buttcoin: A Satoshi Approach to Memecoin Virality

Buttoshi Nakaboto
butt@xbuttcoin.fun
www.xbuttcoin.fun

Abstract

buttcoin is a memecoin designed as if it were launched by Satoshi in 2026, in an environment dominated by social feeds and generative AI rather than CPU cycles. Instead of proof-of-work, it uses Proof of Attention (PoA): short, measurable engagement raids around posts that always feature a butt-focused visual serve as the “work” that authorizes the protocol to spend part of the accumulated fees from pump.fun into buybacks, liquidity, and development.

a block in the buttcoin blockchain — the dev buy amount is calculated through pump.fun total rewards and generated attention on X. These rewards rebuy into buttcoin, permanently marking each buy as an image in the buttcoin chain to create the longest ongoing Proof of Attention buttcoin chain.

The visual universe is intentionally broad: all butts are allowed — models, celebrities, fitness creators, anonymous users; so long as content respects platform rules. The goal is to formalize how attention to a repeatable visual theme (butts) gets converted into transparent, rule-based on-chain actions, rather than ad hoc marketing decisions.

1. Introduction

The cryptocurrency market has gone through several waves of memecoin hype, from the first joke tokens to multibillion-dollar projects built around a single image. Memes have become a real social and cultural force that brings people into web3 faster than most DeFi products or infrastructure. At the same time, another pattern has appeared: fast, “one-use” tokens with no clear idea behind them, no transparency and no plan beyond the first pump.

buttcoin is designed as an experimental project that tries to remove some of these problems. The idea is to formalise what actually makes a meme work: attention in social networks, a repeated visual (butts), and a predictable link between engagement and on-chain actions. Instead of the usual model where everything depends on one lucky viral post and manual actions from the team, buttcoin introduces Proof of Attention (PoA).

Attention to posts with butt visuals is treated as a measurable resource that gives the protocol the right to spend part of the accumulated creator rewards as a block reward. This reward takes the form of buying back BUTT from the market, with buys marked as sequential blocks in the buttcoin attention chain.

2. Proof of Attention (PoA)

1. Why this exists

Bitcoin pays miners for burning electricity. For buttcoin, the scarce resource is different: it's attention, views, reposts, memes, people piling into the same post and pushing it through the feed. The idea of Proof of Attention (PoA) is: If the community creates real engagement around buttcoin posts, the system must react in a predictable way: rebuy BUTT using pump.fun rewards and mark it as a new block. No random "dev decided to buy", no invisible moves. Just fixed rules that convert attention into on-chain blocks.

2. Main components

We keep the system minimal:

- buttcoin (BUTT): The token itself. Ticker: BUTT.
- Attention Treasury: A treasury wallet funded 100% by pump.fun creator rewards (dev fees).
- ButtAI: An AI agent that generates butt-focused memes, posts them on X, starts raids in Telegram/community channels, tracks engagement, and on success triggers the PoA flow: claim fees, calculate buy amount based on attention score, execute rebuy, and log it as a butt block with the image.
- DEX pool BUTT/SOL: Where buybacks happen and market cap is derived.

Notation:

AMT— current balance of the Attention Mining Treasury in SOL currency, after the last fee claim.

MC — current market cap of CAP (DEX price × circulating supply).

$p \in (0,1)$ — fraction of the EMT balance used per PoA block (prototype: p = 0.05, i.e., 5%).

B_n — PoA block budget for block n.

$(\lambda MC) \in [0, 1]$ — fraction of the block budget directed into liquidity, as a function of market cap.

sTarget — target engagement score for the current raid.

tMax — time limit for the current raid (starts at 20 minutes and is adjusted over time).

3. What is a PoA block

Each PoA block is one full cycle:

Each PoA block is one full cycle: ButtAI posts a butt image → raid starts → community pushes engagement → if target hit, rebuy executes and gets etched as a block linked to that image.

- Post: AI generates/remixes a butt visual (models, celebs, anon, etc.), posts on official X with raid call.
- Raid: Time-boxed window (e.g., 20 min initial). Engagement scored: likes (w1), replies (w2), reposts/quotes (w3). Total S = sum(wi * ci).
- Success: If S ≥ Starget in Tmax, claim pump.fun rewards, compute buy size proportional to rewards + attention, rebuy BUTT, pair with image/metrics as "Block #N" in the chain. Failed raids skip — no buy, no block.

Engagement scoring:

Each action type has a fixed weight: like — wlike, reply — wreply, repost — wrepost, quote/meme reply — wquote, etc.

During the raid, the bot counts how many actions of each type were made under the tweet and computes a single score: $\sum S = i\ wi \cdot ci$

where ci is the number of actions of type i.

Time window and success:

Each raid has a time window Tmax. Initially:

$$Tmax = 20 \text{ minutes.}$$

A PoA block is mined (successfully closed) if, within that window,

$$S \geq Starget$$

If the window closes and the score is still below the target, the block is failed: no buyback, no liquidity add, no treasury spend for that block.

4. Dynamic difficulty: not too easy, not impossible

We want raids to require effort, but not be impossible. So difficulty adapts after every raid, based on a simple rule:

- if the community failed — make the next raid easier and slower;
- if the community succeeded — make the next raid harder and faster.

Two parameters move: engagement target Starget, time limit Tmax.

