

Twitter Thread by Trader Joe



Trader Joe

@YieldFarmerJoe



1) Getting Alpha on your \$alpha. Why it's my biggest gamble.

Many may think that the product @AlphaFinanceLab = Alpha Homora, but that would be akin to saying Adobe Photoshop is Adobe.



2) Alpha aims to be a suite of #DeFi products that looks for market opportunities in DeFi by solving its problems. They will identify a problem and then build quick to be the first mover.

3) Alpha Homora (leveraged yield farming) happens to be the first. Heck Homora could very well turn out to be its Adobe Prelude and we haven't even gotten to its flagship Photoshop yet.

4) In this thread I will focus on it's upcoming project AlphaX (beta testnet round2), because people are already familiar with Alpha Homora.

5) Here are 2 great threads if you want to look into Alpha Homora more in depth:

V2 overview - <https://t.co/srsWaDoCyH>

Ironbank (capital efficient lending) integration with @creamdotfinance \$cream - <https://t.co/j7gzAc8CTE>

As the release of [@AlphaFinanceLab](#) Homora v2 seems to get closer by the day, let's take closer look at the protocol, the various components, and what's coming.

\U0001f9f5\U0001f447

— Sawit Trisirisatayawong (@tansawit) [January 24, 2021](#)

6) Before I delve into AlphaX a disclaimer that this is after preliminary research and if I am mistaken about anything, please correct me.

AlphaX is a decentralized, non-orderbook perpetual swap trading market. Quite a mouthful.

7) Decentralized (you own your keys) non-orderbook (you're not interacting with a buyer when you're selling and vice versa) perpetual (this contract doesn't have an expiration date, it lasts forever) trading market (trading market).

8) To keep things simple, unlike other futures (puts and calls), which have settlement dates, a perpetual contract -- as the name suggests -- lasts forever. So they are very much like buying SPOT. However you do not hold the underlying token.

9) Perps keep pegged to the underlying asset through a funding rate. At reg intervals either the LONG/SHORT side of the trade pays the other (the less popular side) a premium (grows the further away its from SPOT) to give incentive for the less popular side to be more popular.

10) So over time if you're on the popular side, you'll be paying quite a bit the longer you stay in and the more popular the side you're on is.

11) AlphaX's 1st unique selling point:

It doesn't matter how long you hold it for -- even if you're on the popular side -- there is no funding rate.

12) Instead it's baked into the buy and sell price. So it only matters WHEN you buy or sell. The premium or discount at the buy and sell price provides the incentive/disincentive to bring the price closer to the asset's spot price.

13) This makes for much easier simplification and a lot less babysitting. You only need to pay attention to WHEN you buy and sell instead of every 8 hours or every day (or whatever the funding payment occurs for the exchange you're on)

14) The second advantage of AlphaX and I believe more exciting is tokenizing the perp as an ERC-20 token. You don't have to open it on AlphaX. After anyone does, they can trade it on any secondary market ([@UniswapProtocol](#) [@SushiSwap](#)).

15) So u can hold a leveraged perpetual token OF a token, not have to worry or think about funding rates & trade in & out of it on popular DEXES & only look at its current price to make your decision. Imagine buying a baked in leveraged token of \$btc easily on uniswap.

16) Keep in mind the increased risk, b/c unlike spot, there is a price where your token is liquidated & becomes worthless.

That liquidation price is clearly written in the token name and it's closer and closer to current SPOT price the higher you choose to leverage.

17) AlphaX's products will reach more people because it leverages [@ethereum's](#) money legos.

You don't need to log into AlphaX or Perpetual Protocol or Injective protocol. The Total Available Market or demand for the product is tapped into the DEX market size.

18) I always mess up writing these threads. here's the last two parts.

