Margin Token Audit Information

1. Margin Token Introduction

Margin Tokens allow for anyone to gain exposure to a dual momentum trading system acting on a portfolio of tokens, simply by holding the Margin Token in their Metamask wallet. Each Margin Token is 100% backed by its underlying assets. To purchase a Margin Token, supply one of the assets in the portfolio to the Margin Token pool. Send a Margin Token back to the pool to redeem a portfolio token of your choice.

If a token, such as BNB, is in a strong uptrend and therefore desired in the Margin Token portfolio, there will be a reward for supplying this asset and a fee for redeeming it. If the market turns and BNB is no longer desired, there will be a reward for redeeming this asset and a fee for supplying it. This creates a potential arbitrage opportunity that incentivizes market participants to keep the pool's allocations consistent with the dual momentum trading strategy.

An automated dual momentum trading system will update the deposit and redeem fees or rewards based on its strategy. Rather than the pool of assets incurring trading fees internally by interacting with a DeFi protocol such as PancakeSwap, the fees are passed on to the users via the rewards for depositing and redeeming assets as desired.

Margin Tokens have the ability to supply assets to Venus, and use that as collateral to borrow other assets such as USDT which can be swapped for assets such as BTC to gain added exposure to it.

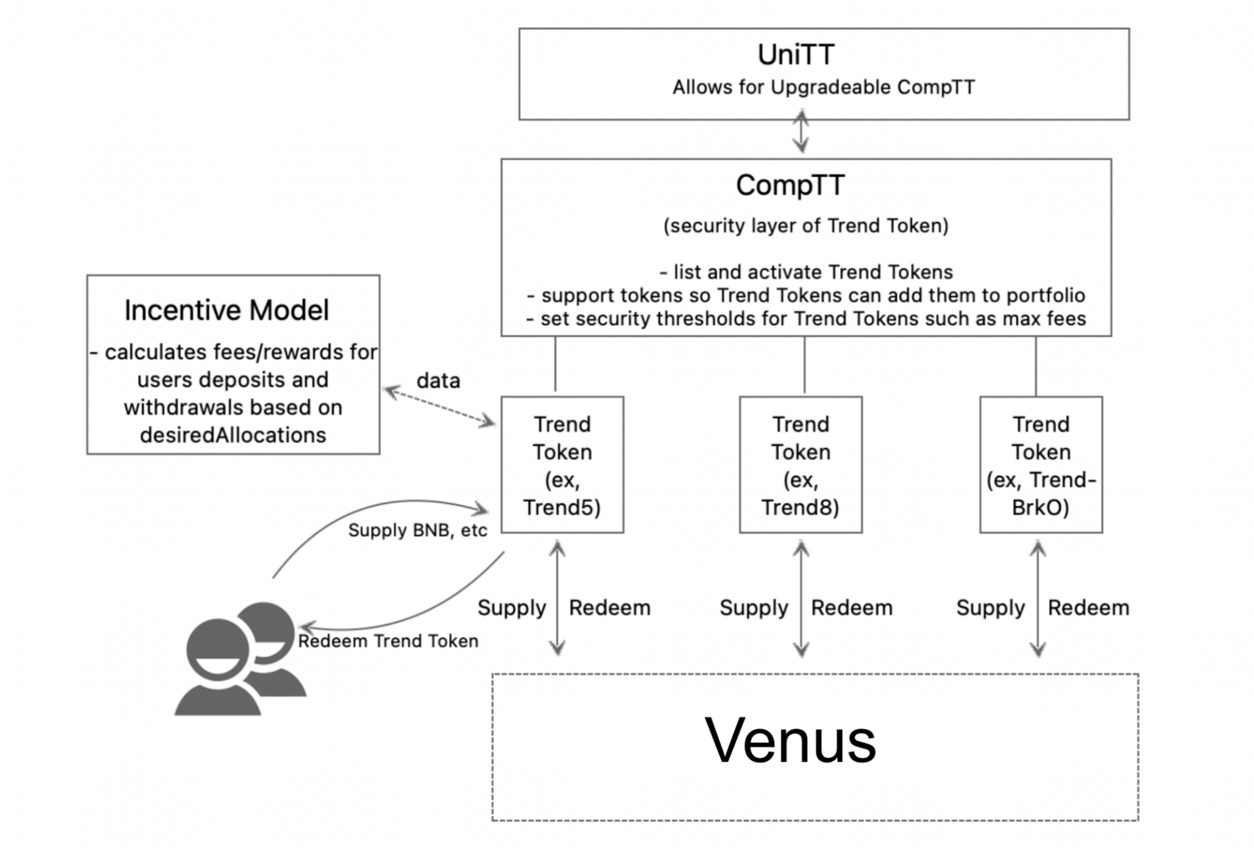
2. Portfolio Options

There will be types of TreMarginnd Tokens, each with a different portfolio theme. This is similar to the different types of portfolio options offered for Trendbot and Marginbot. To begin, three Trend Tokens will be offered.

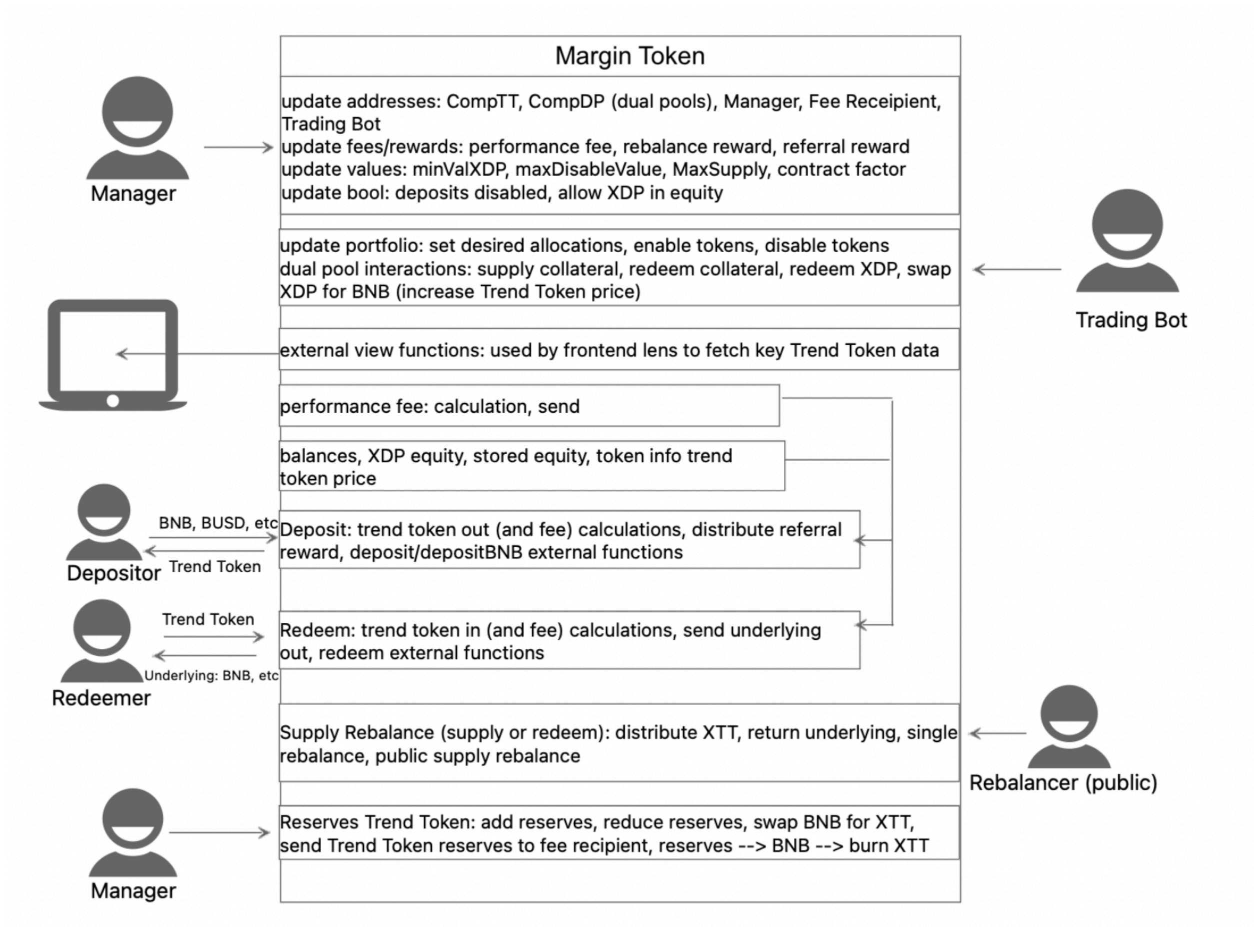
1) **MARGIN5:** Portfolio of USDT, BNB, and 3 top tokens by liquidity on PancakeSwap.
2) **MARGIN8:** Portfolio of USDT, BNB, and 5 top tokens by liquidity on PancakeSwap.
3) **MARGIN-BrkO:** Portfolio of USDT, BNB, and 0-5 tokens making new all time highs out of the highest 10-20 liquid tokens on PancakeSwap.

