

Swaps

Hot Path

The most common operation on a DEX is a vanilla swap. Therefore `CrocSwapDex` contains the entire logic for a full swap inside the core contract. This avoids the gas overhead of `DELEGATECALL` to another proxy contract.

Note: Hot path swaps are not guaranteed to be available. Any integrations using Hot path swap should include a fallback to a cold path swap call. (Or just use cold path swap exclusively).

Hot path swaps are currently closed on the Ethereum and Scroll deployment. See section [#cold-path](#) for how to format using this callpath. The hotpath swap method is

...

Copy function swap (address base, address quote, uint256 poolIdx, bool isBuy, bool inBaseQty, uint128 qty, uint16 tip, uint128 limitPrice, uint128 minOut, uint8 settleFlags) public payable returns (int128 baseFlow, int128 quoteFlow)

...

The parameter arguments are as follows:

- base - The address of the base token or virtual token
- quote - The address of the quote token or virtual token
- poolIdx - The arbitrary index of the pool type the user is swapping on.
- isBuy - True if the user wants to pay base token and receive quote token. False if the user wants to receive base token and pay quote token
- inBaseQty - True if the quantity field is fixed in base token, false in quote token. (Regardless of whether they're being paid or receive)
- qty - The quantity of the fixed side of the swap.
- tip - If zero the user accepts the standard swap fee rate in the pool. If non-zero the user agrees to pay up to this swap fee rate to the pool's LPs. In standard cases this should be 0.
- limitPrice - Represents the worse possible price the user is willing to accept. If buying this represents an upper bound. For information on how to set this value or encode prices see [#note-on-limitprice](#)
- minOut - Minimum (maximum) expected output (input) of the token being bought (sold). Exceeding this value will revert the transaction.
- settleFlags - Flag indicating how the user wants to settle the traded tokens for this swap. For standard direct transfer of tokens to/from the caller's wallet, this should be set with a value of 0. (see [Type Conventions](#))
-)
-

The return value is the token flows associated with the swap for the base and quote token side of the pool. Negative indicates a credit that the caller received, positive indicates a debit that the caller paid.

Note on limitPrice

Limit price is based on the price of the curve, not the fill price of the swap. If the swap pushes the price of the curve to the limit price, then the swap will stop and leave the remaining quantity unfilled (rather than reverting the transaction). Traditional slippage limits should be set with the `minOut` parameter.

Price value is based on a specific encoding of the square root of the pool price in Q64.64 fixed point format. For more information on encoding see [Type Conventions](#)

Using a meaningful value here is not necessary if the caller is uninterested in partial fills and slippage is set with `minOut` parameter value. In this case this value can be set to "max values" below based on the direction of the swap:

- isBuy=false
- :65538
- isBuy=true
- :21267430153580247136652501917186561137
-

Cold Path

The same swap functionality is available using the `standarduserCmd()` method call rather than `swap()` call. The behavior is identical, except the swap code is called on a proxy contract which imposes an overhead of approximately 5000 gas.

The swap path is called with callpath index 1. The bytestring command is just the standard ABI packing of the above swap parameters:

...

```
Copy userCmd(1, abi.encode( base, quote, poolIdx, isBuy, inBaseQty, qty, tip, limitPrice, minOut, settleFlags))
```

...

The parameter arguments are as follows:

- base - The address of the base token or virtual token
- quote - The address of the quote token or virtual token
- poolIdx - The arbitrary index of the pool type the user is swapping on.
- isBuy - True if the user wants to pay base token and receive quote token. False if the user wants to receive base token and pay quote token
- inBaseQty - True if the quantity field is fixed in base token, false in quote token. (Regardless of whether they're being paid or receive)
- qty - The quantity of the fixed side of the swap.
- tip - If zero the user accepts the standard swap fee rate in the pool. If non-zero the user agrees to pay up to this swap fee rate to the pool's LPs. In standard cases this should be 0.
- limitPrice - Represents the worse possible price the user is willing to accept. If buying this represents an upper bound. For information on how to set this value or encode prices see [#note-on-limitprice](#)
- minOut - Minimum (maximum) expected output (input) of the token being bought (sold). Exceeding this value will revert the transaction.
- settleFlags - Flag indicating how the user wants to settle the traded tokens for this swap. For standard direct transfer of tokens to/from the caller's wallet, this should be set with a value of 0. (see [Type Conventions](#))
-)
-

[Previous userCmd Callpaths](#) [Next Hot Path Swap Migration](#) Last updated 19 days ago On this page * [Hot Path](#) * [Cold Path](#)