



User's MANUAL

Sinope Software

for *Jupiter* Amplifiers

Issue date: 2023-06

	Version Number
<i>Sinope</i> software	2
<i>Driver</i> libraries	3.2

NOTICE
Please only store the equipment under the following conditions: <ul style="list-style-type: none">• Ambient temperature range of 10°C to 25°C• Relative humidity range of 20% to 60%• Keep away from sunlight!

Warranty and Liability:

The warranty does not cover the following items, even during the warranty period:

- Damage or loss caused by Acts of God such as fires, earthquakes, floods, lightning, etc.
- Damage or loss caused by failure to meet the specified conditions for this equipment, such as inadequate power supply, improper installation, or unacceptable environmental conditions.
- Damage due to changes or modifications made to the product.

In no event shall MR Shim be liable for the following:

- Damage loss or problems caused by relocation, modification, or repair performed by personnel not explicitly authorized by MR Shim GmbH.
- Damage or loss that results from negligence or from ignoring the precautions and operating instructions contained in this operation manual.

CAUTION: Not a medical device. Limited by Federal (or United States) law to investigational use.

Product Disclaimer Notice:

By using the Sinope software system, you accept and agree to be bound by and comply with the following terms of use and product disclaimer notice:

MR Shim GmbH does not grant, explicitly or implicitly, to any party any patent rights, licenses, or any other IP rights, whether with regard to such information itself or anything described by such information. Information provided in this document is proprietary to MR Shim GmbH, and MR Shim GmbH reserves the right to make any changes to the information in this document or to any products and services at any time without notice.

MR Shim GmbH does not assume any liability whatsoever arising out of the application or use of any part of the product or system. The products sold by MR Shim GmbH have been subject to limited testing and should not be used in conjunction with mission-critical equipment or applications. Any performance specifications are believed to be reliable but are not verified, and the buyer must conduct and complete all performance and other testings of the product, alone and together with, or installed in, any end-products. The Buyer shall not rely on any data and performance specifications or parameters provided by MR Shim

GmbH. It is the Buyer's responsibility to independently determine suitability of any products and to test and verify the same. The information provided by MR Shim GmbH hereunder is provided "as is, where is" and with all faults, and the entire risk associated with such information is entirely with the Buyer.

The Sinope software system is provided as is without any guarantees of any kind, either expressed or implied, including but not limited to warranties of merchantability, fitness for a particular purpose, of title, or of noninfringement of third party rights. Use of the product by a user is at the user's risk

