

# Frankencoin\*

Luzius Meisser\*\*  
*Meisser Economics*

Basile Maire\*\*\*  
*Desma Eight, LLC*

January 9, 2022

---

\*Preliminary

\*\*[luzius.meisser@gmail.com](mailto:luzius.meisser@gmail.com)

\*\*\*[basile.maire@desma8.io](mailto:basile.maire@desma8.io).

# Frankencoin

## Abstract

blabla

*JEL Classification Codes: D40, G23*

# 1 Introduction

We start from the use-case of a Swiss Frank Lombard loan, collateralized with tokens such as Bitcoin, or tokenized shares. The borrower deposits collateral into the system and thereby mints a token, the “Frankencoin” ZCHF, that is pegged to the Swiss Frank. If the value of the borrower’s collateral falls below a certain threshold, the loan can be liquidated and the borrower incurs a haircut. If the value of the collateral falls below the loan value before the liquidation ends, the loss eats into capital. Implementing this system in a fully decentralized way on the Blockchain leads us to our main contribution.

First, we introduce an algorithmic stablecoin that allows for *different collateral types and liquidation rules*. This diversification of collateral and liquidation rules reduces the impact of non-systemic market events to the stablecoin peg. Second, we propose a specific mint plugin setup for which

- We use an *auction mechanism* that avoids using external Oracle price feeds
- To calibrate the system, we apply traditional risk management techniques that have not found their way into the world of stablecoins

The system entails a decentralized governance that governs risk parameters and can deny the community’s addition of new peg-methodologies if deemed unfit.

The Frankencoin can be minted by anyone using one of the available mint plugins. We define mint plugins as smart contracts that have the ability to mint Frankencoin in accordance with their rules. Mint plugins have to be approved by the governance mechanism and there might be different types of mint plugins for different types of collateral and different liquidation mechanisms. Anyone can propose to add a new mint plugin and if it is approved by the governance mechanism, it can be used according to its rules. These rules can include interest rates that are owed by the minters, liquidation mechanisms, collateral requirements, and required ZCHF to be held by “stakers” as a reserve per ZCHF token minted. While every mint plugin defines their required reserve capital, the reserves are shared among all mint plugins.

Stakers are participants that lock their Frankencoin into the system. Stakers earn interest rates paid for by borrowers and in turn risk to lose Frankencoin in case the liquidation process is not able to recover the loan amount with the available collateral. Staked Frankencoin are subject to a lockup period. This ensures that in case of a "bank run" on the system, stakers are the last who can liquidate their Frankencoin and will therefore also suffer from the greatest losses while the other participants are likely to be able to reclaim the full value.

This paper focuses on two aspects of Frankencoin. First, we discuss how the Frankencoin embeds with the wider economy. Many existing stablecoin systems fall short of this aspect and can arguably only be sustained with significant growth of the system or demand of their own token, see e.g., [Clements, 2021]. We propose a specific setup of mint plugins for which we use standard risk methodology techniques to quantify adverse events and calibrate capital reserves and fees. This is in stark opposition to existing stablecoins that set parameters such as the collateralization-ratio without risk methodological quantification methods. *TODO: reference??*

## 2 Economy

The immediate participants of Frankencoin consist of *minters*, *users*, and *stakers*.

*Minters* deposit collateral into a mint plugin and thereby enter a Frankencoin Lombard loan. Minters benefit from the system to the extent that it allows them to use liquid or illiquid assets that they want to hold long-term, to generate short-term liquidity. This can be motivated e.g., by tax considerations in some jurisdictions, or by justifications that hold for traditional Lombard loans. Similar to Lombard loans, the position is overcollateralized.

*Stakers* stake Frankencoin in a dedicated contract, thereby locking these funds up for a certain period of time. As a consequence, stakers are the last to be able to liquidate their holdings in case of a collapse of the value of Frankencoin. Stakers have to be compensated for their risk. This compensation is provided by the minters. Fees and interest rates paid by the minters are distributed to the stakers. By staking Frankencoin, stakers release governance tokens that give them voting rights.

*Users* hold and transfer Frankencoin as a means of payment or store of value. No fees are charged to the users, but they also are not provided with any financial gain from holding Frankencoin by the system. The system should be designed such that the tail risk of a complete default is negligible for the users as that risk is outsourced to the stakers.

This setup separates risk-takers (stakers) from the users. Governance token holders have "skin in the game" and are thus incentivized to maintain a healthy system. For instance, because staker capital is shared for all mint plugins, stakers have a vested interest to retain a healthy ecosystem of mint plugins.

Creditors of Frankencoin loans should be paid a risk-free rate corresponding to the Swiss Frank, plus a risk-premium. Otherwise, there is no rationale why stakers should sustainably lock their funds in Frankencoin (other than for non-pecuniary reasons). From the perspective of the minters, this means that costs to minters should be in line with the market capital costs for their loan.