3. Architecture

3.1 Overall Architecture


UniTT
Allows for Upgradeable CompTT
CompTT
(security layer of Trend Token)
- list and activate Trend Tokens
- support tokens so Trend Tokens can add them to portfolio
- set security thresholds for Trend Tokens such as max fees
Incentive Model
- calculates fees/rewards for users deposits and withdrawals based on desiredAllocations
data
Trend Token (ex, Trend5)
Trend Token (ex, Trend8)
Trend Token (ex, Trend-BrkO)
Supply BNB, etc
Redeem Trend Token
Supply
Redeem
Supply
Redeem
Supply
Redeem
Venus

3.2 Margin Token Architecture


Margin Token
update addresses: CompTT, CompDP (dual pools), Manager, Fee Receipient, Trading Bot
update fees/rewards: performance fee, rebalance reward, referral reward
update values: minValXDP, maxDisableValue, MaxSupply, contract factor
update bool: deposits disabled, allow XDP in equity
update portfolio: set desired allocations, enable tokens, disable tokens
dual pool interactions: supply collateral, redeem collateral, redeem XDP, swap XDP for BNB (increase Trend Token price)
external view functions: used by frontend lens to fetch key Trend Token data
performance fee: calculation, send
balances, XDP equity, stored equity, token info trend token price
Deposit: trend token out (and fee) calculations, distribute referral reward, deposit/depositBNB external functions
Redeem: trend token in (and fee) calculations, send underlying out, redeem external functions
Supply Rebalance (supply or redeem): distribute XTT, return underlying, single rebalance, public supply rebalance
Reserves Trend Token: add reserves, reduce reserves, swap BNB for XTT, send Trend Token reserves to fee recipient, reserves --> BNB --> burn XTT
Manager
Trading Bot
BNB, BUSD, etc
Trend Token
Depositor
Trend Token
Underlying: BNB, etc
Redeemer
Rebalancer (public)
Manager

4.0 Deploy and Management Instructions

4.1 Comptroller Deployment

Governs all deployed Trend Tokens and ensures safety parameters

1. Deploy CompTT and UniTT:
   Result compTT: 0x673a9ad7ae3aa076fffc1ecd194a62fb31bf4a63
   Result UniTT: 0x8ce9443b7a6baed8151be38d074e2940bfe158cd
2. Configure CompTT and UnitTT
   - _setPendingImplementation() in UniTT using CompTT address
   - _updateLockedState() in CompTT to ‘false’
   - _become() in CompTT using UniTT address
   - use UniTT address and CompTT ABI
3. Deploy ChainlinkOracle.sol
   input: 1000000000000000000,0xae13d989daC2f0dEbFf460aC112a837C89BAa7cd

4. _setPriceOracle in CompTT
   result: 0x64740F1521d36021bF50Cd45Dc31346d5158C130
5. _setVenusComp() using Venus Comptroller
   Input 0x94d1820b2D1c7c7452A163983Dc888CEC546b77D
6. _setBNB() using wbnb/vbnb addresses
   Input: 0xae13d989daC2f0dEbFf460aC112a837C89BAa7cd,0x2E7222e51c0f6e98610A1543Aa3836E092CDe62c

