hOHM Final Specs

hOHM

`hOHM is tokenised leveraged OHM generated against perpetually maximised Cooler Debt`

What is Cooler? (aka MonoCooler, aka Cooler v2)

- An improved version of the Olympus borrow/lend platform.
- Supply gOHM collateral, borrow USDS with a high LTV at 0.5% interest
- MonoCooler Tech Docs
- CoolerV2 Final Specs
- Code: MonoCooler.sol

In Kind Mint and Redeem

- hOHM share price is defined as two values:
  - `hOHM/gOHM`: The gOHM collateral balance per hOHM
  - `hOHM/USDS`: The net USDS debt per hOHM
- When minting hOHM users provide the gOHM and receive USDS + hOHM
- When redeeming hOHM users provide hOHM + USDS and receive gOHM
- No entry fee
- There is an exit fee which burns a % of the hOHM shares being redeemed (if the exit fee is 1% then 1% of the hOHM redeemed gets burned prior to running the share calculation)

    - Can be thought of as a ‘retention bonus’ for long term holders since each exit causes the share price to increase.

Share Price Calculations

- Both share prices are determined on initialization of the vault, via a `seed()` function. This also serves to avoid inflation style attacks.
- After the seed, the share price is dynamic based on the current collateral balance and net debt which hOHM has. eg:
    - As the hOHM debt position grows from the accrued Cooler interest, `hOHM/USDS` share price increases
    - As buybacks happen, `hOHM` is burned increasing both the `hOHM/gOHM` and `hOHM/USDS` share prices

Tokenised Balance Sheet Vault

- A new ERC-20 based vault type has been created in order to accurately represent this type of structure, one we hope to use for multiple products over time
- Rather than using on-chain oracles to convert tokens within the vault into a single deposit token (like ERC-4626), each token in the vault maintains its own share price
- These tokens can be categorized into Assets and Liabilities
- On mint: Caller provides each asset, and receives the vault shares (minted) and receives each of the liabilities
- On redeem: Caller provides the vault shares (burned) and each of the liabilities, and receives each of the assets
- Further details can be found in ITokenizedBalanceSheetVault

USDS Surplus (aka buffer)

- hOHM will always max out the origination loan value for its MonoCooler position
    - It will do this on any new mint or redeem, and can also be called periodically.
- That being said if the debt balance is above the origination max value then it may not be able to withdraw enough gOHM to handle a redemption
- To handle this, a portion of borrowed USDS is withheld as a buffer (this portion is not included in the USDS per hOHM share price)
- This would be a nominal USDS amount, but kept as sUSDS+s to earn yield on the idle funds.
- As the Cooler origination LTV increases over time (it will linearly increase per second), hOHM gains additional borrowing capacity
    - The extra debt will be added into the surplus so it can grow/replenish over time.

    - If the surplus grows large enough, buyback and burns can be performed using some of this surplus to soft cap the surplus.

hOHM Buybacks (aka `sweep`)

- TempleDAO will provide liquidity in a sUSDS/hOHM 1% UniV3 range
- When the surplus is above a soft target the excess sUSDS+s can be sold for hOHM repurchases
    - A small cut of this is sent to the OrigamiDAO fee collector (3.3%)
    - The remaining is burned, raising both the `hOHM/USDS` and `hOHM/gOHM` share prices as a result
- The swaps can be performed by a dex aggregator (eg 1inch/kyberswap), and when there's support can also be upgraded to use CowSwap programmable swaps (as we have internally for sUSDS+s sky farm compounding)
    - If also not supported in 1inch/kyberswap then we can integrate directly with the univ3 pool.
- Arbitrageurs (including ourselves) would then buy OHM, convert it to gOHM and mint hOHM to dump into the Uniswap pool if it is profitable to do so
    - This will keep a soft peg for the pool vs the vault share prices.
- To avoid rogue buybacks, `sweep()` has both:
    - A cooldown so it cannot be called too frequently
    - Each call can only sell a max amount of USDS

UX – Buying/Selling vs Mint/Redeem

- The UX to mint and redeem is more complicated than say an ERC-4626 vault, since there are 3 tokens involved here.
- Because of this, the primary route we see new users buying and selling hOHM will be via the UniV3 pool above.
- Existing users will be able to migrate their position (via contract which will use USDS flashloan)
- Advanced users can still mint/redeem with the 3 tokens.