USER'S MANUAL

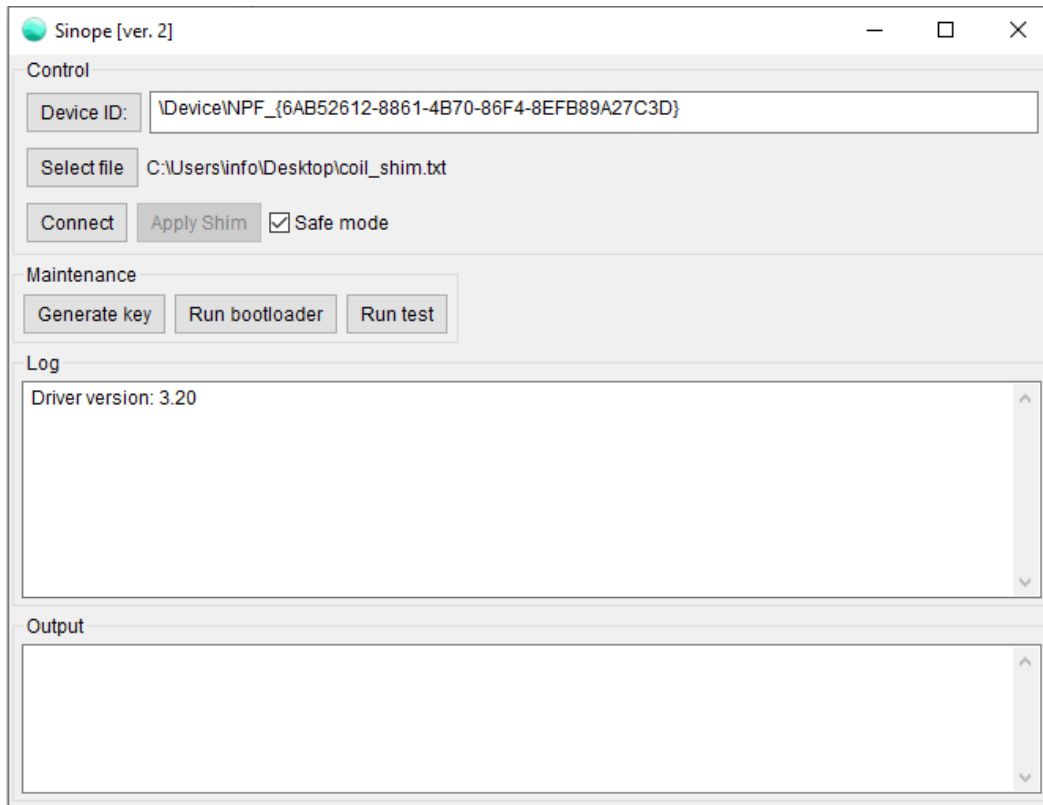
TABLE OF CONTENTS

	<u>Page #</u>
A. CONNECTING TO HARDWARE	1
1.1 Finding the Device ID	1
1.2 Connecting to the Device	2
B. APPLYING SHIM VALUES	5
2.1 Generating the Shim Text File	5
2.2 Applying the Shim Values	6
2.3 Trigger-based Update	6
2.4 Safe-mode	6
C. REAL-TIME UPDATES	8
3.1 Generating the Shim Plugin	8
3.2 Performance of Real-time Updates	8
3.3 Using Real-time Update	9
D. MAINTENANCE	11
APPENDIX	13

1.0 CONNECTING TO HARDWARE

A. CONNECTING TO HARDWARE

The Sinope software application is shown below.

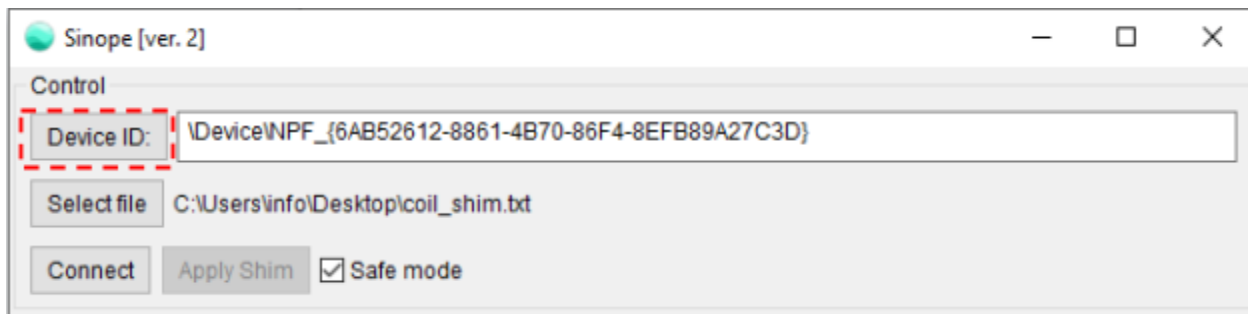


1.1 Finding the Device ID

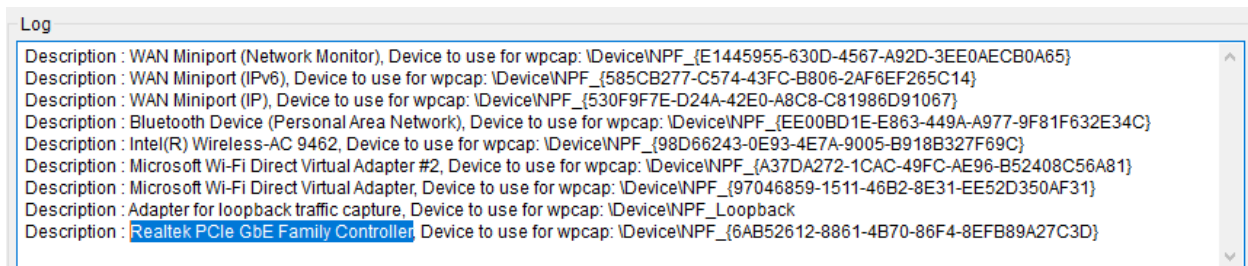
In order to connect to the *Jupiter* amplifiers, the amplifiers must first be connected to the control computer via an Ethernet cable. The relevant device ID of the Ethernet port must then be entered into the “Device ID” text box near the top (see below).



