

# Auto Mining Token

## AMT

### Whitepaper

16 de julho de 2021

**Resumo:** Existem várias técnicas sobre as quais tem-se tentado tokenizar a mineração de BitCoin (BTC). Nossa aposta é criar um mecanismo único e simples para o usuário, permitindo que ele mantenha seus tokens em todos os momentos e os negocie sempre que desejar. Combinando a técnica de snapshot-token com a velocidade, segurança e baixo custo de transação do BSC, poderemos criar um ecossistema escalável, transparente e lucrativo para os investidores.

#### Introdução

O Auto Mining Token é um ecossistema que permite, através de seu token (AMT), investir em projetos de mineração de BTC. Os usuários que possuem AMT poderão cobrar a lucratividade gerada pelo projeto com base na quantidade de AMT que possuem em todos os tokens em circulação. Ou seja: se uma carteira possui, por exemplo, 50% de todos os tokens em circulação, ela cobrará 50% do retorno pago aos titulares.

(Produção de BTC - custos operacionais) → 50% Lucro Direto

→ 50% Cofre de Garantia

Além disso, metade do retorno recebido diariamente pelo titular do token (após a dedução dos custos) será colocado em um contrato inteligente que funcionará como um cofre. O cofre sempre permitirá que os detentores de AMT liquidem seus tokens a uma taxa determinada pelo BTC no cofre sobre a totalidade dos AMT. Dessa forma, será gerado um lastro progressivo sobre o AMT, que só poderá ser extraído da liquidação do token. Todos os tokens liquidados no cofre de garantia serão queimados e retirados de circulação para sempre.

Poderão existir um **máximo de 100 milhões de AMT** que serão emitidos ao longo do tempo em função da aquisição de maior poder de mineração. O projeto começará a operar lastreado em aproximadamente 70.000 TH/s com uma emissão AMT calculada para que a rentabilidade seja interessante para o investidor, e para que o minerador possa tolerar o impacto da diminuição de seu poder de hash (para benefício do investidor do token) até que novos equipamentos sejam adquiridos de forma a aumentar a lucratividade para os

investidores iniciais, e para que o minerador possa tolerar o impacto da diminuição de seu poder de hash (em benefício do investidor do token) até que novos equipamentos sejam adquiridos de forma a aumentar a lucratividade para os investidores iniciais e para o minerador que contribui com o hash power inicial, compartilhando o novo poder de hash introduzido no ecossistema.

### **Transparência e poder de hash**

Todas as nossas equipes se conectarão a um pool de mineração aberto com observadores públicos que colocaremos à disposição online. Dessa forma, será possível comparar diariamente qual é o benefício que todo o projeto está gerando. Da mesma forma, será notificado quando novos equipamentos forem conectados e, também, será notificado com antecedência a emissão de novos tokens..

Todo o esquema do projeto contempla os custos de administração, energia e amortização com os tokens que o próprio projeto manterá. Essas despesas serão tratadas de forma transparente e os detentores de AMT poderão participar das decisões por meio de votação.

Emissão Inicial de AMT → 75%	Minerador contribuinte do poder de hash
→ 20%	Market place (1 AMT = 1 USD)
→ 5%	Administração e Marketing

Além disso, auditores externos e independentes serão incorporados para garantir o bom funcionamento dos Contratos Inteligentes e a segurança dos mesmos.

Exposição à mineração de BITCOIN, liquidez e oportunidade para trading

Este projeto adicionará ao ecossistema geral de exposição à mineração um novo mecanismo de cobrança e aumentará a liquidez geral. Ao contrário de outros projetos, não será necessário colocar os tokens em um Smart Contract ou bloqueá-los por um determinado tempo para receber os benefícios. Além disso, será criado um pool de liquidez no Pancake Swap e os tokens de liquidez serão bloqueados por um período de 2 anos. Com um volume de 1.000.000 de AMT e o equivalente em BTCB de 1.000.000 USDT no momento do lançamento, para garantir um valor inicial em paridade com o dólar americano. O bloqueio será realizado a partir de um Smart Contract derivado do TimeLock tradicional, agregando a funcionalidade de poder coletar rentabilidade sobre os tokens bloqueados. Desta forma, garantimos o desenvolvimento contínuo do projeto e o compromisso com o mesmo. Soma-se à exposição à mineração em geral, a exposição ao desenvolvimento do projeto de 2 formas:

1. Continuamente se aumentará o lastro natural do AMT contra BTC através do cofre que recebe parte dos rendimentos. Sempre garantindo uma saída para todos os holders, que poderá ser, dependendo do momento e do preço pelo qual o token foi comprado, com prejuízo ou lucro. Mas sempre com a possibilidade de retirada com parte dos lucros gerados ao longo da história do projeto.