hOHM Arb bot

- In order for hOHM to be completely efficient and not rely as much on MeV bots we should arbitrage it ourselves, since the arb bot can have a balance of DAI/sUSDS/hOHM/gOHM it should be more efficient than MeV bots at executing arbs, the goal is not to profit but to ensure efficient market pricing for hOHM.

gOHM Delegation

- Holders of hOHM (on mainnet only) are eligible to delegate their proportional amount of gOHM for voting rights (https://snapshot.box/#/s:olympusdao.eth) to at most one address.
- Holders **cannot** delegate if the proportional gOHM amount is less than 0.1 gOHM
- hOHM will have one single borrow/lend position in cooler, and will be whitelisted to delegate to many accounts in cooler (by default that's only 10)
    - It needs (at max) one delegate per hOHM holder.
    - In reality it will be less – hOHM holders who don't care about voting may not delegate at all, and many hOHM holders could delegate to the same underlying account.
- Since the `hOHM/gOHM` share price may increase over time, the amount of gOHM a holder has to delegate will also increase over time. Holders will need to manually synchronise their delegation to increase the actual amount delegated
- If a holder has already delegated, then their delegation amount will automatically synchronize to their latest proportional gOHM when they either:
    - Mint more
    - Redeem some
    - Transfer (to self or someone else)
- Similarly if a holder is the recipient of an hOHM transfer and that holder already has a delegate setup then the delegation amount to that delegate will be automatically synchronized to the higher voting total.
- This synchronization will strip the delegation if the amount of gOHM is under the 0.1 gOHM threshold. It works as housekeeping to reduce any dust delegations which may impact the ability to liquidate this account in Cooler and also reduce the number of delegates counting towards the cooler cap

hOHM Fees

- Protocol fee from the total amount which is bought back. Set to 3.3% (updatable, but max hardcoded in the contract to 3.3%)
- Retention Bonus (ie exit fee) with a max of 3.3% (set to 1% at launch)

What parameters does Origami DAO Multisig have control over?

- Setting the exit fee (max 3.3%)
- Setting the buyback protocol fee (max 3.3%)
- Setting the fee collector address
- Disabling Cooler max borrows (to ensure repays can still happen if upstream cooler borrows are paused for whatever reason)
- Setting the sweep cooldown and max amount to sell
- Updating to the latest Cooler debt token (it may change upstream)

- Setting the savings vault for the debt, eg sUSDS+s

Cooler Upgrades?

- The debt token within Cooler can be upgraded - eg from USDS to USDC
- hOHM can handle this with an admin function to set the debt token to the latest in Cooler.
    - However there is some extra manual work required:
        - Updating the sweep params
        - Selling any surplus USDS and buying USDC
        - Etc
    - For this reason, it requires the vault to be paused for both mints and redemptions first.
- If Cooler itself has to be re-deployed, then it will require a new hOHM-v2.

Berachain and L2s

- LayerZero will be utilised for transferring hOHM cross-chain
- The mechanics above describe hOHM on mainnet only. Including
    - The tokenized balance sheet 3 token mint/redeem
    - gOHM Delegation
    - Buybacks
    - Etc
- On Mainnet, hOHM will use a separate OFT Adapter contract, which we call `OrigamiTokenTeleporter`
    - The tokens on mainnet will be locked (not minted/burned) in this teleporter contract
- On Berachain and others, hOHM will just be a pure OFT
    - Liquidity will be supplied by Temple, but then also incentivised for PoL on Berachain primarily by OlympusDAO.
- So if users want exposure to hOHM on Berachain they can just buy/sell on Kodiak for example.

hOHM's Cooler Debt and LTV => Surplus

Under normal circumstances:

1. The Cooler Origination LTV will increase over time, increasing the amount hOHM can borrow
    - Olympus' treasury is primarily backed in sUSDS earning >> 5% APY
    - The Cooler Origination LTV is set at a conservative level to also increase with this increased backing
    - This linear increase is set by governance, not an on-chain oracle
2. The Cooler interest rate is set at 0.5% APY, decreasing the amount hOHM can borrow
3. Any surplus USDS that hOHM is holding is kept in sUSDS+s (greater of sUSDS or auto-compounding the Sky farms), earning >> 5% APY

It is expected that (1)+(3) will outpaces (b) substantially. On aggregate, this means the overall borrowing power will increase over time, and hence the surplus will increase.