To find the correct device ID, click on the “Device ID” button next to the text box (shown below).



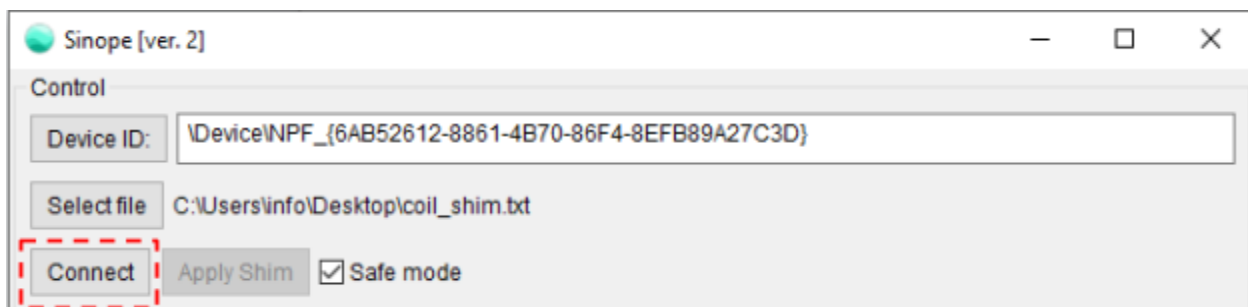
This will display a list of Ethernet interface device IDs and descriptions. Find the device that matches the description. For example, for a typical Ethernet port this will be a description like *PCIe GbE Family Controller*:



Copy the device ID from the description and paste it into the text box. In the example, this will be `\\Device\\NPF_{6AB52612-8861-4B70-86F4-8EFB89A27C3D}`.

1.2 Connecting to the Device

Connecting to the *Jupiter* device can be done by clicking the “Connect” button.

