

How to use Supra's price feed oracle

Community member contribution Shoutout to [@ksdumont](#) for contributing the following [third-party document](#) ! Supra is a novel, high-throughput Oracle & IntraLayer: A vertically integrated toolkit of cross-chain solutions (data oracles, asset bridges, automation network, and more) that interlink all blockchains, public (L1s and L2s) or private (enterprises).

Integrating with Supra's price feeds is quick and easy. Supra currently supports several Solidity/EVM-based networks, like Arbitrum, and non-EVM networks like Sui, Aptos.

To see all of the networks Supra is on, please visit [Supras' Networks](#) !

To get started, you will want to visit [Supras' docs site](#) and review the documentation or continue to follow this guide for a quick start.

Step 1: Create The S-Value interface

Add the following code to the solidity smart contract that you wish to retrieve an S-Value.

```
interface ISupraSValueFeed {  
  
    function getSValue(uint64 _pairIndex) external view returns (bytes32, bool);  
  
    function getSValues(uint64[] memory _pairIndexes) external view returns (bytes32[] memory, bool[] memory);  
  
} This creates the interface that you will later apply in order to fetch a price from SupraOracles.
```

Step 2: Configure The S-Value feed address

To fetch the S-Value from a SupraOracles smart contract, you must first find the S-Value feed address for the chain of your choice.

For Arbitrum, the address is [0x8a358F391d93f7558D5F5E61BDf533e2cc3Cf7a3](#)

When you have the proper address, create an instance of the S-Value feed using the interface we previously defined for Arbitrum:

```
contract ISupraSValueFeedExample { ISupraSValueFeed internal sValueFeed; constructor() { sValueFeed =  
ISupraSValueFeed(0x8a358F391d93f7558D5F5E61BDf533e2cc3Cf7a3); } }
```

Step 3: Add unpack function to decode response for S-Value feed

To decode S-value response from SupraOracles smart contract, you need to add the following code in your contract.

```
// Some codefunction unpack(bytes32 data) internal pure returns(uint256[4] memory) { uint256[4] memory info;  
  
info[0] = bytesToUint256(abi.encodePacked(data >> 192)); // round info[1] = bytesToUint256(abi.encodePacked(data << 64  
>> 248)); // decimal info[2] = bytesToUint256(abi.encodePacked(data << 72 >> 192)); // timestamp info[3] =  
bytesToUint256(abi.encodePacked(data << 136 >> 160)); // price  
  
return info; }  
  
function bytesToUint256(bytes memory _bs) internal pure returns (uint256 value) { require(_bs.length == 32, "bytes length is  
not 32."); assembly { value := mload(add(_bs, 0x20)) } }
```

Step 4: Get the S-Value crypto price

Now you can simply access the S-Value crypto price of our supported market pairs. In this step, we'll get the price of single or multiple trading pairs in our smart contract.

```
function getPrice(uint64 _priceIndex) external view returns (uint256[4] memory) {  
  
(bytes32 val,)= sValueFeed.getSValue(_priceIndex);  
  
uint256[4] memory decoded = unpack(val);  
  
return decoded; }  
  
function getPriceForMultiplePair(uint64[] memory _pairIndexes) external view returns (uint256[4][] memory) {  
  
(bytes32[] memory val, ) = sValueFeed.getSValues(_pairIndexes);
```

```
uint256[4][] memory decodedArray = new uint256[4];
```

```
for(uint i=0; i< val.length; i++){
```

```
uint256[4] memory decoded = unpack(val[i]); decodedArray[i] = decoded; }
```

```
return decodedArray; }
```

Tada! You now have a method in your smart contract that you can call at any time to retrieve the price of any crypto pair!

Going further with Supra

If you want to take the next step, consider registering for the [Supra Network Activate Program \(SNAP\)](#).

The Supra Network Activate Program (SNAP) offers companies discounted oracle credits, technical documentation, and customer support to embed much-needed oracles and VRF/RNG. SNAP supports Web3 scaling and growth to buffer costs which could typically inhibit a company's success.

The SNAP program is partnered with some of Web3's most prolific names who are helping with project selection and qualification.

Connect with us

Still looking for answers? We got them! Check out all the ways you can reach us:

- Visit us at supraoracles.com
- Read our [Docs](#)
- Chat with us on [Telegram](#)
- Follow us on [Twitter](#)
- Join our [Discord](#)
- Check us out on [Youtube](#) [Edit this page](#) Last updated on Mar 19, 2024 [Previous](#) [Particle Network](#) [Next](#) [How to use Supra's VRF](#)