

# SIS3316 16 Channel VME Digitizer

## Software Instruction Guide “SIS3316\_root\_gui”

SIS GmbH  
Harksheider Str. 102A  
22399 Hamburg  
Germany

Phone: ++49 (0) 40 60 87 305 0  
Fax: ++49 (0) 40 60 87 305 20

email: [info@struck.de](mailto:info@struck.de)  
<http://www.struck.de>

Version: SIS3316-M-Application-Software-Root-Gui-V100.doc as of  
12.08.2015

## Revision Table:

Revision	Date	Modification
1.00	12.08.2015	Initial release

## Table of contents

1	Introduction .....	4
1.1	Operating System .....	4
1.2	Graphical Interface .....	4
1.3	Communication Interface.....	5
1.4	Executables .....	5
1.4.1	Windows 7 (Windows 8).....	5
2	Build the program “sis3316_root_gui” .....	6
2.1	Windows 7 (Windows 8) .....	6
2.1.1	Open the project “sis3316_root_gui.sln” with VisualC++ 2010.....	7
2.1.2	VC10 Project Properties.....	8
2.1.2.1	Include path settings.....	8
2.1.2.2	Library path settings.....	10
2.1.3	VC10 Build project .....	11
2.1.4	VC10 Run project .....	12
2.2	Linux (Scientific 6.5).....	13
2.2.1	Eclipse Project Directory Structure.....	13
2.2.2	Build with Eclipse.....	15
2.2.3	Build with a terminal shell.....	16
2.2.4	Run from a terminal shell.....	16
3	Run the program “sis3316_root_gui” .....	17
3.1	FPGA Programming Menu .....	19
3.1.1	VME FPGA.....	19
3.1.2	ADC FPGA .....	20
3.1.3	FPGA reboot.....	21
3.2	SIS3316 Test Menu .....	22
3.2.1	Tab Frames .....	25
3.2.1.1	Sample Control .....	25
3.2.1.2	Display Control.....	26
3.2.1.3	Polarity .....	27
3.2.1.4	Gain/Offset .....	28
3.2.1.5	Trigger.....	29
3.2.1.6	Energy .....	30
3.2.1.7	Sample Clock.....	31
3.2.1.8	NIM Outputs.....	32
3.2.2	Save/Load Configuration parameters .....	33
3.2.3	Write Data File Format.....	34
4	Index.....	35

## 1 Introduction

This document will guide you through the build process of the software project “sis3316\_root\_gui” which is helpful to demonstrate the use of the software and libraries, that are shipped with the module and to facilitate initial sampling with the digitizer and display of the acquired signals. It is not optimized for speed, data rate and special applications like gamma-, neutron/gamma and .... It illustrates a lot of capabilities of the sis3316 like trigger generation, trigger selection, ..., however and it can demonstrate the excellent resolution (output code histograms, FFT).

All project files and libraries are available in source code.

The software allows you to:

- update the FPGA Firmware
- configure the SIS3316 (analog offset, input range, ...)
- start sampling: software trigger, external trigger, internal trigger
- graph: display signal(s)
- graph: display Trigger FIR filter trapezoid
- graph: display Energy FIR filter trapezoid
- graph: display FFT
- histogram: display ADC output code
- save acquired Hits/Events to file(s)
- load/save configuration data

### 1.1 Operating System

The software project “sis3316\_root\_gui” is based on the CERN ROOT package and supports the three platforms:

- Windows (WIN7) built with the development tool VisualC++ 2010 (VC10)
- Scientific Linux 6.6 built with the development tool Eclipse (3.6.1)
- Mac OS 10.9.4 built with the development tool Eclipse (Kepler Service Release 2)

The OS is selected with “#define(s)” in the file project\_system\_define.h.

**Note:** at this point in time ROOT 6.x does not support Windows (yet?)

### 1.2 Graphical Interface

The program package ROOT (<http://root.cern.ch>) is used as Graphical User Interface (GUI) and to display graphs and histograms.

To build/run the program, ROOT has to be installed. The projects from the DVD are linked with ROOT version 5.34.18.

The user can start program “sis3316\_root\_gui” immediately provided that ROOT version 5.34.18 is installed.

### 1.3 Communication Interface

The software provides several interface options to access SIS3316 on board resources (control, write and read) from your computer:

- VME Interface sis3100/sis310x (optical interface)
- VME Interface sis3150 (USB2 interface)
- VME Interface sis3153 (USB3 interface)
- Ethernet interface (UDP)

The interface is selected with a “#define” in the file `project_interface_define.h`.

The user can replace the SIS-interface files (`vme_interface_class.h`, `sis1100w_vme_class.h/cpp`, `sis3150w_vme_class.h/cpp`, `sis3153w_vme_class.h/cpp`, `sis3316_ethernet_access_class.h/cpp`) with his own interface files to build the program to run with other VME interfaces or single board computers (SBC).

### 1.4 Executables

#### 1.4.1 Windows 7 (Windows 8)

The user can start the program “`sis3316_root_gui.exe`” immediately provided that:

- ROOT 5.34.18 is installed
- VME crate powered with at least one Struck interface and at least one SIS3316
- And the following files are in the same directory:
  - `sislogo.bmp`
  - `libfftw3-3.dll`

Depending on the used interface you will find the executable in the following directories:

- `sis3316_root_gui_ethernet`
- `sis3316_root_gui_sis310x`
- `sis3316_root_gui_sis3150usb` (USB2)
- `sis3316_root_gui_sis3153usb` (USB3)