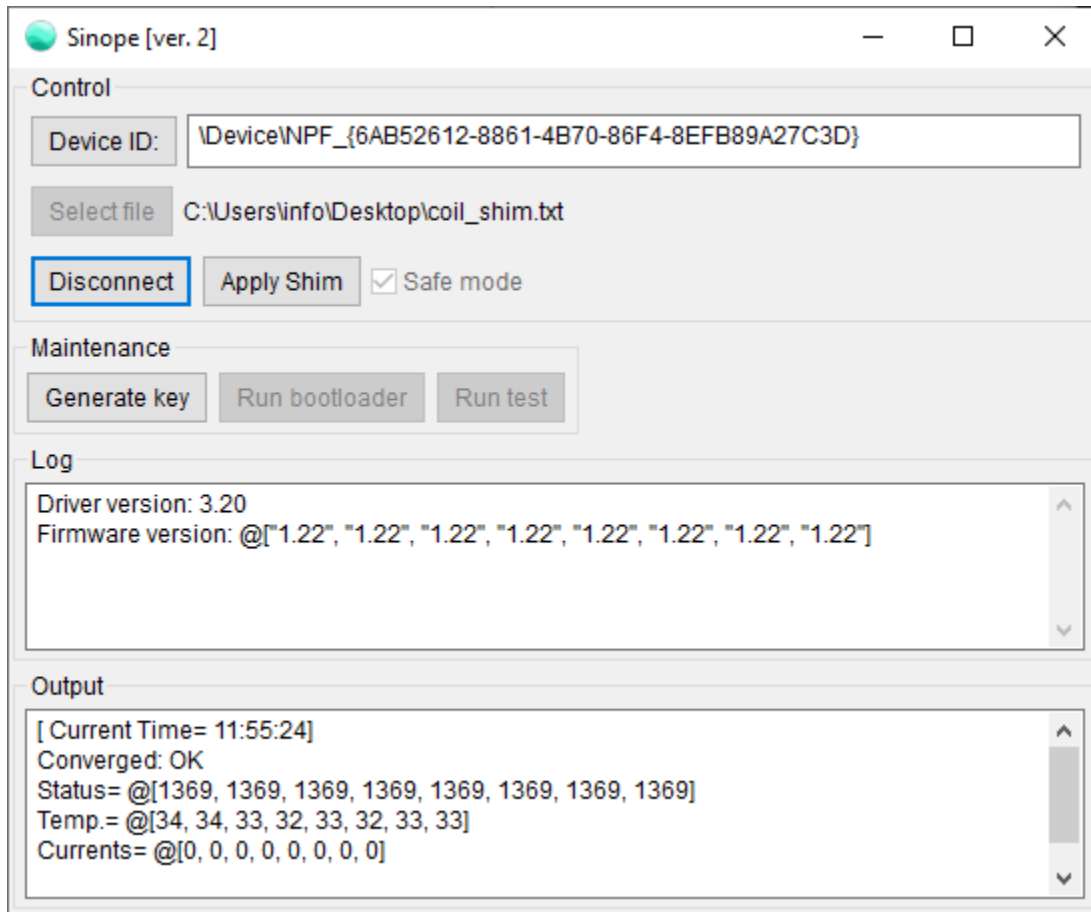


Once connected, the “Log” display will show the driver version and the firmware version for each channel of the connected *Jupiter* amplifiers. The “Output” display will display the status of each channel, which includes the following parameters:

- *Converged* - OK when the measured currents match the desired currents.
- *Status* - Status codes for each channel (for internal use for MR Shim employees)

- *Temp* - On-board temperature of each channel (below 70 deg C is ok). Overheating will automatically shut-off the amplifiers.
- *Currents* - Measured output currents of each channel (in mA).

When connected, the “Output” display will continuously update.



To connect to the amplifiers, a valid shim value file needs to be selected, otherwise an error will be displayed. See the section “Applying Shim Values” for more information.

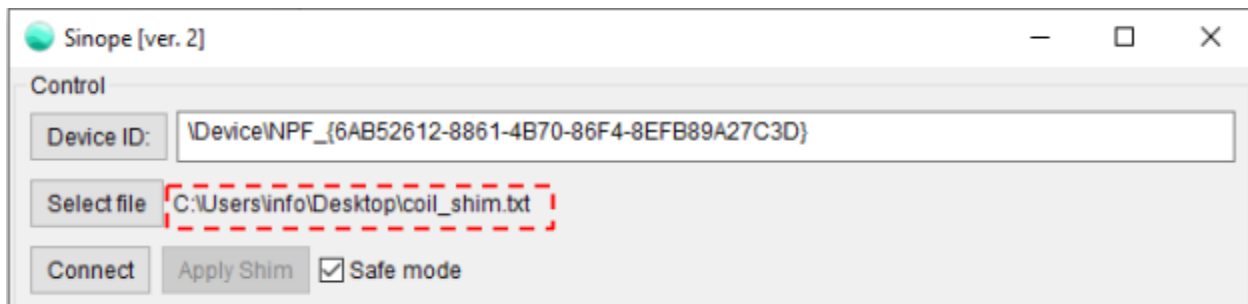
2.0 APPLYING SHIM VALUES

B. APPLYING SHIM VALUES

Shim current values are applied to the shim amplifiers using a text file. The text file should be selected by clicking on the “Select file” button (see below). This will open up a file browser where the file can be selected.

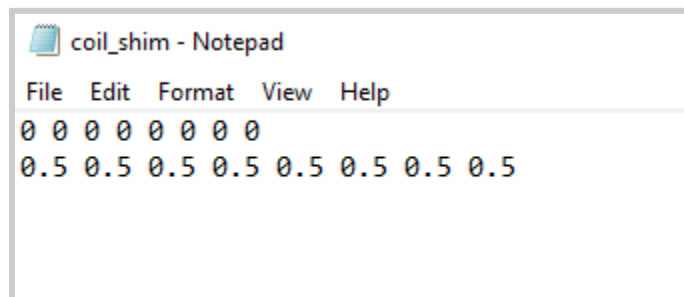


The selected file will be displayed next to the “Select file” button.