If (2) outpaces (1)+(3) then the surplus will shrink.

Appendix: Example Calcs

NB: These calcs may be a bit stale / not taking into account every policy setting/threshold above

Full hOHM Flow Example:
T0:

- 0.003714158 OHM/gOHM (fixed conversion rate)
- Origination Loan Value of 2,961.64 USDS/gOHM (roughly 11 USDS/OHM)
- Liquidation Loan Value of 2,976.45 USDS/gOHM (roughly 11.055 USDS/OHM or 1 year of interest until liquidation)
- 1% Redeem Fee (no Mint Fee)
- 1,000,000 hOHM in circulation
- 3,714.158 gOHM posted as collateral in CoolerV2 (1,000,000 OHM worth)
- 11,000,000 USDS debt in Cooler V2
- USDS buffer of 100k, held as sUSDS+s
- Net USDS balance of 10,900,000 USDS (USDS Cooler debt - sUSDS+s buffer)
- Share Prices:
  - gOHM per hOHM: 0.003714158 gOHM/hOHM (1 OHM/hOHM)
  - Net USDS per hOHM: -10.9 USDS/hOHM

Mint Example:

- Alice wants to mint 1,000 hOHM
- With the current share price Alice needs to provide 3.714158 gOHM and they will receive 10,900 USDS
- hOHM takes the 3.714158 gOHM, posts it on CoolerV2 and max borrows 11,000 USDS against it
  - 10,900 USDS goes towards paying Alice
  - Remaining 100 USDS goes to the buffer
- sUSDS+s buffer would now have 100,100 USDS, since it is over the target buffer once there is a large enough amount it will be used for buybacks
- Post Mint Share Prices:
  - gOHM per hOHM: 0.003714158 gOHM/hOHM (1 OHM/hOHM)
  - Net USDS per hOHM: -10.9 USDS/hOHM

Redeem Example:

- Bob wants to redeem 1,000 hOHM
- A 1% exit fee is burnt, so 10 hOHM gets burnt and only 990 hOHM is actually being redeemed
- With the current share prices, the 990 hOHM:
  - Bob needs to send 10,791 USDS (990 * 10.9)

    - Bob would receive 3.67701642 gOHM (990 * 0.003714158)
- In order to free up the 3.67701642 gOHM, hOHM needs to repay 10,889.9989101288 USDS (3.67… * 2,961.64)
    - Since it received 10,791 USDS from the Bob redeeming it would need to get the 98.9989 USDS difference from the buffer
- sUSDS+s buffer would now have 99,901.001 USD, since it is below the target buffer there would be no buybacks
- Post Redeem Share Prices:
    - **Changes because of the fee burned**
    - gOHM per hOHM: **0.00371419514195** gOHM/hOHM (1.00001 OHM/hOHM)
        - `(3,714.158 - 3.67701642) / (1,000,000 - 1,000)`
    - Net USDS per hOHM: **-10.900109** USDS/hOHM
        - Net USDS = `(11,000,000-10,889.99891) - (100,000 - 98.9989) = 10,889,209`
        - `10,889,209 / (1,000,000 - 1,000)`

Buyback Example:

- Let's suppose that the buyback batch size is set to 25k sUSDS+s and the Buffer has grown to 125k sUSDS+s (so 25k above the target)
- In this case a CowSwap order would be placed to buy 25k sUSDS+s worth of hOHM from the `UniV3 sUSDS+s/hOHM pool`
- The hOHM proceeds would be burnt thus increasing the amount of gOHM held by each hOHM

Seeding Example:

- Origination Loan Value of 2,961.64 USDS/gOHM (roughly 11 USDS/OHM)
- Liquidation Loan Value of 2,976.45 USDS/gOHM (roughly 11.055 USDS/OHM or 1 year of interest until liquidation)
- Initialize tx prices 1 hOHM as 0.0037141158 gOHM (roughly 1 OHM) and -10.9 USDS/hOHM
- We deposit 10 gOHM to initialize it, hOHM takes it and max borrows the origination value of 29,616.4 USDS then mints 2,692.4 hOHM (29,616.4 USDS / 11 OHM LTV) giving it to the minter and 29,347.16 USDS to the user (10 gOHM * 269.24 OHM/gOHM * 10.9 USDS/lovOHM), the remaining 269.24 USDS is deposited into the Buffer