## 2 Build the program “sis3316\_root\_gui”

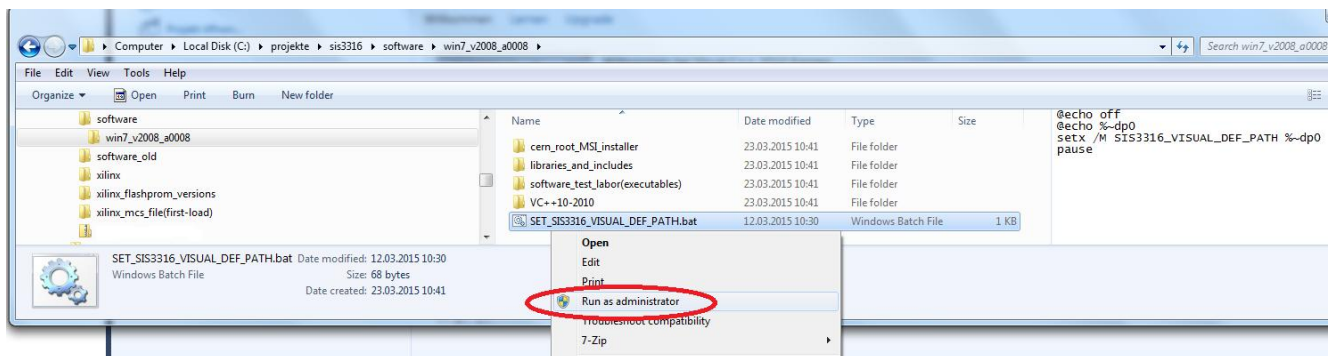
The development tools **VisualC++ 2010** and **Eclipse 3.6.1** are used to build (compile and link) the executable. Valid settings for Include and Library paths are needed for the development tools.

### 2.1 Windows 7 (Windows 8)

The Development Tool **VisualC++ 2010** is used to build the project.

Copy the directory “sis3316/software/win7\_v200A\_a000A” from the DVD to your local disk. For example to c:/projects/sis3316/software.

Execute the batch file “SET\_SIS3316\_VISUAL\_DEF\_PATH.bat” as administrator. It sets the environment variable SIS3316\_VISUAL\_DEF\_PATH with the path of the project. **VisualC++ 2010** uses this variable for the include and library path settings.

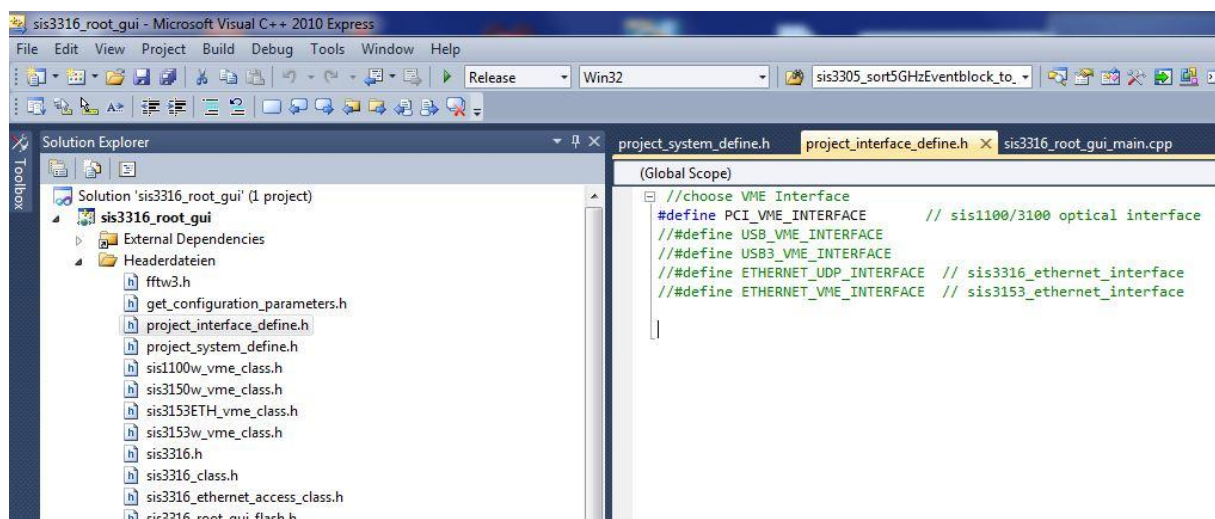


### 2.1.1 Open the project “sis3316\_root\_gui.sln” with VisualC++ 2010.

This project contains the C++ files and the property-settings to run with one of the following communicate interfaces:

- sis3316-DT Ethernet/UDP interface (sis3316\_ethernet\_access\_class.cpp /.h)
- sis310x optical VME interface (sis1100w\_vme\_class.cpp /.h)
- sis3150 USB VME interface (sis3150w\_vme\_class.cpp /.h)
- sis3153 USB3 VME interface (sis3153w\_vme\_class.cpp /.h)
- sis3153 Ethernet/UDP VME interface (sis3153ETH\_vme\_class.cpp /.h)

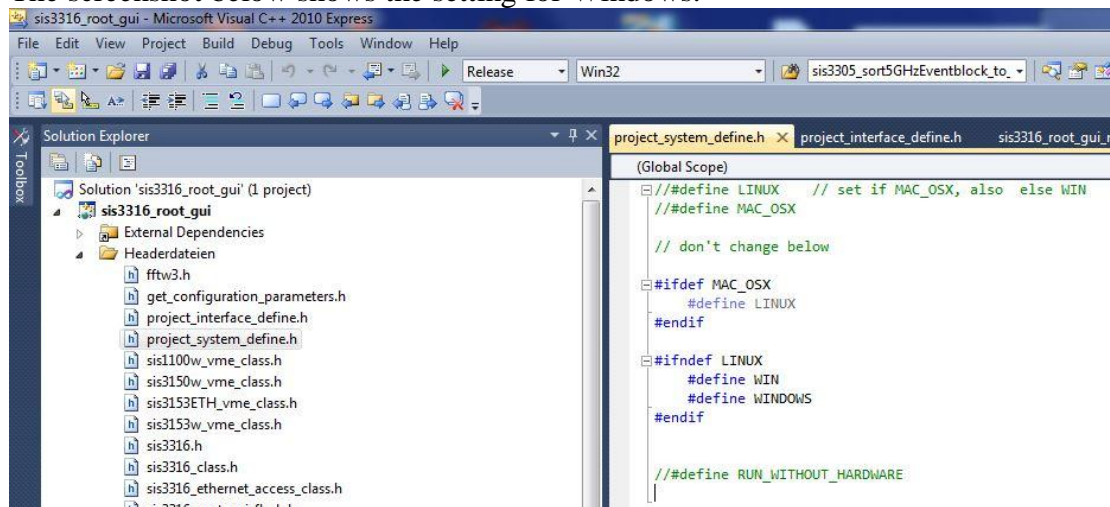
The interface is selected with a “#define” in the file “project\_interface\_define.h”. The screenshot below shows the selection of the SIS1100/SIS310x optical PCIe to VME Interface.