4.2 Margin Tokens Deployment

1. Deploy XTT from XTTgov.sol
   Result: 0x3fF5f7ca6257E29deD56180f12Dd668c4D4b8ad3
    a. Add xtt and xvs in TrendTokenStorage.sol ~line 180-190
   Input: 0x3fF5f7ca6257E29deD56180f12Dd668c4D4b8ad3 & 0xB9e0E753630434d7863528cc73CB7AC638a7c8ff
2. Deploy MarginToken from MarginToken.sol
    a. wbnb and CompTT (4.1) addresses
   Input: 0xae13d989daC2f0dEbFf460aC112a837C89BAa7cd,0x8cE9443B7a6BAeD8151Be38D074E2940bFe158Cd
   Result: 0x1e471D64F386147CFBf5544284886aDa85fB3bA4
3. Deploy IncentiveModelSimple.sol
   Result: 0xc3A6a8DA481b769d79f716A1e5F90475381b73A0
4. _supportTrendToken() in CompTT
   Input: 0x1e471D64F386147CFBf5544284886aDa85fB3bA4
5. _updateCompAndIncentives(0x00, addr) in MarginToken.sol
   Input: 0x0000000000000000000000000000000000000000,0xc3A6a8DA481b769d79f716A1e5F90475381b73A0

4.3 Operation Manual

4.3.1 Adding a token to Margin Token

1. _setFeed() in ChainlinkOracle.sol
    a. required to fetch prices for token
2. _updateFeePerToken() in IncentiveModelSimple.sol
    a. required to calculate fees for trend token deposit, redeems, and trades
3. _supportUnderlying() in CompTT.sol
4. _updatePortfolioAndAllocations() in MarginTokens.sol
    a. Input: [tokens],[>0 contract],[0 collateral],[0 borrow]

4.3.1.1 User Action Testing

- Deposit a token in portfolio
    - Expect: success
- Deposit a token not in portfolio
    - Expect: failure

- Redeem amount less than in portfolio
    - Expect: success
- Redeem amount more than in portfolio
    - Expect: failure
- Trade buy amount enough in portfolio
    - Expect: success
- Trade buy amount not enough in portfolio
    - Expect: failure
- Pause actions for Trend Token in compTT
    - Expect: failure
- Pause actions for underlying in compTT
    - Expect: failure
- Pause entire Trend Token
    - Expect failure

4.3.2 Support token supply and borrow to Venus

1. Steps 1-3 from 4.3.1
2. _supportVToken(token,vToken) in CompTT
3. _updateTrendTokenVenusActions() in CompTT
    a. set to true for supply and borrow as desired
4. _updateUnderlyingForVenusAction() in CompTT
    a. Set to true for isVenus, isSupply, and isBorrow as desired
5. _updatePortfolioAndAllocations() in MarginTokens.sol
    a. Input: [tokens],[contract],[>0 collateral],[>0 borrow]

4.3.2.1 Venus Testing

- Supply
- Redeem
- Borrow
- Repay

4.3.3 Removing a token

1. _setDesiredAllocations() to [0],[0],[0] for token desired to be removed
2. _depositsDisabled(token,true) for token to be removed
3. systematically repay, redeem, and remove asset from pool
    a. Make sure contract+collateral+borrow is below maxDisableValue
4. _updatePortfolioAndAllocations() to not include removed token

4.3.3.1 Venus Testing

- Deposit disabled token (step 2)
    - Expected: token not allowed to be deposited
- Try to remove token with _updatePortfolioAndAllocations()
    - Expected: error maxDisableValue
- Repay, redeem, and remove desired asset from pool
- Borrow
- Repay

4.3.4 Adjust Fees

1. Performance Fee
2. Trade Fee

4.3.4.1 Fees Testing

- Redeem Trend Token fees
- Redeem XVS accrued

5.0 Test Cases