2.1 Generating the Shim Text File

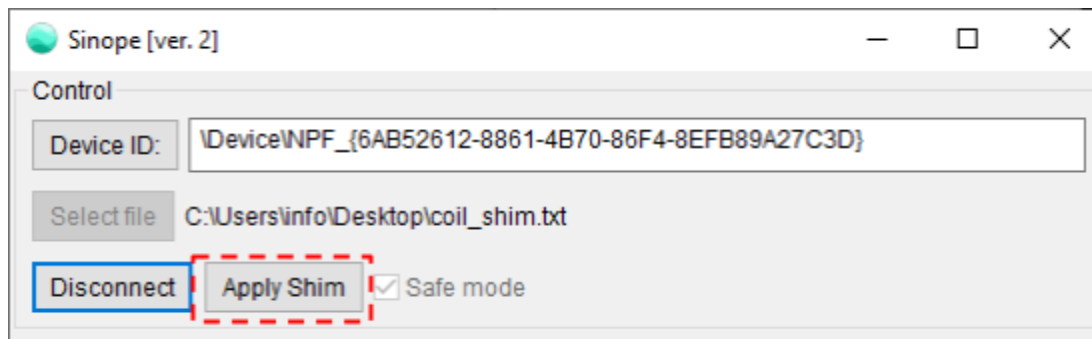
The text file should contain the shim current values (in Amps) where the values are separated by a white space. The file can contain multiple shim sets by adding extra rows of shim values. For example, the figure below shows two sets of shim current values, where the first set will apply 0A to the first 8 channels of the *Jupiter* amplifiers. The second set will apply 0.5A to the first 8 channels.



2.2 Applying the Shim Values

As soon as a connection is established with the *Jupiter* amplifiers (as described in *Section 1.2*), the first set of shim values will be applied to the amplifier channels.

The next set of shim values (i.e. next row of values in the text file) can be applied by clicking on the “Apply Shim” button as indicated below.



Once the last shim set is reached, the next shim set will start from the beginning again.

2.3 Trigger-based Update

If the trigger-based/real-time hardware device has been connected and set up, the shim sets will be automatically updated whenever a rising-edge trigger is detected. In other words, every trigger will apply the next row of shim current values in the text file.

2.4 Safe-mode

The *safe-mode* option can be enabled/disabled using the checkbox (shown below). This option cannot be changed while the *Sinope* software is connected to the amplifiers.

In *safe-mode*, the amplifiers will ramp up/down to the applied current values instead of applying the currents to the shim coils immediately. For time-sensitive and real-time applications, this option should be **disabled**.



3.0 REAL-TIME UPDATES

C. REAL-TIME UPDATES

For real-time updating, the trigger-based/real-time hardware device needs to be connected and set up. Additionally, the real-time update requires a plugin dynamically loaded library. For Windows, this will be the *shimplugin.dll*; and for Linux, *shimplugin.so*. This file should be in the same directory as the *Sinope* software. The library should contain a C/C++ compiled function named *ShimRealtime* that contains the code for how the updates should be applied.

The real-time update functionality can only be used with a computer that has a valid MR Shim license file.

3.1 Generating the Shim Plugin

The library should contain a C-compiled function named *ShimRealtime*, which accepts two arguments: an integer and a pointer to pointer of floats values.

- The first argument is an integer with the analog value received from the real-time hardware device. This value is normalized between 0 and 1.0 (where the analog converter accepts signals between 0 and 5V). Therefore, for a 5V signal, the value of the first argument will be 1.0.
- The second argument is a pointer to an array of floating point numbers that store the shim current values to be applied (in Amps).
- The return value is the number of channels (i.e. the size of the float array).

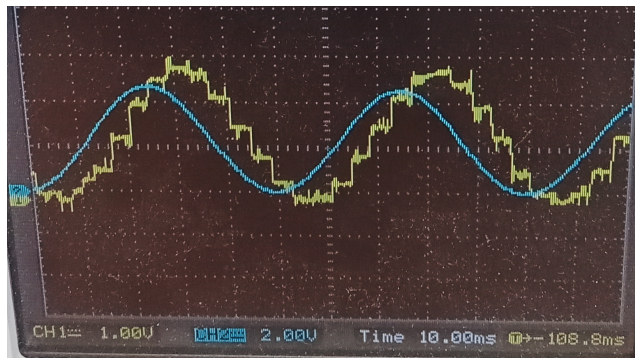
An example source code is given in Appendix A. The example takes the analog input (value between 0.0 and 1.0) and sets the first 8 channels to the same value as the analog input (i.e. output will be a value between 0A and 1A for 8 channels).