The unused interface files have to be removed from the project in case that the drivers of these interfaces are not installed on the computer (except ETHERNET\_UDP\_INTERFACE).

The OS (operating system) is with the file “project\_system\_define.h”.

The screenshot below shows the setting for Windows.

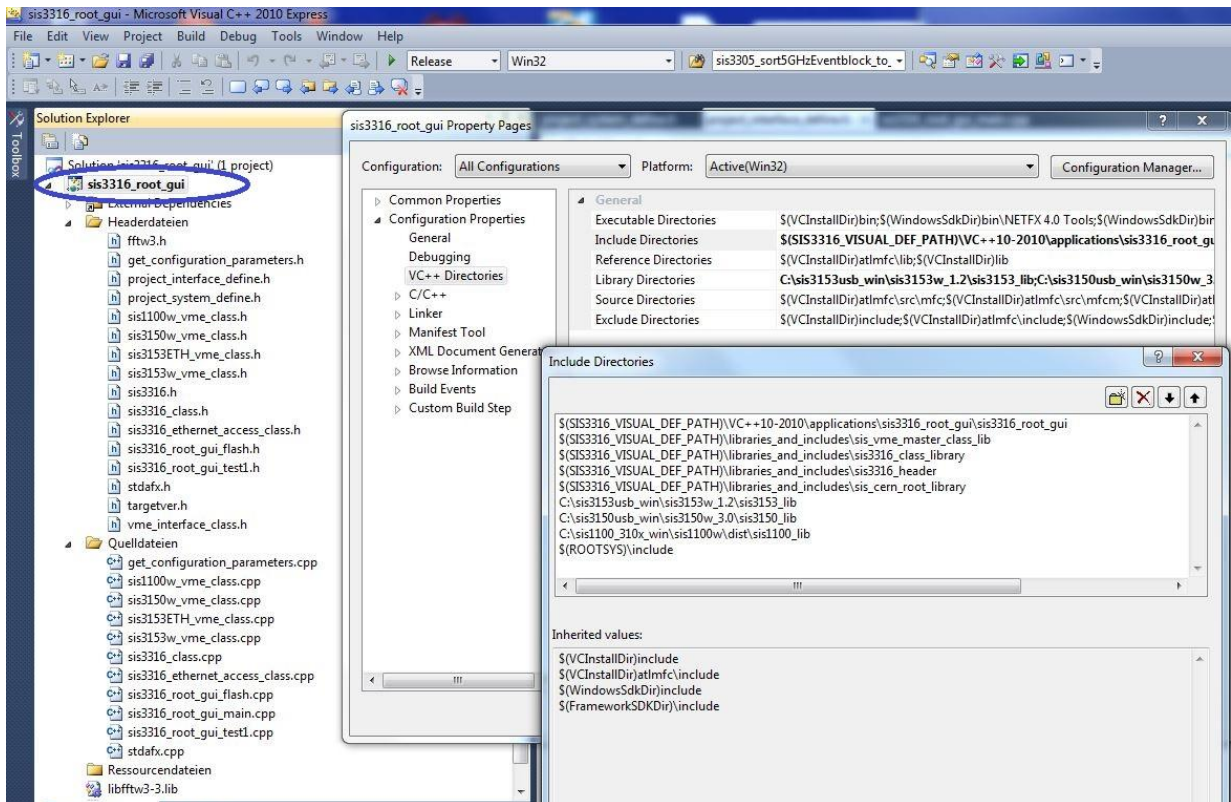




## 2.1.2 VC10 Project Properties

### 2.1.2.1 Include path settings

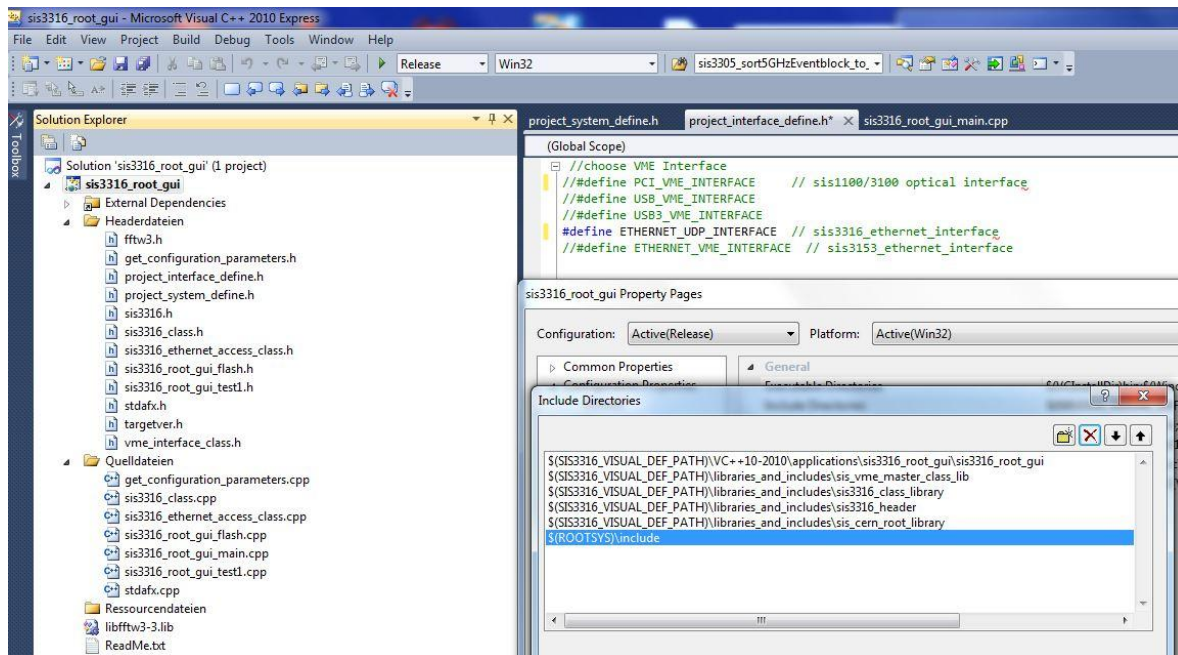
Select the project “sis3316\_root\_gui” (marked in blue) and make a right click. A window pop up and click on Properties (Eigenschaften). The property-window pops up.