2. Melhorar a relação AMT/HashPower com novos investimentos em infraestrutura. O objetivo será sempre garantir aos detentores de AMT uma melhor relação de poder de hash em relação aos novos que adquiram os emitidos recentemente. Esse sistema continuará até atingir o ponto de estabilidade de 100 milhões de tokens circulantes, onde as definições sobre os novos investimentos a serem feitos começarão a ter muito mais peso.

Em todos os sentidos, devido à operação que desenvolvemos em território latino-americano, conseguimos atingir custos extremamente competitivos de energia, mão de obra e aquisição de equipamentos. Com a união de muitas mineradoras da região, nosso token oferecerá melhores resultados do que os atuais no mercado.

### **Blockchain - BNB Smart Chain**

A BNB Smart Chain (antiga Binance Smart Chain) provou ter capacidade, segurança e confiança do público em geral para hospedar grandes projetos. Nesse sentido, temos o compromisso de utilizá-la para ter a melhor relação entre custos de transação do usuário, compatibilidade com carteiras, integração com outros dAPPs e a possibilidade de utilização do BBTC como representação do BitCoin dentro desta rede.

### **Wrapped BitCoin - BTCB**

BTCB é um ativo BEP2/BEP20 definido (fixado) na Binance Chain/Binance Smart Chain com uma relação 1:1 a BTC bloqueada no Bitcoin Blockchain. O BTCB funciona em um modelo centralizado e baseado em confiança. A relação 1:1 significa que a quantidade de BTCB encapsulado é igual à quantidade de BTC bloqueada em um endereço público. O modelo centralizado e baseado em confiança implica que o emissor dos tokens BTCB encapsulados seja a Binance.

A transparência é garantida por meio da página da web Asset Proof (<https://www.binance.org/en/assets-proof>), onde o fornecimento atual de todos os ativos bloqueados e emitidos no blockchain pode ser verificado publicamente. A quantidade de BTC bloqueado pode não ser exatamente a mesma que a quantidade de BTC encapsulado, porque esses dados de auditoria não são atualizados em tempo real, mas são processados semanalmente.

Assim que os BTCs forem minerados, usaremos uma bridge da Binance para gerar os respectivos BBTCs e usá-los para pagar os rendimentos. Além disso, garantimos a existência de carteiras que terão lastro em BBTC para que, em caso de possíveis falhas de bridge, o pagamento correspondente possa ser feito dentro do prazo determinado.

### **Aspectos técnicos do token AMT**

O AMT será desenvolvido em conformidade com o padrão ERC-20 (BEP-20 dentro do BSC). Dessa forma, garantimos sua portabilidade com qualquer carteira e aplicativos externos que os reconheçam.

Será adicionada a funcionalidade de snapshot usando a derivação do padrão desenvolvido por Open Zeppelin

(<https://github.com/OpenZeppelin/openzeppelin-contracts/blob/master/contracts/token/ERC20/extensions/ERC20Snapshot.sol>)

Esta variante permite definir momentos específicos (snapshots) em que o saldo de uma carteira pode ser consultado retroativamente. Esses momentos serão determinados pelas operações do administrador privilegiado na mesma operação que efetua o pagamento dos rendimentos. Desta forma, garante-se que a cobrança pode ser feita exclusivamente e apenas uma vez aos titulares de AMT no momento do pagamento. Até mesmo o valor total de AMT em circulação será registrado no snapshot, de forma tal que os titulares possam cobrar com base na proporção específica de seu AMT em função do total do momento do pagamento, para que seus tokens não percam valor com lançamentos futuros.

### **Geração de rentabilidade para provedores de liquidez**

Devido a possibilidade de prover liquidez em mercados descentralizados será um direito público de todos os titulares. E já que o contrato do pool de liquidez é quem tem os tokens nesse caso, implementamos um sistema pelo qual mesmo quem está disponibilizando liquidez no Pancake Swap pode cobrar os rendimentos gerados pelo AMT.