We now address the conditions under which the *peg to the Swiss Frank* should hold. We approach the valuation of the Frankencoin from the perspective of a *perpetual bond*. A perpetual bond, or consol, is a bond with coupon payments but no redemption date, see, e.g., [Jorion et al., 2010]. The staked Frankencoin is subject to default risk, because the system burns ZCHF when a minter's position is undercollateralized. We price this credit-risky perpetual along the lines of [Jarrow and Turnbull, 2000], by discounting the interest payments on a credit-risky term structure. Let's assume that interest payments happen at discrete time-steps  $0, \dots, \infty$  and we have corresponding risky rates of the term-structure so that the date-0 value of a promised Swiss Franc at time  $t$  of a credit-risky Franc promise is equal to  $\exp(-r_t t)$ . Let the constant coupon rate per Frankencoin be  $c$ . Now, the value of the perpetual can be written as

$$v(0) = \sum_{t=0}^{\infty} c e^{-r_t t} \quad (1)$$

$$= \sum_{t=0}^{\infty} c e^{-y t} \quad (2)$$

$$= \frac{c}{1 - e^{-y}}, \quad (3)$$

where the second line replaces the time-specific discount rates by a yield, and the last line is an application of geometric series. For the value to be at par,  $v(0) = 1$ , we have to choose the coupon rate accordingly:  $c = 1 - e^{-y}$ . Hence, if the interest earned from staking ZCHF are in line with discounting, the present value of one ZCHF is equal to one Swiss Franc.

The credit risky term-structure corresponds to the Swiss Franc risk-free term-structure plus a spread that compensates the investor for the risks. Hence whenever the risk-free term-structure, or the Swiss Franc risk changes,  $c$  has to be adapted for the value  $v(0)$  to be equal to one. This is difficult to automate, and we therefore allow the stakers to collectively set the interest rate (i.e., the risk-free rate plus spread). That is, if the exchange rate of the ZCHF is too low, the stakers use the governance mechanism to increase the interest rate and vice versa.

We design the system so that it is in the interest of the stakers to set the parameters of the system such that the peg is maintained. There should be no abuse of power, for example to set interest rates too high, therefore pushing the value of their ZCHF way beyond one CHF and essentially stealing the collateral as it would become too expensive for the minters to buy ZCHF to get their collateral back. To prevent such an attack, we ensure that stakers can only slowly adjust the interest rate, so that it is possible for the participants to trade-in their ZCHF before rates are too punitive (e.g., minters redeem their collateral by repaying their loan).

The next section presents a specific setup of mint plugins and proposes a calibration method to determine the appropriate spread that should bring the value of the Frankencoin close to a valuation of one Swiss Franc.

### 3 Specific Mint Plugin Setup

The Frankencoin system is open to accept any type of mint plugins. In this paper we describe a setup with two specific mint plugins that we consider to be of particular relevance when bootstrapping and growing the Frankencoin.

#### 3.1 Direct Peg Plugin

The simplest possible mint plugin is one that is based on a stablecoin with the same reference currency. Specifically, "off-chain" custodial stablecoins. For the Frankencoin, this could for example be the CryptoFranc (XCHF) issued by Bitcoin Suisse or the Digital Swiss Franc (DCHF) issued by Sygnum. This mint plugin allows anyone to deposit the specified stablecoin and to get Frankencoins in return. Also, the minting contract would allow anyone to convert Frankencoins back into the specific stablecoin for as long as there are any left.

Direct peg plugins have the advantage of strongly anchoring the value of the Frankencoin to one Swiss Franc by delegating the collateralization mechanism (e.g., directly backed by Swiss Francs). The disadvantage for direct peg plugins is the dependency on the issuers. Overall, direct peg are a great method to bootstrap the Frankencoin and diversify the Frankencoin system.

##### 3.1.1 Calibration

In case of issuer default, stakers have to burn ZCHF equal to the amount of loss given issuer default times the exposure to that stablecoin. To compensate stakers for this risk, we charge a minting fee. Again, we have the advantage that calibration is outsourced to the market. If the value of one stablecoin trades at  $1 - \delta$  to the Swiss Franc, governance sets a minting fee equal to  $\delta$ .

## 3.2 Liquid Collateral Plugin

The second type of mint plugin is designed for liquid collateral, such as Bitcoin. This type of plugin implements the use case of a Lombard loan that we motivated the paper with.