The above project contains the Communication interface files and the project settings of the valid Include path settings for four interfaces. The communication interface files and their settings must be deleted if the corresponding driver is not installed (except ETHERNET\_UDP\_INTERFACE).

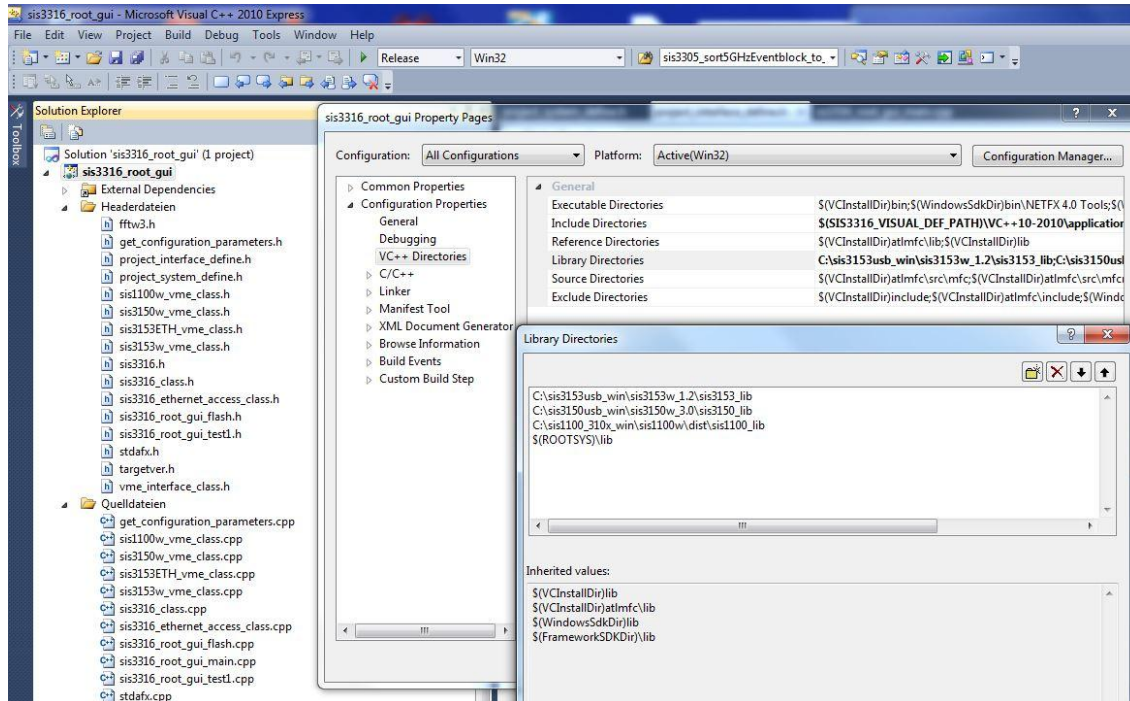


The bottom project (copied from sis3316-DT directory) contains the Communication interface files and the project settings of the valid Include path setting for the Ethernet/UDP interface, only.

