

simetrix

SIMETRIX DATA INTERFACE

VERSION 8.1

DECEMBER 2016

SIMETRIX DATA INTERFACE

COPYRIGHT © SIMETRIX TECHNOLOGIES LTD. 1992-2016

SIMetrix Technologies Ltd.,
78 Chapel Street,
Thatcham,
Berkshire
RG18 4QN
United Kingdom

Tel: +44 1635 866395

Fax: +44 1635 868322

Email: support@simetrix.co.uk

Web: <http://www.simetrix.co.uk>



Contents

| | | |
|----------|---|----------|
| 1 | Overview | 1 |
| 2 | Installation | 2 |
| 3 | Example | 3 |
| 4 | Writing a Remote Control Application | 5 |

Chapter 1

Overview

This document provides a description of an interface that allows an external application to control SIMetrix and access the simulation data of a running instance of SIMetrix.

The interface may be used with programs written in 'C' or 'C++' or other languages that can call 'C' functions exported in a Windows DLL.

A key part of the interface specification is the .h include file - SxRemoteDataInterface.h which may be found in the install CD or CD web image. Refer to the download page for details.

This document assumes the reader has a knowledge of programming in the 'C' language.

Chapter 2

Installation

To use the interface two files are required. These are the interface DLL, `sxdif.dll` and a 'C' header file, `SxRemoteDataInterface.h`, that defines the interface itself.

The `sxdif.dll` file may be found in the `bin64` folder of the 64 bit SIMetrix installation. A 32 bit version may be found in the `bin` folder. You need to use the architecture (32 bit or 64 bit) that matches the architecture of your application. For example, if your application is 32 bit, you must use the 32 bit `sxdif.dll` found in the `bin` directory. It isn't necessary for the architecture of your application to match the architecture of SIMetrix. So, it is OK for a 32 bit application to remotely control a 64 bit SIMetrix process.

Copy the appropriate version of `sxdif.dll` to the same folder as your executable file.

Copy the `SxRemoteDataInterface.h` header file to a suitable location.

Chapter 3

Example

An example application with source code is available which is a useful starting point for using the sxdif interface. See folder

```
CD:\Script\Remote-Access\RemoteDataInterface\example
```

The source file is source\main.cpp. This is a console mode program that provides a simple remote access to a running instance of SIMetrix. A pre-built version - RDIconsole.exe - is provided. Before using this program, copy the 32 bit sxdif.dll from the version of SIMetrix you are using to the same folder as RDIconsole.exe. To start the program, open a command prompt window, CD to the folder where RDIconsole is located then type RDIconsole at the command line.

To experiment, try the following:

1. Start SIMetrix
2. Start RDIconsole if not already as described above
3. At the RDIconsole prompt, type:
?
4. You should see a list of commands. Now type
OpenSxCommand
This should return without any message
5. Now type
SendCommand Echo Hello World!
You should see the message "Hello World" appear in the SIMetrix command shell window
6. Now open a schematic the example schematic amp.sxsch. You would usually find this here:
C:\Users\<loginname>\Documents\SIMetrix\Examples-81\SIMetrix\General\AMP.sxsch

You can open this file within the SIMetrix environment, or you can issue this command in RDIconsole:

```
SendCommand OpenSchem  
"C:\Users\<loginname>\Documents\SIMetrix\Examples-81\SIMetrix\General\AMP.sxsch"
```

7. Create a netlist for the schematic:
SendCommand Netlist design.net
8. Now run the schematic:
SendCommand Run design.net
9. We will now retrieve some data. Enter this in RDIconsole:
GetGroups

You should see a list of group names displayed. The first will be `global`. Pick the group corresponding to the run you just made. This will probably be `tran1`.

10. Enter this command (assuming `tran1`)

```
GetVectorInfo tran1
```

You should see a list of vector names

11. Pick any of the vector names - say `VOUT` then type

```
GetVectorData tran1 VOUT
```

You should see the data values of the vector displayed

The above shows how to use `RDIconsole`, but this is intended only as a demo program. The source code is in `source\main.cpp` and from there you can see how to use the interface to build your own remote control program.

Chapter 4

Writing a Remote Control Application

Your remote control program must perform these tasks:

1. Load the `sxdif.dll` DLL. Use the Windows API `LoadLibrary`
2. Get the function pointer to "`srd_InitialiseInterface`". Use the Windows API call `GetProcAddress`
3. Call `srd_InitialiseInterface`

```
SRD_Interface srdif ;
int    major = SRD_MAJOR_VERSION,
      minor = SRD_MINOR_VERSION;
SRD_ErrorCode errcode = srd_InitialiseInterface(&srdif, &major, &minor) ;
```

4. The variable `srdif` points to a number of function pointers that can be used to send commands and retrieve data. Before doing anything else, you must call `openInterface` to obtain a handle:

```
SRD_ErrorCode errcode = SRDE_Ok ;
void *handle = NULL ;

errcode = srdif->openInterface(ident, &handle) ;
```

The `handle` variable may be used in subsequent calls to `srdif` functions. Refer to `sxremotedatainterface.h` for full details.

5. To send commands to SIMetrix, use `srdif->sendCommand`. Note that a call to `srdif->openSxCommand` must be made just once before using `srdif->sendCommand`.

Copyright © SIMetrix Technologies Ltd. 1992-2016

SIMetrix 8.1 Data Interface