

History and release notes for the Rohde & Schwarz Vector Signal Generator SM300

Contents

Contents	1
SM300 driver history.....	2
Installation of the instrument driver	5
Instrument address descriptor	5
Getting Started	5
Instrument Identification and Logical Names	5
LabWindows/CVI.....	5
VEE	6
LabWindows/CVI	7
CVI Version	7
Additional Help	7
VXIplug&play Instrument Driver for VEE, C++, C#, Visual Basic, Visual Basic .NET etc.....	7
VEE Version.....	7
C#.....	7
Visual Basic .NET.....	7
Additional Help	7
Additional Information	7
Linux	7
R&S Smart Instruments™ Family300 Basic Programming Guide	7

SM300 driver history		
Revision	Date	Note
1.8.4	09/2008	Modifications: Windows Vista support added in installer
1.8.3	10/2007	Modifications: Changed functions: - Set RF Output State (rssism_setRFOutputState) - version checking added - Set RF EMC State (rssism_setRFEMCState) - version checking added - Get RF EMC State (rssism_getRFEMCState) - version checking added - Configure Ref Oscillator (rssism_configureRefOscillator) - ref. source parameter fixed - Get Ref Oscillator (rssism_getRefOscillator) - ref. source parameter fixed
1.8.2	09/2007	Modifications: Bas and VB files fixed New functions: - Get Max RF Level (rssism_getMaxRFLevel) - Get Max Limit (rssism_getMaxLimit) - Get Max Modulation Depth (rssism_getMaxModulationDepth) Changed help in function: - Configure RF Level (rssism_configRFLevel) - Set RF Level (rssism_setRFLevel) - Set AM Depth (rssism_setAMDepth)
1.8.1	08/2007	Changed function: - Set RF Output State (rssism_setRFOutputState) - Set RF EMC State (rssism_setRFEMCState)
1.8	06/2007	Modifications: Execution Error handling added Added C# wrapper to installer New reserved functions: - Set RF EMC State (setRFEMCState) for Firmware Release >= 2.3 - Get RF EMC State (getRFEMCState) for Firmware Release >= 2.3
1.7	03/2006	Modifications: - Code maintenance, bug fixing
1.6	01/2006	Modifications: - included additional support files for SiControl Library (Lite) - fixed behaviour of switching reference oscillator int / ext (PLL unlock error handling) - Changed functions: Configure RF Level (rssism_configRFLevel), Set RF Level (rssism_setRFLevel), Configure AM Modulation (rssism_configAMModulation), Set AM State (rssism_setAMState) Set AM Depth (rssism_setAMDepth) Output AM Modulated Signal (rssism_exOutAmpl) When amplitude modulation (AM) is set to On, maximum RF level continuously decreases from 13 dBm to 7 dBm with dependency on modulation depth (modulation depth = 0.0 % ... max RF level = 13 dBm, modulation depth = 100.0 % ... max RF level = 7 dBm). - Added new functions to define RF level units: Set RF Level Unit (rssism_setRFLevelUnit) Get RF Level Unit (rssism_getRFLevelUnit) Where following functions are also affected: Configure RF Level (rssism_configRFLevel) Set RF Level (rssism_setRFLevel) Get RF Level (rssism_getRFLevel)

SM300 driver history

Revision	Date	Note
		Set RF Level Limit (rssiism_setRFLevelLimit) Get RF Level Limit (rssiism_getRFLevelLimit) Configure RF Level Sweep (rssiism_configRFLevelSweep) Set RF Start Level (rssiism_setRFStartLevel) Get RF Start Level (rssiism_getRFStartLevel) Set RF Stop Level (rssiism_setRFStopFreqLevel) Get RF Stop Level (rssiism_getRFStopFreqLevel) Set RF Level Sweep Manual (rssiism_setRFLevelSweepManual) Get RF Level Sweep Manual (rssiism_getRFLevelSweepManual) - New functions: Configure Level Correction (rssiism_confLevelCorrection) Configure Level Correction Values (rssiism_confLevelCorrectionValues) Get Level Correction (rssiism_getLevelCorrection) Get Level Correction Values (rssiism_getLevelCorrectionValues)
1.5	12/2004	Modifications: Support for firmware 0.808 or later (respective functions has been changed): - Frequency Offset (0x2011) - Level Offset (0x2021) - Ext PWM Polarity (0x2074) Changed functions: - Get PHM State (rssiism_GetPHMState) (obsolete) - Configure Ref Oscillator (rssiism_configureRefOscilator) (obsolete) - Set AM State (rssiism_setAMState) - Get AM State (rssiism_getAMState) - Set FM External Coupling (rssiism_setFMExtCoup) - Get FM External Coupling (rssiism_getFMExtCoup) New functions: - Get PHM State (rssiism_getPHMState) - Set FM External Coupling (rssiism_setFMExtCoupling) - Get FM External Coupling (rssiism_getFMExtCoupling) - Configure Ref Oscillator (rssiism_configureRefOscillator) - Get Ref Oscillator (rssiism_getRefOscillator) Error handling & status checking improved: - State Checking (rssiism_errorCheckState) changed - Non exported function Check Status (rssiism_checkStatus) changed
1.4	03/2004	Modifications: - device identification and logical names management added - hot plug & unplug support added
1.3.1	03/2004	Modifications: - header file (enum replaced by const) - rssiism_init (help changed)
1.3	11/2003	Modifications: - rssiism_waitForDeviceState waits with 1 ms precision. - Device functions status is read directly from the instrument New functions: - rssiism_setTimeOut - rssiism_getTimeOut- New functions:
1.2	10/2003	Modifications: - Function prototypes from SiControl library are changed (added prefix Si). Single SiControl.dll library is distributed instead of formerly used patched DLLs (usbdrv_w2.dll, usbdrv_w.dll). - Source code is optimized for SiControl library "Rev 3.0, 10/2003".
1.1	09/2003	Modifications: - Distribution files usbio.inf and usbio.sys are renamed to rssi.sys and rssi.inf - VISA library is not needed (vpptype.h is replaced by rssi.h) - SiScan.exe utility returns instrument descriptors of connected