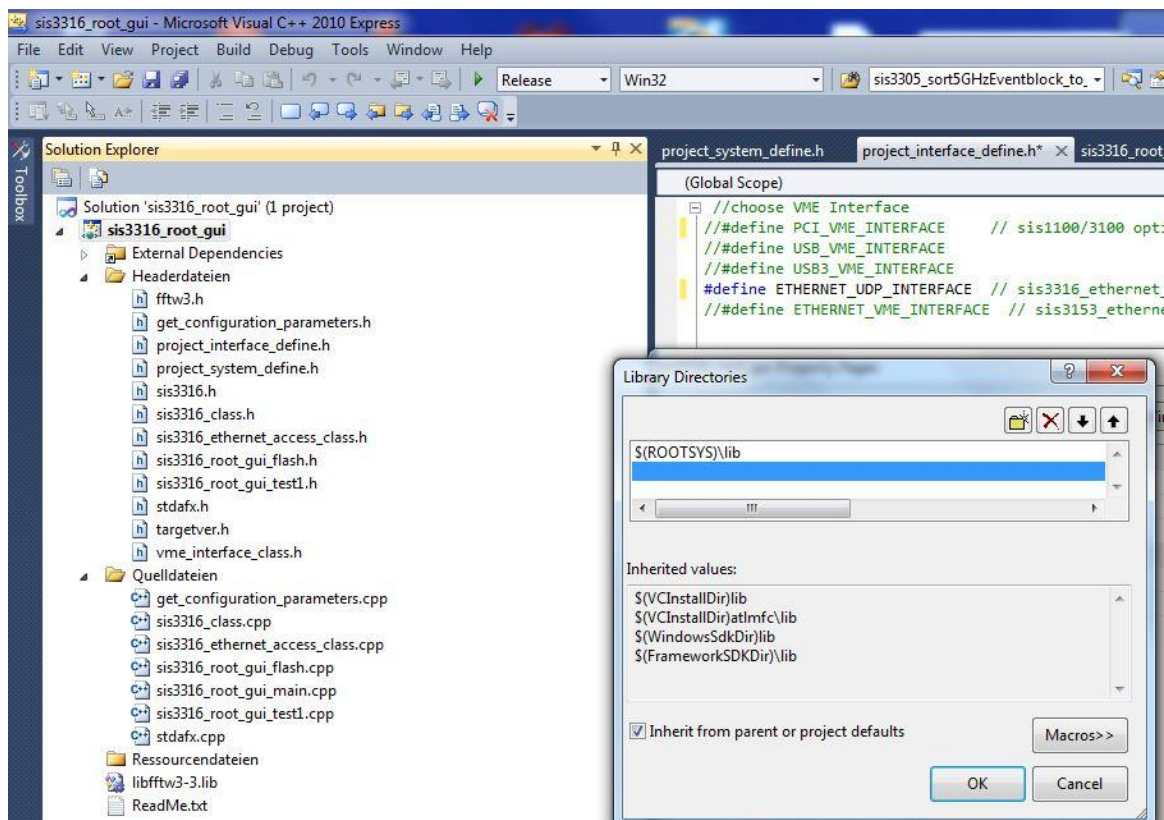


### 2.1.2.2 Library path settings

This project setting contains the Library paths of all SIS interfaces.

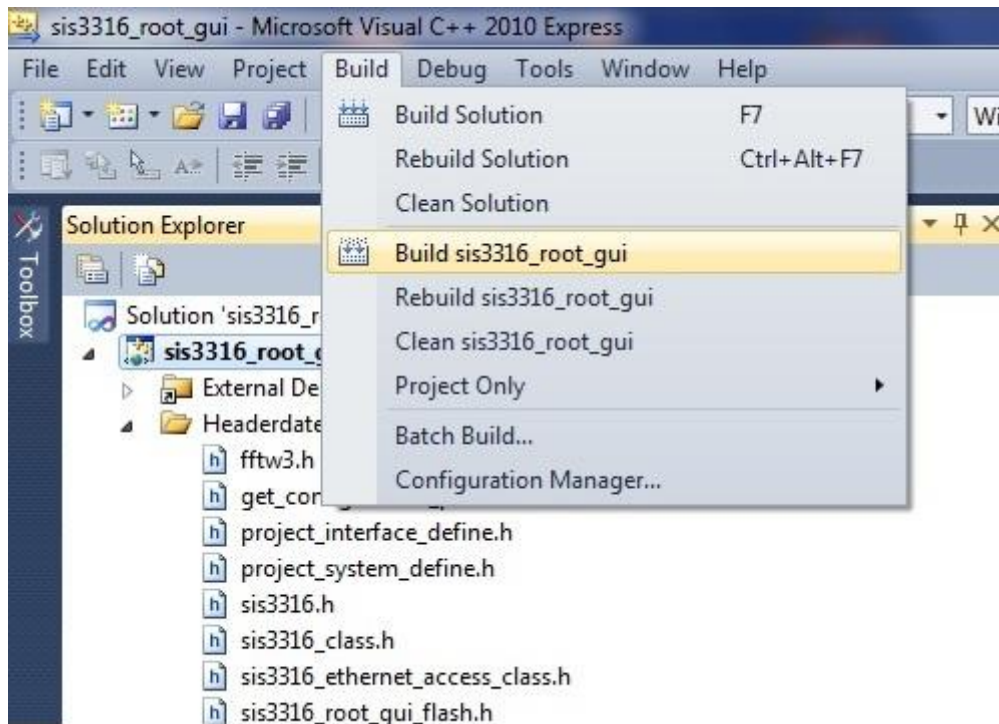


The bottom project (copied from sis3316-DT directory) shows the Library path setting of the customized project for the sis3316-DT Ethernet/UDP interface.



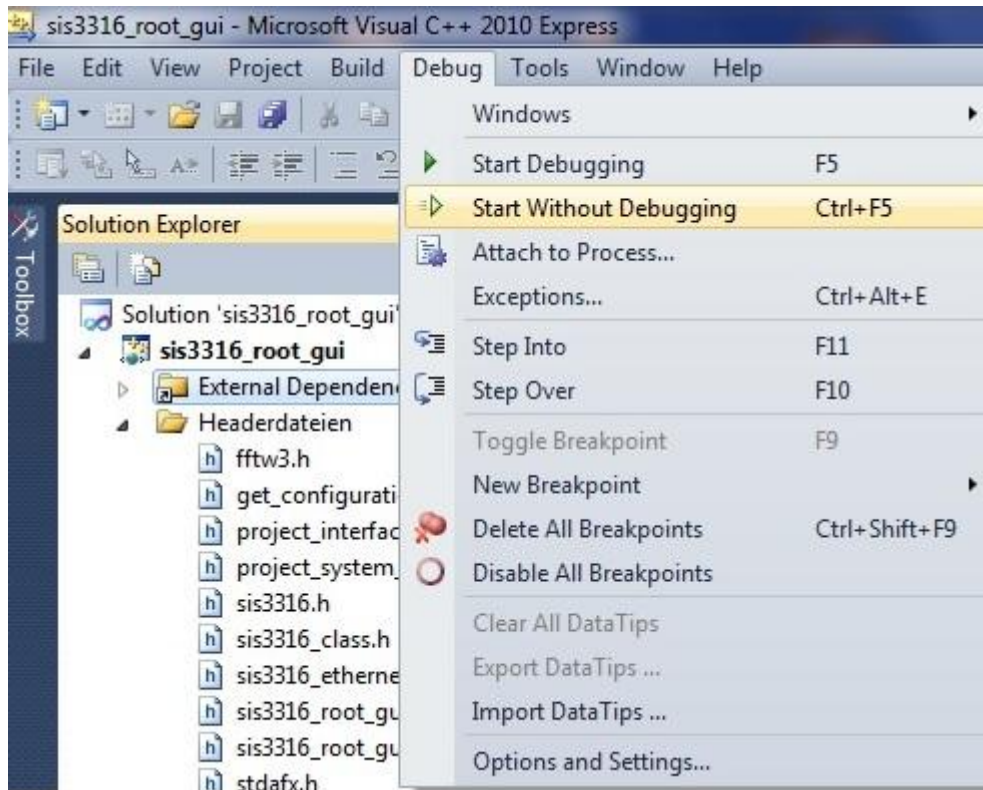
### 2.1.3 VC10 Build project

Select “Build/Build sis3316\_root\_gui”.



