ROOD: Proof of Luck Mining System

Abstract

ROOD is a fixed-supply digital token distributed exclusively through a provably fair weighted lottery. A new block is produced every ten minutes. The winner is selected proportionally to their contributed hashrate using the Efraimidis–Spirakis algorithm — the same statistical guarantee as Proof-of-Work, without energy waste. Total supply is capped at 21,000,000 ROOD. Block rewards halve every 100 days, creating a predictable, deflationary emission schedule modeled after Bitcoin.

1. Introduction

Bitcoin solved the double-spend problem without a trusted third party by linking computational work to block production. The chain with the most accumulated work is, by definition, the canonical chain.

ROOD borrows Bitcoin's emission schedule and scarcity model but operates inside a closed virtual environment. There is no physical hardware, no proof-of-work puzzle to solve, and no energy race. Instead, the protocol runs a weighted random draw every ten minutes. Each participant's probability of winning is exactly proportional to their virtual hashrate — mathematically equivalent to the long-run expected payout of Proof-of-Work, with zero wasted computation.

2. Emission Schedule

The total supply is hard-capped at **21,000,000 ROOD**. A block is produced every **10 minutes**. The initial block reward is **729.17 ROOD**. Every **14,400 blocks** (100 days) the reward is cut in half:

```
reward(height) = 729.17 / 2^floor(height / 14400)
```

Epoch	Days	Reward / Block	Blocks	Issued (ROOD)
0	0 – 100	729.17	14,400	10,500,048
1	100 – 200	364.58	14,400	5,250,012
2	200 – 300	182.29	14,400	2,625,000
3	300 – 400	91.15	14,400	1,312,560
…	…	…	…	…

Table 1. Emission schedule by epoch.

The geometric series converges to 21,000,000. After the final satoshi is emitted, no new ROOD can ever be created.

3. The Weighted Lottery

3.1 Algorithm

At each block height h, the protocol selects one winner from the set of active miners using the **Efraimidis–Spirakis weighted reservoir sampling** algorithm:

```
SELECT id FROM miners
WHERE hashrate > 0
AND electricity_runs_out_at > NOW()
ORDER BY random() ^ (1.0 / hashrate) DESC
LIMIT 1
```

For each miner i with hashrate w_i, a key k_i is computed:

```
k_i = u_i ^ (1 / w_i), u_i ~ Uniform(0, 1)
```

The miner with the highest key wins the block. This produces exactly proportional probabilities:

```
P(miner i wins) = w_i / Σ w_j
```

3.2 Properties

- **Fairness.** No miner can influence their key beyond their declared hashrate.
- **Verifiability.** The draw uses the database's cryptographic RNG, independent of any participant.
- **Proportionality.** Doubling hashrate doubles long-run earnings, identical to Proof-of-Work.
- **No wasted energy.** The lottery is O(n) in the number of active miners and runs in milliseconds.

3.3 Expected Earnings

A miner with hashrate h contributing to a network of total hashrate H expects:

```
E[ROOD per day] = (h / H) × 144 blocks/day × reward
```

GPU Model	Hashrate	E[ROOD/day] at H = 200,000 MH/s
GTX 1060	15 MH/s	~0.79
RTX 3070	110 MH/s	~5.77
RTX 4090	510 MH/s	~26.76
RTX 5090	1,080 MH/s	~56.66

Table 2. Expected daily earnings by GPU model (Epoch 0).

4. Virtual Hardware

4.1 GPUs

Each participant builds a virtual mining farm by acquiring GPU cards. Cards vary in hashrate, power consumption, price, and durability.