SM300 driver history		
Revision	Date	Note
		devices (distributed with source files)
1.0.1	09/2003	Modifications: <ul style="list-style-type: none">- source code bug fixing and verification- range checking improvements- performance improvements

Installation of the instrument driver

The VXIPnP instrument driver also installs the Windows (XP, 2000) USB drivers and the tool SiScan. Please install the SM300 driver before connecting the instrument.

Instrument address descriptor

The syntax for the instrument descriptor is:
USB::<<vendor Id>::<<product Id>::<<serial number>

where <vendor Id> is 0xaad for Rohde&Schwarz
<product Id> is 0x7 for SM300
<serial number> of the SM300 consists of 6 digits. For example 100003.

Example: "USB::0xaad::0x7::100003::INSTR"

Getting Started

Instrument Identification and Logical Names

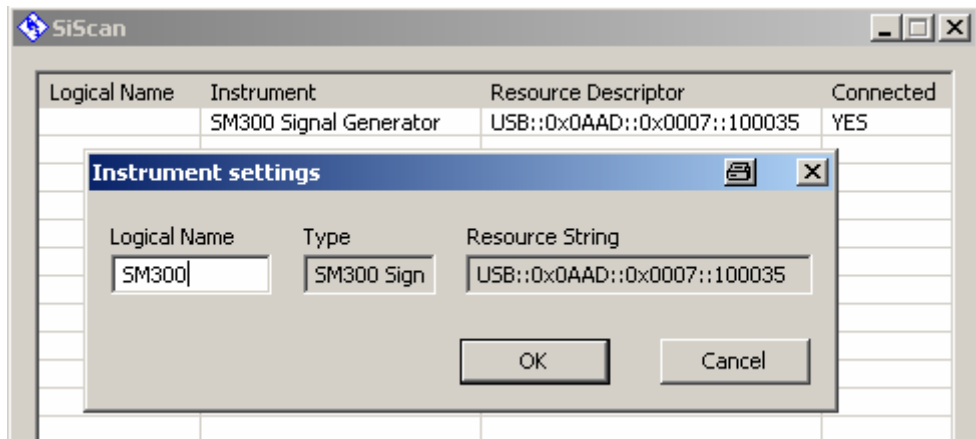
For finding the instrument address on the USB bus easily, use the SiScan application, which is installed with the VXIPnP driver and found in the menu

start > Programs > Rohde & Schwarz > Series300

SiScan provides a table showing the addresses of all connected Smart instruments.

The driver supports logical names. You can pass the logical name instead of the instrument descriptor. For example: "SM300" instead of "USB::0xaad::0x7::100003::INSTR"

Logical names can be configured with the SiScan application.



LabWindows/CVI

To use the LabWindows/CVI driver it is necessary to install the VXIPnP instrument driver first.

