

## **Sunrise Self-Test**

Using the Signal Hound API you can now test the basic hardware blocks of the SA44B. You will utilize this functionality with the added function

```
int CMySignalHound::SelfTest();
```

## **Prerequisites**

- The CMySignalHound class must be created
- You must call CMySignalHound::Initialize() and it must return successfully
- The Self-Test BNC port must be connected to the RF input

## **Timing**

The SelfTest() function will take about 2-5 seconds to perform.

## **Return Values**

The SelfTest() function returns '0' (zero) if no issues are found.

If issues are found, the function returns a positive integer in which the 5 lower bits each represent the pass/fail status of the main SA44B electronic blocks. The bit mask values can be found in *MySignalHound.h*. The bit masks are prefixed with "ST".

Example.

```
// Self-Test BNC must be connected to RF input
int self_test = ptr_hound->SelfTest();
// Check return value
if(self_test) {
    // Issue found
    if(self_test & ST_POWER_DETECTOR)
        cout << "Power Detector Failure\n";
    if(self_test & ST_HIGH_BAND_MIXER)
        cout << "High Band Mixer Failure\n";
    ...
} else {
    cout << "Self-Test found no errors\n";
}
```

## Post Self-Test

The SelfTest() function leaves the settings of the device in an unknown state. If you wish to continue normal operation (sweeps/audio) the device must be fully reconfigured. All settings should be set again to ensure any leftover state from the SelfTest() function is removed.