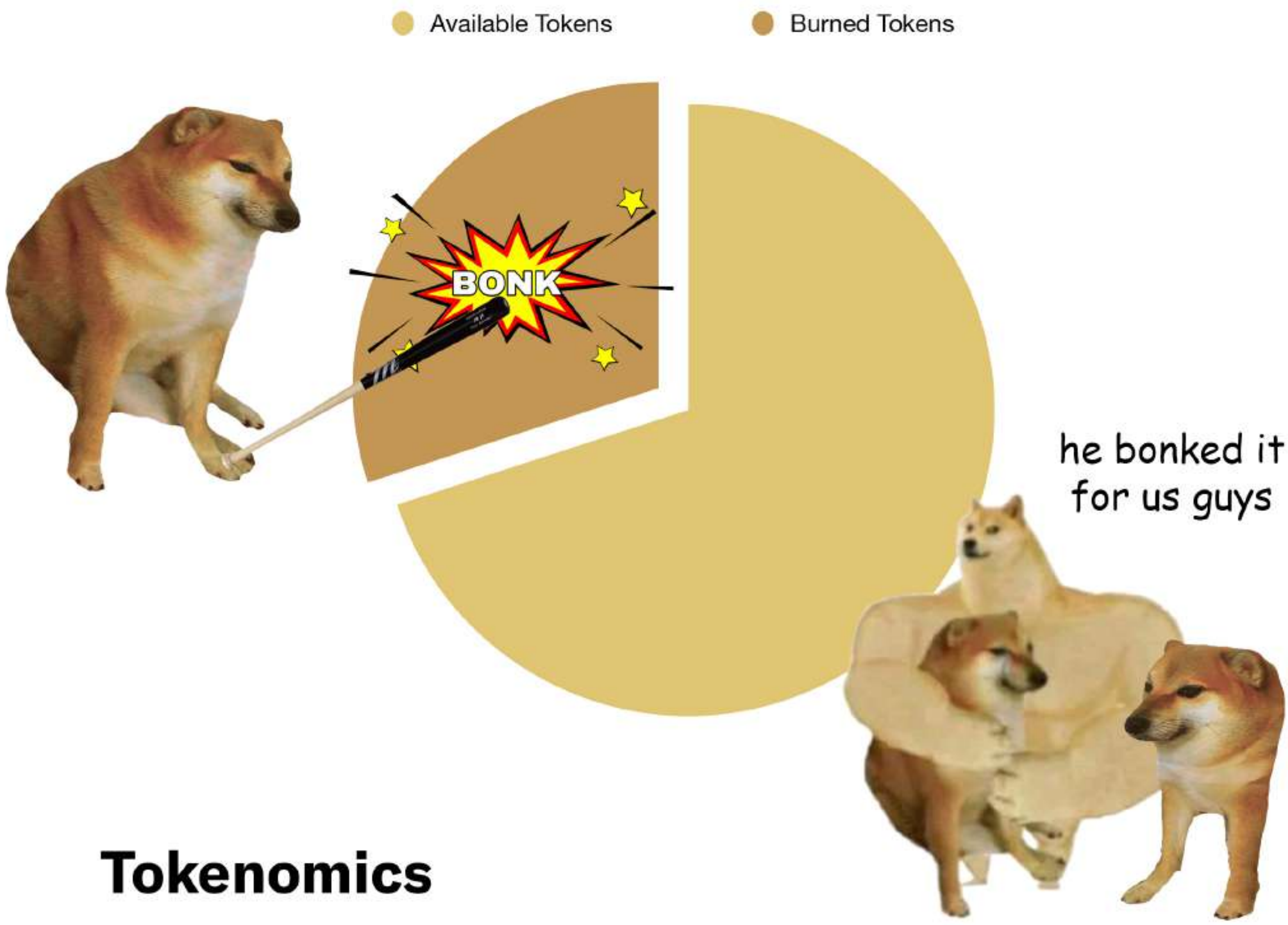
FOR BONKING USE ONLY

DogeBonk Whitepaper

The Memetics of Bonking

In an era of peak corporate nihilism and social imbalance, the simple gesture of bonking, as unelegant and unpolished it sounds, stands on its own amidst a sea of memecoin clones. What else is as crude and pedestrian as a smug, simple dog with a baseball bat? **DogeBonk** has captured the value of a memetic concept in the form of the $**DOBO** token, giving all holders a strong incentive to 'meme' their token into collective conciousness.

Through simple jpeg edits of already existing memes to film and classical art references, the **DogeBonk** meme became a flurry of original and organic content not seen since 2016's memetic arc. At the moment of writing, it has already evolved further, and its limits know no reach. The rare ocurrance that a landscape sapped of authenticity and creativity yearns for. This is **DogeBonk**, a memecoin crafted by strangers across the internet, changing lives.



Tokenomics

30% of the initial token supply was burned. The contract was fairly launched with no presale and the ownership is renounced. DOGEBONK is a 100% decentralized community token.

Total Supply: 1,000,000,000,000,000
Initial Burn: 300,000,000,000,000
Available Supply: and 700,000,000,000,000

DOGEBONK Protocol

DOGEBONK employs two simple functions: Reflection + liquidity pool acquisition. In each trade, the transaction is taxed a 10% fee, which is split two ways.

- 5% fee = redistributed to all existing holders.
- 5% fee = added to the liquidity pool.

Reflection Rewards

In a static reward system, the reward amount is dependent upon the volume of the token being traded. This goal is to alleviate downward sell pressure put on the token caused by earlier adopters selling their tokens.

DOGEBONK is a deflationary token utilizing static rewards and reflections to both incentivize holding and discourage swing trading. A 10% tax on every transaction made is taken of which 5% is sent to the liquidity pool and 5% is 'reflected' to token holders as static rewards for continuous holding. This consistent adding to the liquidity pool reduces volatility over time, whilst the static reflections both enrich holders and reduce the supply

Liquidity Pool (Permanently Locked)

First, the contract receives tokens both sellers and buyers alike. These tokens are then added to the liquidity pool creating a solid price floor. 5% of every transaction made is added to the liquidity pool. This liquidity pool is locked forever and can not be abused by anyone.

Second, this tax acts as a resistance mechanism that secures the volume of DOGEBONK as a reward for the holders. In theory, the added liquidity pool creates a stability from the supplied liquidity pool by adding the tax to the overall liquidity of the token, thus increasing the tokens overall liquidity pool and supporting the price floor of the token.

As the DOGEBONK token liquidity pool increases, the price stability mirrors this with the benefit of a solid price floor and cushion for holders. The goal here is to prevent the larger dips when whales decide to sell their tokens later in the game, which keeps the price from fluctuating as much as if there was no automatic liquidity pool function.

All of this is an effort to alleviate some of the troubles we have seen with the current DeFi reflection tokens. We are confident that this model and protocol will prevail over the outdated reflection tokens for these reasons.

The DOGEBONK liquidity pool is also permanently locked. Meaning that the developers can not touch it. It will remain in liquidity in perpetuity. This along with the renounced contract makes DOGEBONK a truly unique and decentralized community built token.

Burn

Coin burns directly affect the dynamics of supply and demand. The most notable objective is to create a deflationary effect.

DOGEBONK initially burned 300,000,000,000,000 tokens (30%) of the supply. The current burn wallet earns reflections thus continuously burning small amounts of the token supply.

As time goes on, the burn wallet continues to burn the small amount it receives in reflections, thus contributing a continuous and automatic burn of tokens. This mechanism reduces the available supply and in theory increases each tokens value held by users.