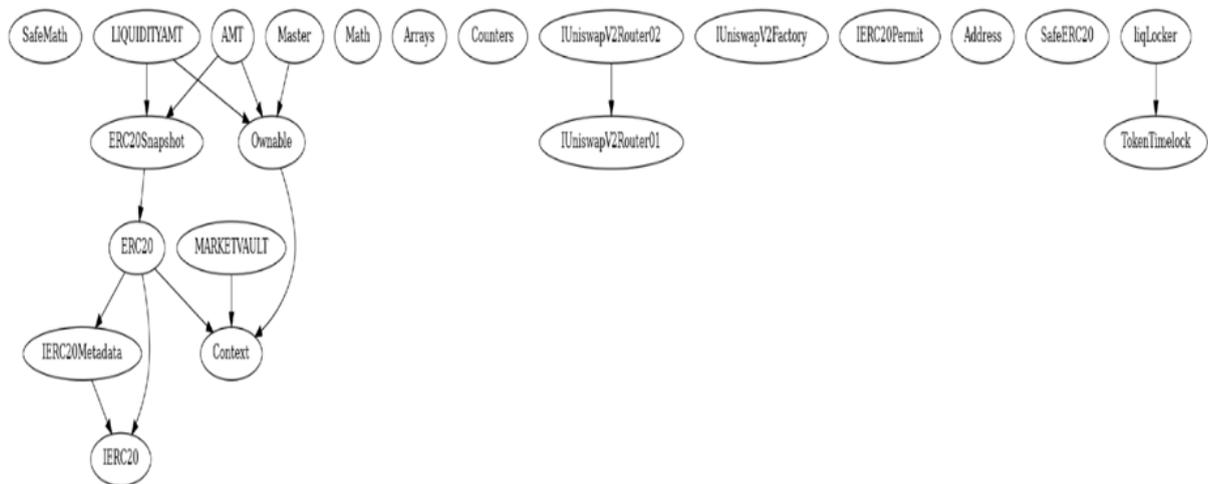
Será implementado um contrato que mascara o processo de provisão de liquidez permitindo que aqueles que fornecem através de nossa interface obtenham, em vez do token de liquidez LP original, um token auxiliar chamado liqAMT emparelhado na relação 1:1 com o respectivo token LP. O contrato protegerá o token LP e o utilizará quando o usuário decidir retirar a liquidez, queimando os liqAMT emitidos e devolvendo os tokens do pool representados pelo LP.

### **Mercado e cofre**

As vendas iniciais do token serão realizadas por meio de um Smart Contract criado especificamente para esse fim, que também funcionará como cofre de garantia para o AMT, recebendo seu respectivo percentual por cada pagamento de devolução efetuado. Assim, o processo de vendas será 100% transparente.

Além disso, o cofre receberá sistematicamente BBTC a cada pagamento, entregando-os apenas aos detentores de AMT caso queiram liquidar sua posição em uma relação estabelecida pelo *saldo BBTC do cofre : AMT liquidado*. Queimando os AMTs no processo. Desta forma, a relação estabelecida só pode ir aumentando, garantindo sempre um preço mínimo de liquidação aos usuários que só aumenta ao longo do tempo. Implementamos este sistema porque acreditamos firmemente no BTC e promovemos o seu desenvolvimento e entendemos que é essencial que os investidores tenham um mecanismo de lastro transparente. Não haverá outra forma de retirada do BBTC do cofre além da estabelecida mediante a liquidação do AMT.

## Estrutura



## Integração de mineradores externos

Criaremos um sistema pelo qual mineradores que não estão em nossos estabelecimentos poderão participar do projeto. Em virtude do poder de hash fornecido, será garantido um aumento no poder de hash de 5%, devido à capitalização obtida pela venda dos tokens emitidos com a garantia desta contribuição externa, poderemos usá-lo para:

1. Amortizar o poder de hash fornecido no caso de uma possível desconexão os AMT gerado continuarão respaldados.
2. Aquisição de novos equipamentos para melhorar a relação geral entre AMT:HashPower

Desta forma, enquanto nossa capacidade energética nos permitir, poderemos transferir o investimento em mineração de BTC de áreas menos rentáveis para áreas mais rentáveis. Permitindo ao minerador externo uma transição mais suave e lucrativa do que a mineração em um local menos lucrativo.

