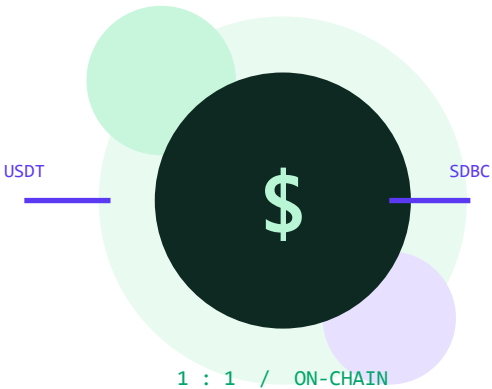


STABLE DOLLAR BASED COIN

SDBC Whitepaper

A non-custodial mint and redemption structure on Arbitrum One, connected to USDT reserves



NETWORK	OFFICIAL POOL	FEE
Arbitrum One	0xf2ad...a5c6B4	0.10% / 10 bps

Verifiable rules, user-signed transactions, and immutable core conditions.

This document describes the technical operation and risks of the SDBC smart contract. It is not an investment offer or a guarantee of returns. On-chain code and state take priority over this document.

VERSION 0.1 | 2026.10.01 | ENGLISH EDITION

01 / OVERVIEW

SDBC at a glance

SDBC is an ERC-20 digital asset designed to operate on Arbitrum One in connection with USDT reserves. Official minting and redemption occur only when a user wallet directly calls SDBCStablePool.

01

Clear backing

The relationship between SDBC supply and Pool reserves is visible on-chain.

02

User-signed

Neither the website nor administrators custody assets or sign on behalf of users.

03

Minimal change

Core logic, fee terms and fee recipient are immutable after deployment.

Official contract information

Item	Value
Network	Arbitrum One
Contract	SDBCStablePool
Official address	0xf2adf03D8B7083F5F096ccdf5D51558a61a5c6B4
Token	Stable Dollar Based Coin (SDBC)
Fee	0.10% for mint and redemption (10 bps), permanently fixed

Important: SDBC does not guarantee external market price, liquidity, legal status, or returns. Confirm the official address, network, approval amount, and final received amount before transacting.

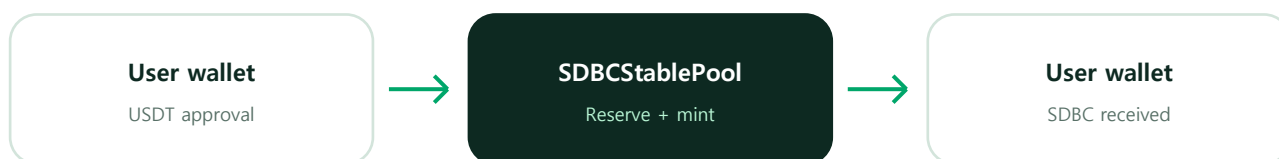
02 / MINT & REDEEM

Minting and redemption follow one on-chain rule.

The website is an interface for a user to call the official Pool. It does not custody assets or approve transactions.

Mint: USDT → SDBC

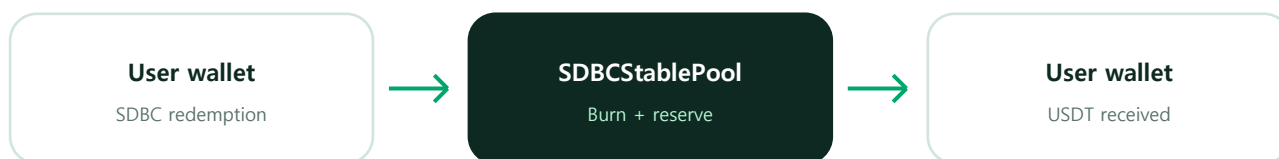
USDT → SDBC



The fixed 0.10% fee is sent directly to the fee recipient.

Redeem: SDBC → USDT

SDBC → USDT



The fixed 0.10% fee is sent directly to the fee recipient.

Fee example

Flow	Input	Fee (0.10%)	User result
Mint	100.0000 USDT	0.1000 USDT	99.9000 SDBC minted
Redeem	99.9000 SDBC	0.0999 USDT	99.8001 USDT received

Calculations use integer arithmetic in the token's smallest unit. Rounding can affect displayed values for very small amounts.

SDBC supply and Pool USDT reserve relationship

TOTAL SUPPLY
SDBC total supply

USDT RESERVE
USDT held by Pool

Designed to keep both values equal in normal mint and redemption flows.

NONE	Arbitrary administrator SDBC minting
NONE	Arbitrary administrator USDT withdrawal
NONE	Transaction approval, rejection, or queue
NONE	Contract pause or upgrade
FIXED	0.10% fee and immutable fee recipient

04 / ON-CHAIN INTERFACE

Code viewer: read functions and events

The following items can also be read directly from the verified Contract tab on Arbiscan. Website displays are for convenience; the final state is Arbitrum One on-chain data.

SDBCStablePool.sol - PUBLIC INTERFACE

```
// Read Contract

feeBps()          -> uint16  // 10 = 0.10%

feeRecipient()    -> address // immutable fee receiver

usdt()            -> address // Arbitrum USDT

totalSupply()     -> uint256 // SDBC total supply

reserveBalance()  -> uint256 // USDT held by Pool

// Events

Minted(account, grossUsdtAmount, sdbcAmount, feeAmount)

Redeemed(account, sdbcAmount, netUsdtAmount, feeAmount)

Transfer(from, to, value)

Approval(owner, spender, value)
```

Verification sequence

1	Open the official contract address on Arbiscan and check Source Code Verified.
2	Use Read Contract to inspect the fee, fee recipient, and USDT address.
3	Compare totalSupply() and reserveBalance() to inspect Pool state.
4	Use Minted, Redeemed, and transaction hashes to verify individual outcomes.

Keys stay with users. Transactions stay on-chain.

The SDBC server does not store private keys, mnemonics, wallet passwords, or user assets. Website administrators are limited to monitoring users, connected wallet addresses, indexed on-chain events, and inquiries.

User wallet	Private key custody · transaction signing · USDT/SDBC balances
SDBC website	Transaction UI · information · event index
SDBCStablePool	USDT reserve · SDBC mint/burn · fixed fee
Admin console	Read-only monitoring · no asset transfer authority

Browser wallet principles

A generated private key and mnemonic are displayed only to the user at creation. Only an encrypted keystore JSON may be stored in that browser’s IndexedDB; the server database records only the wallet address and connection state. Loss of a backup phrase, browser data, or device can make assets unrecoverable.

Never share a private key, mnemonic, or wallet password. SDBC operators and administrators do not need to request them.

Verifiable structure sets the standard for trust.

SDBC’s strength is not a promise; it is a set of on-chain rules that anyone can inspect. Users can check the official contract’s code, state, and events to verify the basis for minting and redemption.

ON-CHAIN PROOF	Verified source code and public events let anyone inspect contract rules and transaction outcomes.
SELF CUSTODY	Users keep assets in their own wallets and directly review and sign each transaction.
IMMUTABLE RULES	Mint and redemption fees, fee recipient, and core logic cannot change after deployment.
NO ADMIN CONTROL	Administrators cannot custody or move assets, or approve minting and redemption.

Next step

SDBC seeks a simple and verifiable mint and redemption structure through user-signed transactions, on-chain reserve visibility, fixed fee conditions, and minimized administrator authority. The official contract address and on-chain state can be checked directly on Arbiscan at any time.

OFFICIAL CONTRACT · 0xf2adf03D8B7083F5F096ccdf5D51558a61a5c6B4