: 4-minute blocks accommodate large signatures Balanced emission schedule (31.3% Year 1) Affordable transaction fees ASIC-resistant CPU mining Professional-grade security Comparison to Competition: | Feature | Bitcoin | Ethereum | Other PQC Project |----|----| | Quantum-safe signatures | X No | X No | A F | ASIC-resistant mining | X No | N/A (PoS) | Va: | Optimized for PQC | X No | X No | A Partial | Fixed supply | 🗹 Yes | 🗶 No | Varies | 🔽 Ye | Launch readiness | 🗸 Mature | 🗸 Mature | 🛕 8.3 Fair Launch Principles Dilithion adheres to fair launch principles:

✓ No premine ☑ No ICO / token sale ✓ No founder allocation ✓ No venture capital pre-allocation ✓ Pure proof-of-work from genesis ✓ Open-source (MIT license) Community-driven development Everyone starts equal on January 1, 2026. 8.4 Long-term Vision Dilithion aims to be: The standard for quantum-safe cryptocurrency A store of value in the post-quantum era A medium of exchange with reasonable fees A platform for decentralized applications A community of quantum-aware developers and users Mission Statement: > "Secure digital currency for the quantum age, k 8.5 Call to Action For Miners: CPU mining opens January 1, 2026 Fair distribution, no ASIC advantage Early adoption opportunity For Developers: Open-source codebase (GitHub)

```
    Documentation available
    Contribute to post-quantum crypto future
    For Users:
    Download wallet before launch
    Participate in first quantum-safe cryptocurrency
    Be part of the solution
    For Investors:
    Study the technology
    Understand the quantum threat
    Position for the post-quantum era
```

Technical Specifications Summary

```
| Signature Algorithm | CRYSTALS-Dilithium3 (NIST
| Hash Algorithm | SHA-3-256 (NIST FIPS 202) |
| Mining Algorithm | RandomX (Monero-derived, ASI
| Difficulty Adjustment | Every 2,016 blocks (~5.
| Address Format | Dilithium3 public key hash (SH
| Transaction Fee | 0.0005 DIL base + 25 sats/byt
| Confirmations (typical) | 3-10 blocks (12-40 mi
| Genesis Block | Hardcoded, January 1, 2026 |
```

References

- NIST. (2024). *FIPS 204: Module-Lattice-Based Digital Signature Sta
- Ducas, L., et al. (2018). *CRYSTALS-Dilithium: A Lattice-Based Digi
- Shor, P. (1994). *Algorithms for quantum computation: Discrete loga

- National Academies of Sciences, Engineering, and Medicine. (2019).
- Monero Research Lab. (2019). *RandomX: CPU-optimized Proof-of-Work*
- Nakamoto, S. (2008). *Bitcoin: A Peer-to-Peer Electronic Cash Syste
- Bernstein, D. J., et al. (2015). *Post-quantum cryptography*. Natur

Appendix A: Glossary

ASIC (Application-Specific Integrated Circuit): Specialized hardwar CRYSTALS-Dilithium: NIST-standardized post-quantum digital signatur Halving: Reduction of block reward by 50%, occurs every 210,000 blo Hash Rate: Measure of mining computational power, typically measure Lattice Cryptography: Post-quantum cryptographic approach based on Module-LWE: Learning With Errors over Module Lattices, the hard pro Orphan Block: Valid block that's not included in the longest chain, Post-Quantum Cryptography (PQC): Cryptographic algorithms designed RandomX: ASIC-resistant proof-of-work algorithm optimized for gener SHA-3: Secure Hash Algorithm 3, NIST-standardized hash function (Ke Shor's Algorithm: Quantum algorithm that can break RSA and ECDSA in

Appendix B: Contact & Community

Website: https://dilithion.org (launching soon)

GitHub: https://github.com/WillBarton888/dilithion

Discord: [Community server - launching Week 2]

Twitter/X: @DilithionCoin

Reddit: r/dilithion

Contact:

General Inquiries: team@dilithion.org
Security Reports: security@dilithion.org
Media Inquiries: media@dilithion.org
User Support: support@dilithion.org

Dilithion Whitepaper v1.0
October 2025
"Quantum-Safe. Community-Driven. Fair Launch."

Disclaimer: This whitepaper is for informational and educational pu