The source code can be compiled with a C/C++ compiler. Using CMake, this will simply be adding the following line (if the code is saved in *shimplugin.cpp*):

```
add_library(shimplugin SHARED "shimplugin.cpp")
```

3.2 Performance of Real-time Updates

The shim current values are updated at a rate of 4ms, where the lag between the analog signal input and the output is approximately 6ms. The figure below shows the analog input signal and one of the output signals when using the shimplugin generated with the source code in the Appendix.



3.3 Using Real-time Update

Once the shimplug.dll/.so file has been prepared. The real-time function can be started with the *Sinope* software.

In *Sinope*, click on “Select file” and choose the MR Shim license file *calcshim.lic*. This is typically located in the following path:

%SHIMDIR%\bin\calcshim.lic

where %SHIMDIR” is the main MR Shim folder path, e.g. C:\mrshim.

Next, click the connect button and the amplifiers will start applying the values according to the values calculated using the shimplugin.dll/.so.



4.0 MAINTENANCE

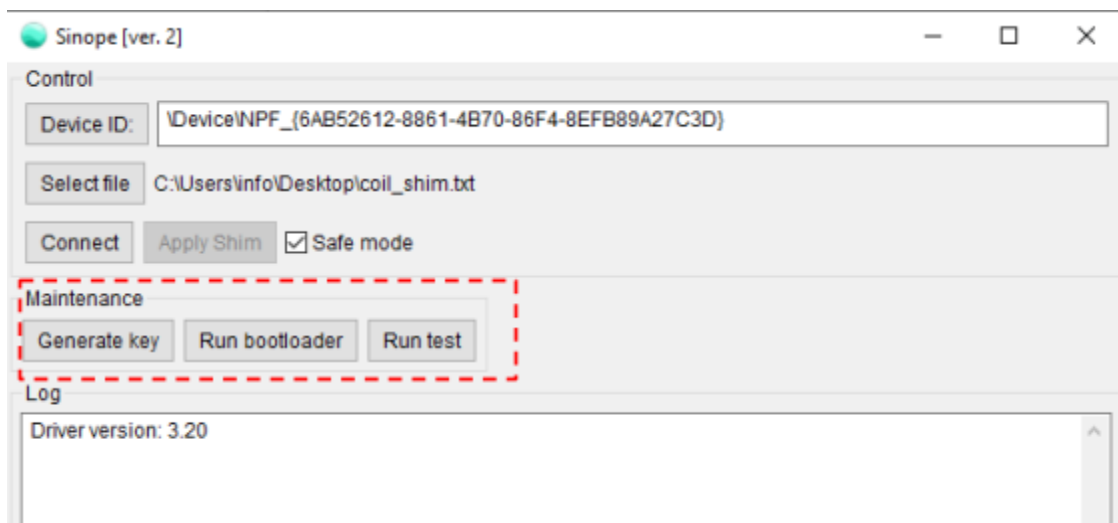
D. MAINTENANCE

4.1 Maintenance Functionality

There are additional functions that are used for MR Shim maintenance. These include:

- “Generate key” - Used to generate a key for the computer. If a license is requested for a computer, this key needs to be sent to *MR Shim GmbH*.
- “Run bootloader” - Used to update the firmware on the *Jupiter* amplifiers.
- “Run test” - Run diagnostic tests.

These functions should not be used by the user, unless specifically instructed by MR Shim personnel. For more information contact info@mrshim.de.



4.2 Support

If you have further difficulties with getting the system to work, please **contact** MR Shim at info@mrshim.de

APPENDIX

APPENDIX

A Shimplugin.cpp

```
#pragma once
#ifdef __cplusplus
    #ifdef _WIN32
        #define EXTERN extern "C" __declspec(dllexport)
    #else
        #define EXTERN extern "C"
    #endif
#else
    #define EXTERN
#endif

#define NUM_CHANNELS 8           // Define number of channels here
float shimvalues[NUM_CHANNELS];

EXTERN int ShimRealtime(float analogValue, float** output) {
    // Custom code here
    for (int i = 0; i != NUM_CHANNELS; ++i) {
        shimvalues[i] = analogValue;
    }

    // DO NOT CHANGE
    *output = &(shimvalues[0]);
    return NUM_CHANNELS;
}

/* shimplugin.cpp end */
```



Manufacturer:
MR Shim GmbH
Burkhardt+Weber Str. 59
72760 Reutlingen
Germany
www.mrshim.de