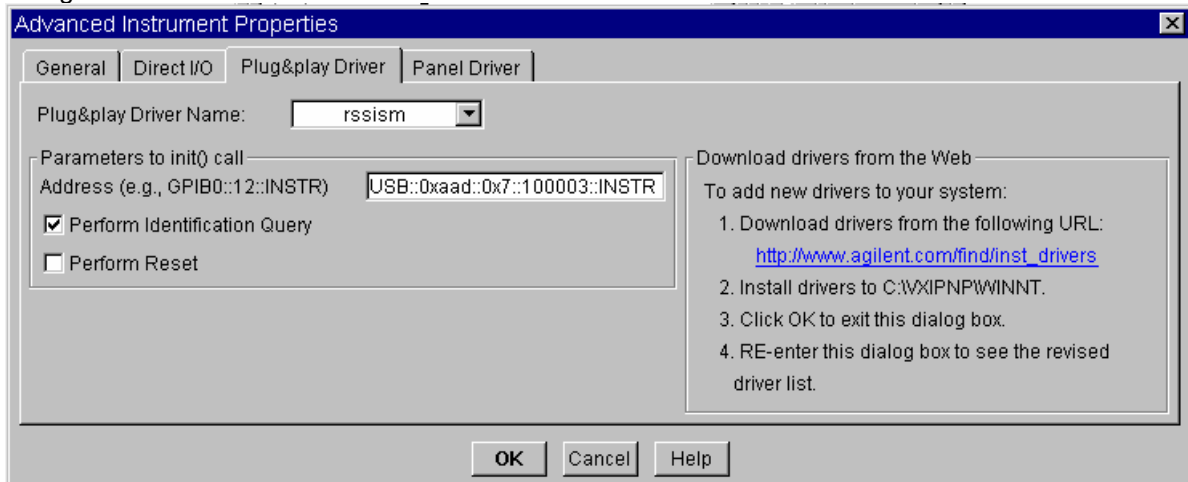
VEE

Set **Plug&Play Driver Name** to **rssism** and **Address** to the SM300 address or the logical name in the menu

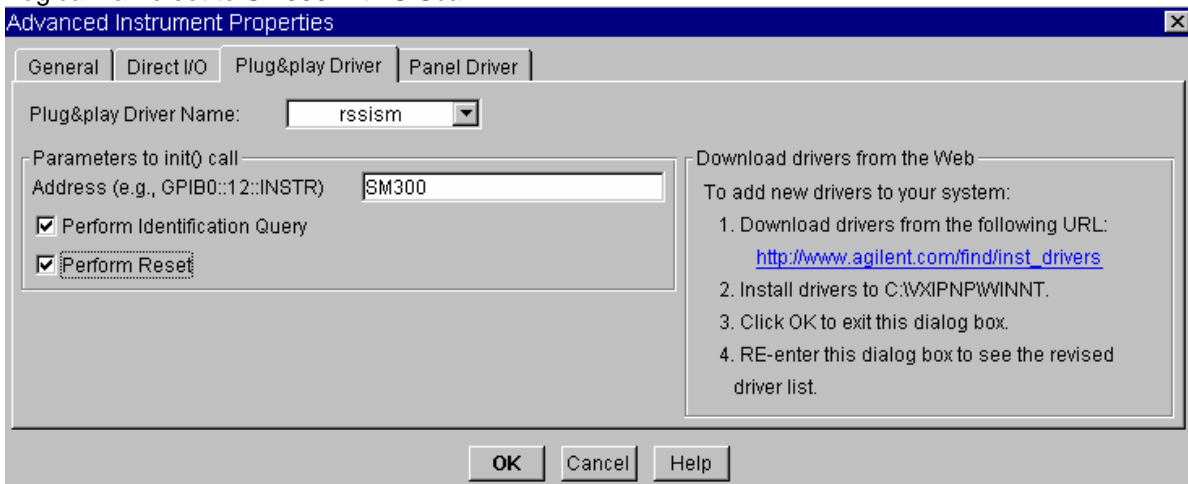
IO > Instrument Manager > Advanced > Plug&Play Driver.

Example :

Using Instrument address:



Logical name set to SM300 with SIScan:



LabWindows/CVI

CVI Version

Use National Instruments LabWindows/CVI 5.5 or later.

Additional Help

The LabWindows/CVI instrument driver consists of a ZIP archive containing the driver sources. In addition, the instrument driver documentation is also included in compressed HTML format (Windows CHM help file) and stored together with the driver sources.

VXIplug&play Instrument Driver for VEE, C++, C#, Visual Basic, Visual Basic .NET etc.

VEE Version

Use VEE 6 or later.

C#

A wrapper is necessary to enable a direct access to the driver DLL.
The rssidm.cs wrapper for C# is automatically installed in the ~\VXIIPnP\WinNT\include directory.

Visual Basic .NET

A wrapper is necessary to enable a direct access to the driver DLL.
The rssidm.vb wrapper for .NET is automatically installed in the ~\VXIIPnP\WinNT\include directory.

Additional Help

In addition, the instrument driver documentation is also included in compressed HTML format (Windows CHM help file) and stored together with the driver sources in the ~\VXIIPnP\WinNT\rssidm directory.

Additional Information

For more information regarding the VXIIPnP instrument drivers, please read the readme.txt file that comes with each driver.

Linux

Drivers for Linux are available - Please contact Rohde & Schwarz Customer Support Center

R&S Smart Instruments™ Family300 Basic Programming Guide

The instrument drivers allow you to access instruments from various programming environments under Microsoft Windows XP/2000. The "Smart Instruments™ Programming Guide" deals with programming the Smart Instruments™ Family300 based on these drivers from different programming languages (C/C++, Visual Basic, LabVIEW, LabWindows/CVI).

Download the R&S Smart Instruments™ Family300 Basic Programming Guide:

<http://www.rohde-schwarz.com/appnote/1MA73.html>