## Conclusões e objetivos finais

Embora atualmente seja impossível gerar um sistema de tokenização de mineração BitCoin 100% descentralizado, nosso ecossistema tenta alcançar a máxima transparência que o atual estado da arte da tecnologia blockchain permite.

Estamos aqui diante de uma integração de várias metodologias aplicadas por outros projetos separadamente para construir um ecossistema próspero em um ambiente competitivo atormentado pela volatilidade do mercado.

Nosso objetivo no final do dia é construir sistematicamente a confiança do usuário no AMT para alcançar não apenas sua avaliação, mas também um crescimento sustentável ao longo do

tempo da mineração em territórios lucrativos. Contribuindo com o investimento obtido através da tokenização uma escala comercial que lhe permite competir firmemente com o resto das ofertas do mercado.

## Anexo

### *1- Estrutura dos Smart Contracts*

- Smart Contracts

- ○ Próprios (serão verificados no bscscan no momento da implantação)
  - AMT.sol
  - LIQUIDITYAMT.sol ■ liqLocker.sol
  - MARKETVAULT.sol ■ Master.sol
- ○ Importados (open zeppelin github) ■ ERC20
  - ERC20-snapshot ■ TokenTimelock ■ Ownable
- ○ Importado (Uniswap liquidity connections) ■ IUniswapV2Router02

### *2- Visão geral das funções*

*(\$)* = payable function

*#* = non-constant function

*Int* = Internal

---

*Ext* = External

*Pub* = Public

+ [*Lib*] SafeMath

- [*Int*] tryAdd

- [*Int*] trySub

- [*Int*] tryMul

- [*Int*] tryDiv

- [*Int*] tryMod

- [*Int*] add

- [Int] sub
- [Int] mul
- [Int] div
- [Int] mod
- [Int] sub
- [Int] div
- [Int] mod
- + [Int] IERC20
  - [Ext] totalSupply
  - [Ext] balanceOf
  - [Ext] transfer #
  - [Ext] allowance
  - [Ext] approve #
  - [Ext] transferFrom #
- + [Int] IERC20Metadata (IERC20)
  - [Ext] name
  - [Ext] symbol
  - [Ext] decimals
- + Context
  - [Int] \_msgSender
  - [Int] \_msgData
- + ERC20 (Context, IERC20, IERC20Metadata)
  - [Pub] #
  - [Pub] name
  - [Pub] symbol
  - [Pub] decimals
  - [Pub] totalSupply
  - [Pub] balanceOf
  - [Pub] transfer #
  - [Pub] allowance

- [Pub] approve #
- [Pub] transferFrom #
- [Pub] increaseAllowance #
- [Pub] decreaseAllowance #
- [Int] \_transfer #
- [Int] \_mint #
- [Int] \_burn #
- [Int] \_approve #
- [Int] \_spendAllowance #
- [Int] \_beforeTokenTransfer #
- [Int] \_afterTokenTransfer #
- + [Lib] Math
  - [Int] max
  - [Int] min
  - [Int] average
  - [Int] ceilDiv
  - [Int] mulDiv
  - [Int] mulDiv
  - [Int] sqrt
- [Int] sqrt
- + [Lib] Arrays
  - [Int] findUpperBound
- + [Lib] Counters
  - [Int] current
  - [Int] increment #
  - [Int] decrement #
  - [Int] reset #
- + ERC20Snapshot (ERC20)
  - [Int] \_snapshot #
  - [Int] \_getCurrentSnapshotId

- [Pub] *balanceOfAt*
- [Pub] *totalSupplyAt*
- [Int] *\_beforeTokenTransfer* #
- [Prv] *\_valueAt*
- [Prv] *\_updateAccountSnapshot* #
- [Prv] *\_updateTotalSupplySnapshot* #
- [Prv] *\_updateSnapshot* #
- [Prv] *\_lastSnapshotId*
- + *Ownable (Context)*
  - [Pub] #
  - [Pub] *owner*
  - [Int] *\_checkOwner*
  - [Pub] *renounceOwnership* #
    - *modifiers: onlyOwner*
  - [Pub] *transferOwnership* #
    - *modifiers: onlyOwner*
  - [Int] *\_transferOwnership* #
- + *AMT (ERC20Snapshot, Ownable)*
  - [Pub] #
    - *modifiers: ERC20*
  - [Pub] *mint* #
    - *modifiers: onlyOwner*
  - [Pub] *snapshot* #
    - *modifiers: onlyOwner*
  - [Pub] *getCurrentSnapshotId*
  - [Pub] *burn* #
  - [Pub] *burnFrom* #
- + [Int] *IUniswapV2Router01*
  - [Ext] *factory*
  - [Ext] *WETH*