- The participants are minters, challengers, auction participants, and ZCHF stakers.
- The minter deposits collateral and thereby mints ZCHF. The ZCHF are overcollateralized at the time of minting, that is, the value of collateral deposited exceeds the value of the minted ZCHF.
- Challengers can initiate an auction process for a given position at any time. To do so, they deposit collateral of the same type. After the challenge initiation, auction participant bid for the collateral.
  1. If, according to the auction, the value of the collateral falls below a specified threshold, the position is liquidated and the minter loses their collateral. *E.g., the collateral deposited is 1500 LUSD, 1000 ZCHF were minted. Now, the position is challenged and the best bid for 1500 LUSD closes at 1095 ZCHF. Let's assume that the threshold is 10%. Now, because  $1095 \text{ ZCHF} < 1000 (1+10\%) \text{ ZCHF}$ , the position is closed out.* The challenger earns a fee, and the bidder gets the remaining part of the collateral.
  2. If, according to the auction, the collateral value is above the threshold, the position remains in the minter's ownership. The bidder gets the challenger's collateral and the challenger gets the bidder's amount of ZCHF.
  3. If, according to the auction, the collateral value falls below the minted amount of ZCHF, the bidder gets the minter's collateral, the challenger gets his collateral back plus he earns a fee, the bidder's ZCHF plus an additional amount belonging to stakers is burnt.

The system outlined above has the following parameters. Let  $h$  be the threshold that defines whether the position can be liquidated or not. That is, if, according to the auction, the value of the collateral is below  $Z(1+h)$ , where  $Z$  is the amount of ZCHF minted for a given position, the position is liquidated. Let  $k$  be the challenger reward (e.g., 2% of the ZCHF position). Finally,  $\tau$  is the duration of the auction (e.g., 3 days).

$$h : h > 0, \text{ liquidation threshold, liquidate if highest bid is below } Z(1+h) \quad (4)$$

$$\tau : \text{duration of liquidation process} \quad (5)$$

$$k : 0 < k < h, \text{ challenger reward} \quad (6)$$

### 3.2.1 Calibration

We approach the calibration of minting fees similar to credit pricing. We calibrate the fees that we charge to the minters so that the expected losses to the stakers are paid for by fees.

By  $\tilde{r}$  we denote the random variable that corresponds to the log-return of the collateral using CHF as the quote-currency (or numeraire). Now, we can express the loss to the stakers as the following random variable:

$$\tilde{L} = -\min \left[ (1+h)e^{\tilde{r}} - (1+k), 0 \right], \quad (7)$$

**Table 1: Minting Fees.** We calibrate minting fees for a collateral with 5% daily volatility, assuming a t-distribution with 4 degrees of freedom (left), or a normal distribution (right). The volatility is scaled by  $\sqrt{\tau}$ , and the mean is assumed to be zero. Further,  $h = 0.10$ ,  $k = 0.02$ . Fees displayed are in percentages.

$\tau$	<b>t-distr</b>	<b>normal</b>
<b>1</b>	0.38	0.08
<b>2</b>	0.90	0.38
<b>3</b>	1.38	0.72

per unit of ZCHF minted (e.g., if the position consists of  $Z$  ZCHF, the loss to the stakers is  $Z\tilde{L}$ ). To see this, first note that the starting value of the position is at least  $(1 + h)$  per unit of  $Z$ . Since  $\tilde{L}$  is per ZCHF, we set  $Z = 1$ . The value of the position at the end of the auction is  $Z(1 + h)e^{\tilde{r}}$ . Stakers need to burn ZCHF for the amount that the end-of-auction value falls short of the minted amount plus the challenger reward, hence we subtract  $(1 + k)Z$ . We need the  $\min[\cdot]$ -function because only losses are charged to the stakers, profits in the liquidation process are not distributed to stakers. Finally, the negative sign is convention to have a positive number for the loss.

If we assume that the auction process will result in prices similar to what we would observe on liquid exchanges, then a remaining risk is that the price falls into a loss region during the auction period. With a known density function for the return distribution over the period  $\tau$ ,  $f_\tau(x)$ , we can calculate the expected loss as follows

$$\mathbb{E}_\tau [\tilde{L}] = - \int_{-\infty}^{\ell} ((1 + h)e^x - (1 + k)) f_\tau(x) dx \quad (8)$$

$$\ell = \log \frac{1 + k}{1 + h}, \quad (9)$$

where the subscript  $\tau$  emphasizes that the distribution depends on the time-horizon of the auction.

Table 1 shows results from calibrating Eq. (8). BTC-USD has a daily volatility of about 5%. The table compares the result for Eq. (8) when assuming either a t-distribution or a normal distribution for the return.

## References

- [Clements, 2021] Clements, R. (2021). Built to fail: The inherent fragility of algorithmic stablecoins. *Wake Forest L. Rev. Online*, 11:131.
- [Jarrow and Turnbull, 2000] Jarrow, R. A. and Turnbull, S. M. (2000). *Derivative securities*. South-Western Pub.
- [Jorion et al., 2010] Jorion, P. et al. (2010). *Financial Risk Manager Handbook: FRM Part I/Part II*, volume 625. John Wiley & Sons.