

GENESIScoin Technical Mechanics

Pre-Sale System

- **Supply:** 33B GENc
- **Stages: 5, with progressive pricing and bonuses up to +15%**
 - **Stage 1:** 0.0001 USD / GENc (+4% bonus) – no time limit (until sold out)
 - **Stage 2:** 0.0002 USD / GENc (+8% bonus) – duration: **50 days**
 - **Stage 3:** 0.0003 USD / GENc (+9% bonus) – duration: **50 days**
 - **Stage 4:** 0.0004 USD / GENc (+10% bonus) – duration: **50 days**
 - **Stage 5:** 0.0005 USD / GENc (+15% bonus) – no time limit (until sold out)
- **Limits (applied only during Pre-Sale, no tax):**
 - Max purchase: **500M GENc** per transaction
 - Max wallet: **1B GENc (1% of total supply)**
- **Unsold tokens redistributed automatically (on-chain):**
 - 15% → Liquidity Pool
 - 15% → Burn
 - 15% → TeamG
 - 15% → Community
 - 35% → Dividend
 - 5% → DevG
- **Funds distribution (BNB):** All collected funds are **redistributed once every 24h** across:
 - DevG (10%)
 - TeamG (10%)
 - Liquidity Pool (50%)
 - Marketing (6%)
 - Community (5%)
 - SecureAudit (7%)
 - LevelUpVision (7%)
 - Project Expansion (5%)
- **Automation:** Distributions are executed trustlessly by the **Oracle bot**, which triggers `distributeFunds()` every **86400 seconds (24h)**

B100D Loyalty & Final Bonus – Full Scheme

Overview

A 100-day loyalty program with daily GENc payouts and a proportional final bonus. Eligibility and activity are tracked on-chain, enforced by bots and oracle automation.

1. Eligibility & Registration

- Requirement: $\geq 15,000,000$ GENc balance.
- Fee: **0.1 BNB** → routed to GENc Gas Reserve.
- Registration available only in Stage 5 of Pre-Sale.
- On registration:
 - `b100dParticipants[user]` created with `initialHold` and `isActive = true`.
 - Event `B100DRegistered` emitted.
 - `B100DBot` records user + `initialHold`.

2. Monitoring & Eject Logic

- **B100DBot:**
 - Subscribes to all `Transfer` events for registered addresses.
 - Verifies `balanceOf(user) ≥ initialHold`.
 - If `balance < initialHold` → calls `ejectFromWhitelist(user)`.

- **Contract effect:**
 - Sets `isActive = false`.
 - Emits `RemovedFromB100DWhitelist(user)`.
 - User excluded from further daily payouts and Final Bonus.

3. Daily Payouts

- **Trigger:** Oracle calls `triggerDailyB100DPayout()` once per day.
- **Distribution:**
 - **15,000 GENc** per active participant.
 - Executed in batches of 50 users for gas safety.
 - If transfer fails → `ClaimFailed` event emitted.
- **Progression:** after all users processed → `B100DDayCompleted(day)` emitted.

4. Final Bonus

- **Trigger:** after **day 100**, `FinalBBot` calls `triggerFinalBonusDistribution()`.
- **Eligibility:** only users with `isActive = true`.
- **Calculation:**
 - Contract sums all `initialHold` of active users.
 - User's share = $(\text{initialHold} \div \text{totalInitialHoldActive}) \times \text{remaining B100D pool}$.
- **Completion:**
 - When all eligible processed → `FinalBonusCompleted()` emitted.
 - Users flagged in `finalBonusClaimed` cannot double-claim.
- **Distribution:**
 - Batched (max 75 users per tx).
 - Each transfer reduces the B100D pool.
 - `FinalBonusDistributed(user, amount)` logged.

Public Sale + Daily Bonus

- **Phase 1 (Days 1–100):**
 - Minimum buy: **200,000 GENc**
 - Daily bonus: **100,000 GENc**
 - Daily cap: **100 wallets/day**
 - Unused bonuses are stored in the Bonus Pool
- **Phase 2 (Days 101–200):**
 - Minimum buy: **300,000 GENc**
 - Daily bonus: **150,000 GENc**
 - Daily cap: **75 wallets/day**
 - Unused bonuses are stored in the Bonus Pool
- **Phase 3 (Days 201–300):**
 - Minimum buy: **500,000 GENc**
 - Daily bonus: **200,000 GENc**
 - Daily cap: **50 wallets/day**
 - Unused bonuses are **stored in the Bonus Pool**
- **Phase 4 (Day 301 → until Bonus Pool exhausted):**
 - Minimum buy: **700,000 GENc**
 - Daily bonus: **300,000 GENc**
 - Daily cap: **25 wallets/day**
 - From this point, the **Bonus Pool is gradually distributed** together with the daily allocation.
 - **No expiry** – the system continues until the pool is fully depleted.

Dividend System

- **DividendGENc Pool**
 - Minimum holding requirement: **5,000,000 GENc**
 - **Cycle length:** 21–28 days (randomized by DividendVerifier).
 - **Progressive combo:** Starts at 2%, increases by +1% every 2 completed cycles, capped at ~10%.
 - **HoldProof enforced:** If balance drops below required threshold during a cycle, the address is removed and receives no payout.
 - **+151% Rule:** If a holder's balance increases by $\geq 151\%$ compared to the current `initialHold` during an active cycle, the system updates `initialHold` immediately. All subsequent payouts (even in the same cycle) are calculated against this new updated threshold.
- **Rewards (BNB)**
 - Integrated directly into the **GENc Token tax logic**.
 - Funded and distributed automatically in **BNB** from transaction taxes.
 - Minimum holding requirement: **10,000,000 GENc**.
 - Operates as a **standard dividend distribution system**, fully independent from the DividendGENc pool.

Airdrop System

- Management
 - Controlled by airT tracker and airdropConsole CLI.
 - airT maintains the `trackedHolders.json` file with all eligible addresses.
- Entry Requirement
 - Minimum balance: 100,000 GENc to participate.
- Distribution Modes (via CLI)
 - Proportional mode:
 - Each eligible wallet receives a share of the pool proportional to its GENc balance.
 - Formula: $(\text{userBalance} \div \text{totalEligibleBalance}) \times \text{airdropPoolGENc}$.
 - Fixed mode:
 - Each eligible wallet receives the same fixed amount.
 - Requires that the pool has sufficient balance before execution.
- Pools
 - Pre-Sale Airdrop: up to 1.5B GENc, unlocked progressively across 5 stages (minimum 600M guaranteed).
 - Milestone Holders Airdrop: up to 1.0B GENc, unlocked at holder-count milestones (500 → 5K holders).
 - Additional Allocation: up to 2.5B GENc reserved (DAO or future events; distribution logic TBD).
 - Total Airdrop Allocation: 5.0B GENc.
- Automation & Batching
 - Distributions are executed in batches of up to 200 addresses per transaction for gas safety.
 - All executed distributions are logged in `airdropLog.json`.
- On-Chain Logic
 - Uses `airdropByList(recipients, amounts)` in the PublicSaleGENc contract.
 - Pool status tracked via `airdropPoolGENc()`.