- [Ext] *addLiquidity* #
- [Ext] *addLiquidityETH* (\$)
- [Ext] *removeLiquidity* #
- [Ext] *removeLiquidityETH* #
- [Ext] *removeLiquidityWithPermit* #
- [Ext] *removeLiquidityETHWithPermit* #
- [Ext] *swapExactTokensForTokens* #
- [Ext] *swapTokensForExactTokens* #
- [Ext] *swapExactETHForTokens* (\$)
- [Ext] *swapTokensForExactETH* #
- [Ext] *swapExactTokensForETH* #
- [Ext] *swapETHForExactTokens* (\$)
- [Ext] *quote*
- [Ext] *getAmountOut*
- [Ext] *getAmountIn*
- [Ext] *getAmountsOut*
- [Ext] *getAmountsIn*
- + [Int] *IUniswapV2Router02* (*IUniswapV2Router01*)
  - [Ext] *removeLiquidityETHSupportingFeeOnTransferTokens* #
  - [Ext] *removeLiquidityETHWithPermitSupportingFeeOnTransferTokens* #
  - [Ext] *swapExactTokensForTokensSupportingFeeOnTransferTokens* #
  - [Ext] *swapExactETHForTokensSupportingFeeOnTransferTokens* (\$)
  - [Ext] *swapExactTokensForETHSupportingFeeOnTransferTokens* #
- + [Int] *IUniswapV2Factory*
  - [Ext] *feeTo*
  - [Ext] *feeToSetter*
  - [Ext] *getPair*
  - [Ext] *allPairs*
  - [Ext] *allPairsLength*
  - [Ext] *createPair* #

- [Ext] setFeeTo #
- [Ext] setFeeToSetter #
- + LIQUIDITYAMT (ERC20Snapshot, Ownable)
  - [Pub] #
    - modifiers: ERC20
  - [Pub] mint #
    - modifiers: onlyOwner
  - [Pub] snapshot #
    - modifiers: onlyOwner
  - [Pub] getCurrentSnapshotId
  - [Pub] burn #
  - [Pub] burnFrom #
- + [Int] IERC20Permit
  - [Ext] permit #
  - [Ext] nonces
  - [Ext] DOMAIN\_SEPARATOR
- + [Lib] Address
  - [Int] isContract
  - [Int] sendValue #
  - [Int] functionCall #
  - [Int] functionCall #
  - [Int] functionCallWithValue #
  - [Int] functionCallWithValue #
  - [Int] functionStaticCall
  - [Int] functionStaticCall
  - [Int] functionDelegateCall #
  - [Int] functionDelegateCall #
  - [Int] verifyCallResult
- + [Lib] SafeERC20
  - [Int] safeTransfer #

- [Int] *safeTransferFrom* #
- [Int] *safeApprove* #
- [Int] *safeIncreaseAllowance* #
- [Int] *safeDecreaseAllowance* #
- [Int] *safePermit* #
- [Prv] *\_callOptionalReturn* #
- + *TokenTimelock*
  - [Pub] #
  - [Pub] *token*
  - [Pub] *beneficiary*
  - [Pub] *releaseTime*
  - [Pub] *release* #
- + *liqLocker (TokenTimelock)*
  - [Pub] #
    - *modifiers: TokenTimelock*
  - [Pub] *charge* #
  - [Pub] *release* #
- + *Master (Ownable)*
  - [Pub] #
  - [Pub] *addressLiquidityPool*
  - [Pub] *addressLiquidityLocker*
  - [Pub] *payRent* #
    - *modifiers: onlyOwner*
  - [Pub] *charge* #
  - [Pub] *liqCharge* #
  - [Pub] *addLiquidityLocking* #
    - *modifiers: onlyOwner*
  - [Pub] *addLiquidity* #
  - [Pub] *removeLiquidity* #
  - [Pub] *mintMaster* #

- *modifiers: onlyOwner*

+ *MARKETVAULT (Context)*

- *[Pub] #*

- *[Pub] getBackRate*

- *[Pub] backingWithdrawal #*

- *[Pub] buy #*