Technical Guide

1. **Overview of MXX DEFI Platform**

The MXX DEFI platform is built on Ethereum smart contracts using solidity. This platform allows users to interact with a fully open-source & audited smart contract to achieve the following:

* Create a Yield Contract by depositing their erc20 collateral. Each contract creation requires some gas fee (MXX) to be burned.
* Redeem a Yield Contract when it’s tenure matures. When a contract is redeemed, the user gets back the deposited collateral (in full) as well as the MXX minted by the contract.
* Early-redeem a Yield Contract, and make it available in the Open-Market for acquisition (by any user). Early redeeming a contract requires to burn some gas (MXX) from the minted MXX in contract. User get back his full deposited collateral.
* Acquire a Yield Contract that is available in the Open-Market. User do not need to burn any MXX gas to acquire a contract in the Open-Market.

Create Contract

Upon maturity

Redeem Contract

Finish

Create Contract

Early Redeem maturity

Available in  
Open Market

Active  
Contract

Acquired by user

Upon Maturity

Redeem Contract

Finish

1. **Contract life-cycle**
   1. The stages (status) of a contract.

A screenshot of a cell phone

Description automatically generated

Inactive:  
When the contract is not created

Active:

When the contract is created by a user or acquired by the user in the Open Market

OpenMarket:

When the contract is early redeemed and available in the Open Market for take-over by other users.

Claimed:

When a contract’s tenure matures and the reward is claimed. This is the final status of a contract.

Destroyed:

When a contract that is in the Open Market is un-acquired for a long period of time, the smart contract owner may call destroyOMContact() method to dissolve this Open Market contract (if required).

1. **Terms, Convention and Precision**

For the purpose of this smart contract, the follow conventions are used:

* Interest Rate Related (eg APY)

Shall multiply by 106

Example: 5% is expressed as 0.05 \* 106 = 5,000

Example: 300% APY is expressed as 3 \* 106 = 3,000,000

* mFactor (mint factor)

The mFactor is a term used to represent the Price factor of the assets. The mFactor is used instead of using a Price Oracle is to remove the possibility of price manipulation or price volatility due to market condition. By having a fixed mFactor, we ensure a stable yield return rate over a period of time.

Each supported erc20 token will have a mFactor set in our smart contract when asset is added. This value will be reviewed weekly and updated (and announced to public as required)

A screen shot of a smart phone

Description automatically generated

For example:   
  
MXX at a price of $0.05 will be expressed as mFactor = 0.05 \* 1018 = 50,000,000,000,000,000

ETH at price $400 will be expressed as mFactor = 400 \* 1018 = 400,000,000,000,000,000,000

mFactor of each supported erc20 tokens can be updated with this function:

**A screenshot of a cell phone

Description automatically generated**

* Tenure and APY rate

The default tenures are 90d, 180d and 270d. The default APYs are 200%, 400% and 1,000% respectively.

New tenure/apy can be added or set using the following function.

A screen shot of a smart phone

Description automatically generated

* Gas (Burn Fees)

There are 2 operations that will burn MXX token as gas fee.

1. Contract Creation

When a contract is created, the total MXX (to be minted) is first calculated. A 8% of MXX gas fee is to be used (from user’s wallet) to be sent to the official Burn address. Once every week, all the MXX coins in the burn address will be burned, and thus reducing the total MXX in circulation.

A close up of a screen

Description automatically generated

The default value used is 8%. However, this value can be changed using the setParamType() method, passing in ParamType.ContractFee and the new value.

1. Early Redeem a Contract

When a user “early redeem” a contract, there is a gas (burn fee) to be paid in the form of MXX. The graph below illustrates the % of the MXX Minted by the user that will be used as gas fee (to be burned), for a 90 days tenure.

0

Tenure (eg 90)

50

5

Again, the MXX is sent to the official burn address and will be burned away every week.

1. **Create a Yield Contract**

A new Yield Contract can be created using the following function.

A screenshot of a cell phone

Description automatically generated

By specifying the erc20 token address, collateral amount and tenure, a new contract can be created.

1. **Claim a Yield Contract**

To claim (redeem) a contract upon maturity, the follow function is called:

**A screenshot of a cell phone

Description automatically generated**

Only the user who owns the contract can claim his/her contract.

1. **Early Redeem a Yield Contract**

To early-redeem a contract, the follow function is called:

**A screenshot of a cell phone

Description automatically generated**

Only the user who owns the contract can early-redeem his contract.

1. **Acquire a Yield Contract**

To acquire a contract from the Open Market, the follow function is called:

A screenshot of a cell phone

Description automatically generated

When acquiring a contract from the open market, the user does not have to pay for the MXX Gas fee. The user is only required to deposit the collateral requirement.   
  
Note that for contracts in the Open Market, the remaining MXX (to be minted) will be a lower value as compared to the original MXX quantity when the contract is first created. This is because when this contract is early-redeemed by the previous owner, part of the MXX amount is already minted by the previous owner.

1. **Withdraw MXX**

As a safe-guard, there is a function that allows the withdrawing of MXX that is deposited into this contract by the core team.

A screenshot of a cell phone

Description automatically generated

This is a precautionary measure only and only applies to the MXX token.  
For the rest of the erc20 tokens collateral, ONLY the contract owners can withdraw using either “Early Redeem” or “Claim” function.