If a raid fails: If S < Starget after Tmax:

$$S_{target,new} = S_{target,old} \cdot \alpha_{\downarrow}, \ \alpha_{\downarrow} \in (0,1)$$

$$T_{max,new} = T_{max,old} \cdot \beta_{\uparrow}, \ \beta_{\uparrow} > 1$$

If a raid succeeds: If $S \geq S_{target}$ within Tmax:

$$S_{target,new} = S_{target,old} \cdot \alpha_{\uparrow}, \ \alpha_{\uparrow} > 1$$

$$T_{max,new} = T_{max,old} \cdot \beta_{\downarrow}, \ \beta_{\downarrow} \in (0,1)$$

Coefficients $\alpha_{\downarrow}, \alpha_{\uparrow}, \beta_{\uparrow}, \beta_{\downarrow}$ are system parameters and can be clamped so that difficulty doesn't jump insanely from one raid to the next.

5. Fees and block budget

Treasury only moves on successful blocks.

Claiming pump.fun dev fees:

Once a PoA block is mined ButtAI:

- Calls the dev-fee claim on pump.fun.
- All newly claimed dev fees are sent to the EMT.
- After that, the updated treasury balance is FEMT.

No other inflows are assumed at the protocol level.

PoA block budget:

For this block n, the protocol allocates a fixed fraction of the current treasury:

$$B_n = p \cdot FEMT$$

with p = 0.05 (5%) by default.

This is the only amount allowed for buyback and liquidity operations for PoA block.

6. How much goes into liquidity vs pure buyback

Behavior should change with market cap:

- small BUTT — more raw buyback
- pressure; larger BUTT— more liquidity and stability.

We use a function λ(MC) that tells us what fraction of Bn goes into liquidity.

Example step function (numbers can be tuned later):

$$\lambda(MC) = \begin{cases} 0.2, & MC < 100{,}000 \\ 0.5, & 100{,}000 \le MC < 1{,}000{,}000 \\ 0.7, & 1{,}000{,}000 \le MC < 10{,}000{,}000 \\ 1.0, & MC \ge 10{,}000{,}000 \end{cases}$$

From here:

$$B_n^{LP} = \lambda(MC) \cdot Bn$$

$$B_n^{pump} = (1 - \lambda(MC)) \cdot Bn$$

At low MC, B_n^{pump} dominates;athigh MC BLP_ndominates.

7. On-chain actions: buyback and LP

Once B_n is known, execution is deterministic.

Buyback:

First, the whole block budget is used to buy BUTT:

$$Bn \xrightarrow{swap} BUTT\ bought$$

via the BUTT/SOL pool on the DEX (e.g., BUTT/SOL on PumpSwap), at the current market price.

Split of acquired BUTT:

The acquired amount $BUTT_{bought}$ is conceptually split according to B_n^{LP} and B^{pump}:

pool. LP tokens from this operation are held by the Engagement Mining Treasury.

The remaining part $B\ pump_n$ can:

– stay in EMT as BUTT removed from the open market (pure buyback and hold),
– be distributed to raid participants as PoA rewards,
– be partially burned or used in other on-chain mechanics defined elsewhere in the whitepaper.

The split between liquidity and pump is entirely determined by the formulas and not chosen by hand per block.

8. Dynamic Difficulty

Raids adapt: success ramps Starget up/time down; failure eases them. Converges to sustainable pace where attention reliably mines blocks, extending the buttcoin chain. Each block's rebuy size grows with treasury from prior fees/attention

9. Visual & Attention Mechanics

Butts as infrastructure: broad, remixable (AI/user edits stay "butt"-identifiable), algorithm-friendly (high-engagement theme, compression-robust). PoA locks raids to butt posts, training feeds on the pattern. Early: PoA drives chain growth. Later: user butts sustain it, treasury stabilizes via rule-based buys.

10. What this achieves

- Treasury spending is gated by engagement: No block — no spend. EMT only moves when the community actually hits the raid goals.
- pump.fun fees are used by rule, not by mood: All dev fees flow into EMT and are consumed
- according to $B_n = p \cdot F_{EMT}$ and λ(MC). The rule is public and doesn't change per block.
- Difficulty follows the community: If raids are trivial, targets go up and time goes down. If raids are too hard, they relax. Over time this should converge to a natural tempo of PoA blocks.
  Early = wild, later = stable: At low market cap more budget acts as pure buyback. As the cap grows, more goes into liquidity. The system naturally shifts from "degen meme" dynamics into something more robust. Symbolic "blockchain-like" reporting: PoA doesn't try to re-implement a
- full blockchain. Instead,
  it adds a light "block log": numbered blocks, public summaries in Telegram and X, and clearly verifiable on-chain transactions behind each block.

11. Feedback into future block strength

Successful waves of attention are not only visible in social metrics; they also tend to correlate with higher trading activity and higher dev fees. Since all dev fees accumulate in the Engagement Mining Treasury and each PoA block spends a fixed fraction of that balance, a period of strong virality directly increases the absolute size of subsequent buybacks and liquidity adds. In other words, attention does not stop at the tweet level: it increases the capacity of future PoA blocks, linking social performance and on-chain reaction into a repeating feedback loop.

Conclusion

buttcoin has viral mechanics baked into its core: a degenerate-yet-repeatable visual theme (butts) that thrives in AI-edited meme factories, paired with PoA that turns social raids into a deterministic chain of rebuy blocks. No discretionary dev spends — just attention farming that lengthens the longest ongoing Proof of Attention buttcoin chain, with each butt-marked buy pulling from pump.fun rewards scaled by X engagement. This formalizes virality as code, not hype, creating a self-reinforcing loop where community focus directly builds treasury strength and on-chain history. The mission: extend this chain to 1B market cap (10M anon launch target), proving attention can mine harder than hashpower in 2026's feed wars