## 2.1.4 VC10 Run project

Select “Debug/Start Without Debugging” to start the program.



Or start the program from a DOS box.

```
ROOT session
C:\projekte\sis3316\software\VC++10-2010\applications\sis3316_root_gui\Release>
C:\projekte\sis3316\software\VC++10-2010\applications\sis3316_root_gui\Release>
C:\projekte\sis3316\software\VC++10-2010\applications\sis3316_root_gui\Release>dir
Datenträger in Laufwerk C: ist OS
Volumeserienummer: 100D-3E63

Verzeichnis von C:\projekte\sis3316\software\VC++10-2010\applications\sis3316_root_gui\Release

10.08.2015  14:54    <DIR>          .
10.08.2015  14:54    <DIR>          ..
26.07.2011  20:54             2.201.931 libfftw3-3.dll
10.08.2015  10:47             277.504 sis3316_root_gui.exe
14.01.2005  17:21             114.596 sislogo.bmp
               3 Datei(en),       2.594.031 Bytes
               2 Verzeichnis(se), 4.819.927.040 Bytes frei

C:\projekte\sis3316\software\VC++10-2010\applications\sis3316_root_gui\Release>sis3316_root_gui.exe
```



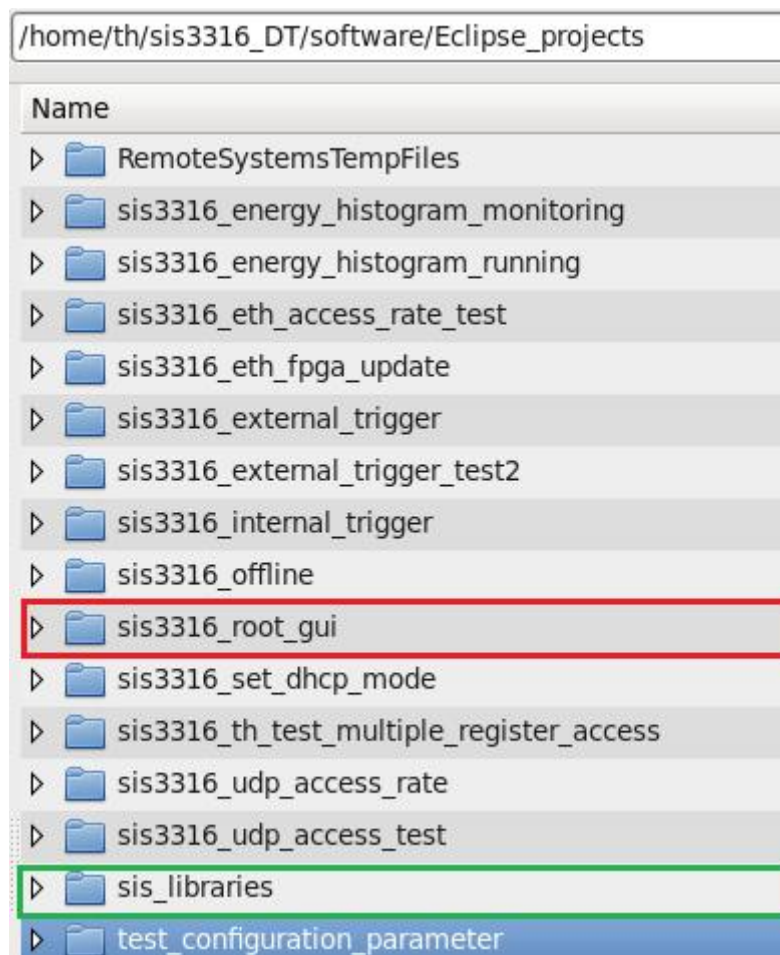
## 2.2 Linux (Scientific 6.5)

The development tool **Eclipse 3.6.1** is/was used to build the project(s). Each project contains a makefile in the “Release” subfolder. Therefore it is possible to build the project from a terminal shell, too.

Copy the file “sis3316\_DT\_xxxx.tar.gz to your local disk and unpack it.

### 2.2.1 Eclipse Project Directory Structure

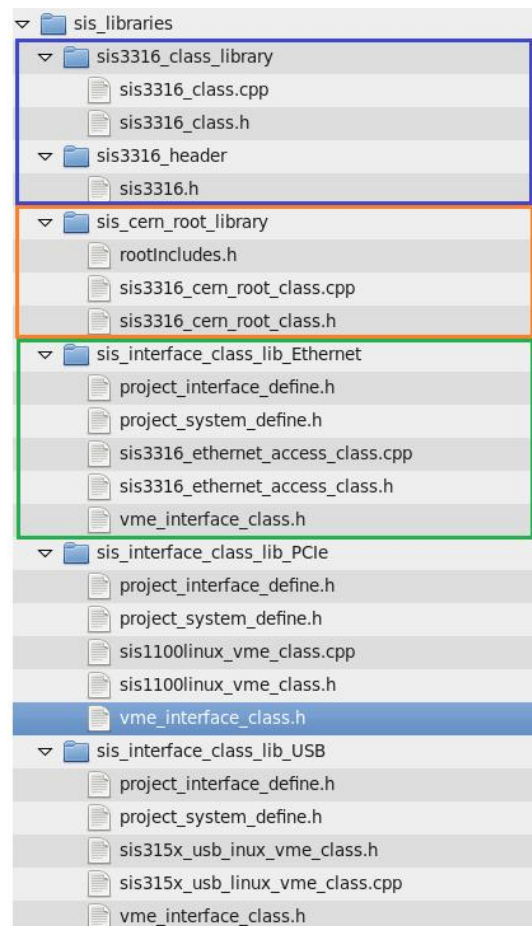
The “../Eclipse\_projects” directory contains among other subfolders the two subfolders “sis3316\_root\_gui” and “sis\_libraris”. This document addresses these two subfolders, only.