Model	Hashrate	Power (kWh/h)	Lifespan	Repair Cost	Price (ROOD)
GTX 1060	15 MH/s	0.08	7 days	20	Free
GTX 1080Ti	35 MH/s	0.13	6.5 days	54	1,000
RTX 3060	50 MH/s	0.12	6 days	70	1,500
RTX 3070	110 MH/s	0.26	5.5 days	135	2,500
RTX 3080	240 MH/s	0.54	5 days	270	4,500
RTX 4090	510 MH/s	1.10	4.5 days	480	8,000

Model	Hashrate	Power (kWh/h)	Lifespan	Repair Cost	Price (ROOD)
RTX 5090	1,080 MH/s	2.25	4 days	900	15,000

Table 3. GPU specifications.

4.2 Wear

A GPU accumulates wear at a constant rate determined by its model:

```
wear_rate = 100% / (lifespan_days × 86400 seconds)
```

When wear reaches 0%, the card stops contributing hashrate and must be repaired before returning to service. Repair cost scales with remaining wear:

```
repair_cost = base_cost × (1 - wear / 100)
```

Repairing a fully broken card costs the full base_cost. Repairing a 50%-worn card costs half.

4.3 Electricity

Cards only mine while electricity is available. Each card draws power at its rated kWh/h. When electricity reaches 0, all installed cards are suspended from the lottery until electricity is replenished. The protocol tracks electricity_runs_out_at and excludes expired miners from each draw.

5. Rigs

Cards are installed into rig frames. Each rig provides a fixed number of GPU slots:

Rig Model	GPU Slots	Price (ROOD)
Starter Rig	7	3,000
Mid Rig	14	8,000
Full Rig	21	20,000

Table 4. Rig specifications.

A miner's total hashrate is the sum of hashrates of all installed, operational, powered cards across all rigs.

6. Token Flow

6.1 Minting

ROOD is minted exclusively through block rewards. Every 10 minutes, exactly one winner receives reward(height) ROOD added to their pending_reward balance. No other minting mechanism exists.

6.2 Claiming

Accumulated pending_reward can be claimed at any time. Claimed ROOD moves to the participant's spendable balance and can be used to purchase hardware, electricity, or withdrawn to an external Solana wallet.

6.3 Withdrawal

Withdrawals are processed on-chain via the Solana SPL token program. A 5% protocol fee is applied:

```
sent_on_chain = amount × 0.95

fee_burned = amount × 0.05
```

Minimum withdrawal: 10 ROOD.

6.4 Deposit

Participants may deposit ROOD from an external Solana wallet to their in-game balance using their assigned deposit address. Deposits are detected via Helius webhook and credited instantly. Upon detection, deposited tokens are swept from the user's deposit address to the protocol treasury, ensuring the treasury always holds sufficient liquidity for withdrawals.

7. Scarcity and Halvings

The halving schedule creates four distinct eras of increasing scarcity:

```
Epoch 0 (days 0–100): ~50% of total supply emitted

Epoch 1 (days 100–200): ~25% of total supply emitted

Epoch 2 (days 200–300): ~12.5% of total supply emitted

Epoch 3+ (days 300+ ): remaining supply over many years
```

Early participants capture a disproportionate share of the total supply. This incentive structure mirrors Bitcoin: the cost of waiting is paid in dilution.

8. Network Hashrate and Difficulty

Unlike Bitcoin, ROOD has no dynamic difficulty adjustment. The lottery draw is inherently self-normalizing: as total hashrate H grows, each individual miner's probability w_i / H shrinks proportionally. Reward per unit of hashrate decreases with competition, exactly as in Proof-of-Work, without requiring a difficulty retarget mechanism.

9. Conclusion

ROOD demonstrates that Bitcoin's core economic properties — fixed supply, halving schedule, proportional reward, and Sybil resistance — can be achieved without proof-of-work energy expenditure. The Efraimidis–Spirakis weighted lottery provides the same long-run fairness guarantees as hash racing, with deterministic O(n) complexity and full auditability. The result is a system where the only path to increased earnings is acquiring more hashrate, and the only path to total supply is waiting — one block at a time.

`Total supply: 21,000,000 ROOD · Block time: 10 minutes · Halving: every 100 days`