

# Algorand

Details for working with Algorand environment chains.

## Developer Tools

The recommended development tool for Algorand is [Algokit](#).

## Addresses

Because Wormhole works with many environments, the Wormhole address format is normalized.

For Algorand chains, this means a wormhole formatted address is the 58 character address decoded from base32 with its checksum removed.

e.g. M7UT7JWIVROIDGMQVJZUBQGBNNIIVOYRPC7JWMGQES4KYJIZHVCRZEGFRQ  
=>0x67e93fa6c8ac5c819990aa7340c0c16b508abb1178be9b30d024b8ac25193d45

Algorand also uses a uint64 for Asset and Application IDs. These are converted to a 32 bytes by first converting to a 8 byte big endian byte array, then padding with 24 bytes of 0s.

e.g. 123 =>0x007b

## Emitter

The emitter is the application address, normalized to the wormhole address format.

## Algorand

### Ecosystem

- [Web site](#)
- [Algoexplorer](#)
- [AlgoScan](#)
- [Developer Docs](#)
- [Faucet](#)
- 

### Wormhole Details

- Name
- :algorand
- Chain ID
- :8
- Contract Source
- [:algorand/wormhole\\_core.py](#)
- 

### Consistency Levels

The options for [consistencyLevel](#) (i.e finality) are:

Level Value Finalized 0 This field is may be ignored since the chain provides instant finality.

For more information see [https://developer.algorand.org/docs/get-started/basics/why\\_algorand/#finality](https://developer.algorand.org/docs/get-started/basics/why_algorand/#finality)

### Mainnet Contracts (mainnet-v1.0 )

Type Contract Core 842125965 Token Bridge 842126029 NFT Bridge N/A

### Testnet Contracts (testnet-v1.0 )

Type Contract Core 86525623 Token Bridge 86525641 NFT Bridge N/A

### Local Network Contract

Type Contract Core 1004 Token Bridge 1006 NFT Bridge N/A

Last updated 1 month ago

On this page \* [Developer Tools](#) \* [Addresses](#) \* [Emitter](#) \* [Algorand](#) \* [Ecosystem](#) \* [Wormhole Details](#)

Was this helpful? [Edit on GitHub](#)