The “sis3316\_root\_gui/src” folder contains all source files (\*.cpp, \*.h) which are necessary to build an executable (make) of the project. The marked files (blue, orange and green) are copied from the subfolders “sis\_libraries/sis3316\_class\_library”, “sis\_libraries/sis\_cern\_root\_library” and “sis\_libraries/sis\_interface\_class\_lib\_Ethernet”. This project uses the Communication Interface “Ethernet”.

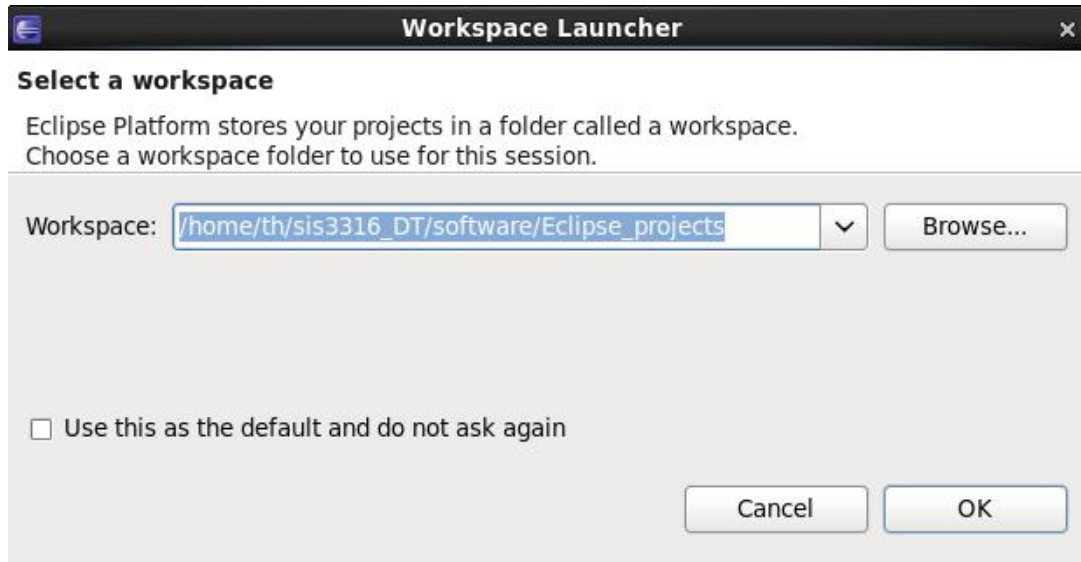
To build an executable which uses the SIS1100/SIS310x optical Interface you have to exchange the green marked files (copied from the subfolder “sis\_libraries/sis\_interface\_class\_lib\_Ethernet”) with the corresponding files of the subfolder “sis\_libraries/sis\_interface\_class\_lib\_PCIE”.

To build an executable which uses the SIS3150 or SIS3153 USB2/USB3 Interface you have to exchange the green marked files (copied from the subfolder “sis\_libraries/sis\_interface\_class\_lib\_Ethernet”) with the corresponding files of the subfolder “sis\_libraries/sis\_interface\_class\_lib\_USB”.

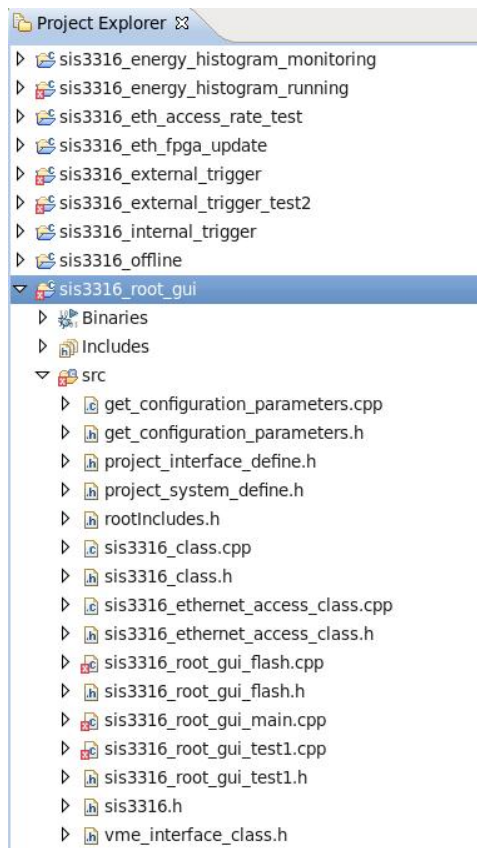


## 2.2.2 Build with Eclipse

Open the workspace with **Eclipse**.



Make a right click on the project “sis3316\_root\_gui” (blue marked) and click on “Build Project”.





### 2.2.3 Build with a terminal shell

Type “make clean” and then type “make sis3316\_root\_gui”.

```
[th@linux-th Eclipse_projects]$ ls
RemoteSystemsTempFiles      sis3316_eth_fpga_update      sis3316_offline              sis3316_udp_access_rate
sis3316_energy_histogram_monitoring  sis3316_external_trigger      sis3316_root_gui            sis3316_udp_access_test
sis3316_energy_histogram_running      sis3316_external_trigger_test2  sis3316_set_dhcp_mode        sis_libraries
sis3316_eth_access_rate_test          sis3316_internal_trigger      sis3316_th_test_multiple_register_access  test_configuration_parameter

[th@linux-th Eclipse_projects]$ cd sis3316_root_gui/
[th@linux-th sis3316_root_gui]$ cd Release/
[th@linux-th Release]$ make clean
rm -rf ./src/get_configuration_parameters.o ./src/sis3316_class.o ./src/sis3316_ethernet_access_class.o ./src/sis3316_root_gui_flash.o ./src/sis3316_root_gui_test1.o ./src/get_configuration_parameters.d ./src/sis3316_class.d ./src/sis3316_ethernet_access_class.d ./src/sis3316_root_gui_flash.d ./src/sis3316_root_gui_test1.d
[th@linux-th Release]$ make sis3316_root_gui
Building file: ./src/get_configuration_parameters.cpp
Invoking: GCC C++ Compiler
g++ -I/home/th/root/include -O3 -Wall -c -fmessage-length=0 -MMD -MP -MF"src/get_configuration_parameters.d" -MT"src/get_configuration_parameters.o" -o"src/get_configuration_parameters.o" ./src/get_configuration_parameters.cpp
Finished building: ./src/get_configuration_parameters.cpp

.....

Building target: sis3316_root_gui
Invoking: GCC C++ Linker
g++ -L/home/th/root/lib -L/usr/local/lib -o"sis3316_root_gui" ./src/get_configuration_parameters.o ./src/sis3316_class.o ./src/sis3316_ethernet_access_class.o ./src/sis3316_root_gui_flash.o ./src/sis3316_root_gui_test1.o -lfftw3 -lCint -lGraf -lGui
Finished building target: sis3316_root_gui
[th@linux-th Release]$
```

### 2.2.4 Run from a terminal shell

Type “./sis3316\_root\_gui”.

```
[th@linux-th Release]$
[th@linux-th Release]$
[th@linux-th Release]$ ./sis3316_root_gui
█
```

### 3 Run the program “sis3316\_root\_gui”

Start the program “sis3316\_root\_gui” and the “main control panel” will appear (left screenshot).

This example shows the “main control panel” of the sis3316\_root\_gui which uses the SIS1100/SIS310x VME Interface.

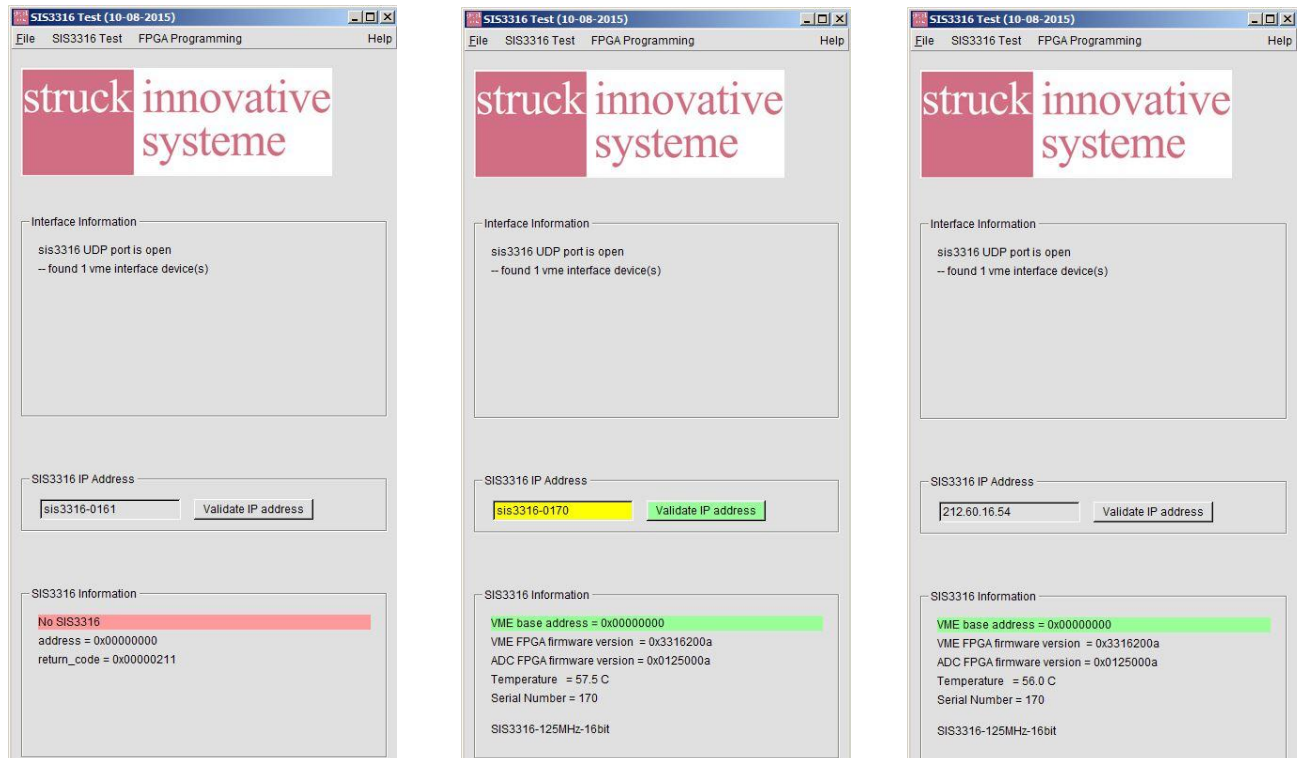
Type the SIS3316 VME base address into the “SIS3316 VME Base Address” box and you will get the “SIS3316 Information” (right screenshot) if the address is valid.



Start the program “sis3316\_root\_gui” and the “main control panel” will appear (left screenshot).

This example shows the “main control panel” of the sis3316\_root\_gui which uses the sis3316 Ethernet Interface.

Type the SIS3316 IP address into the “SIS3316 IP address” box and click the button “Validate IP address”. If the address is valid you will get the “SIS3316 Information” (right screenshot). Instead of the IP address it also possible to use the Hostname, provided that the DHCP feature is enabled and a Name-Server is active (middle screenshot).



### 3.1 FPGA Programming Menu

Select “FPGA Programming” in the menu bar and select “FPGA Programming Menu”.

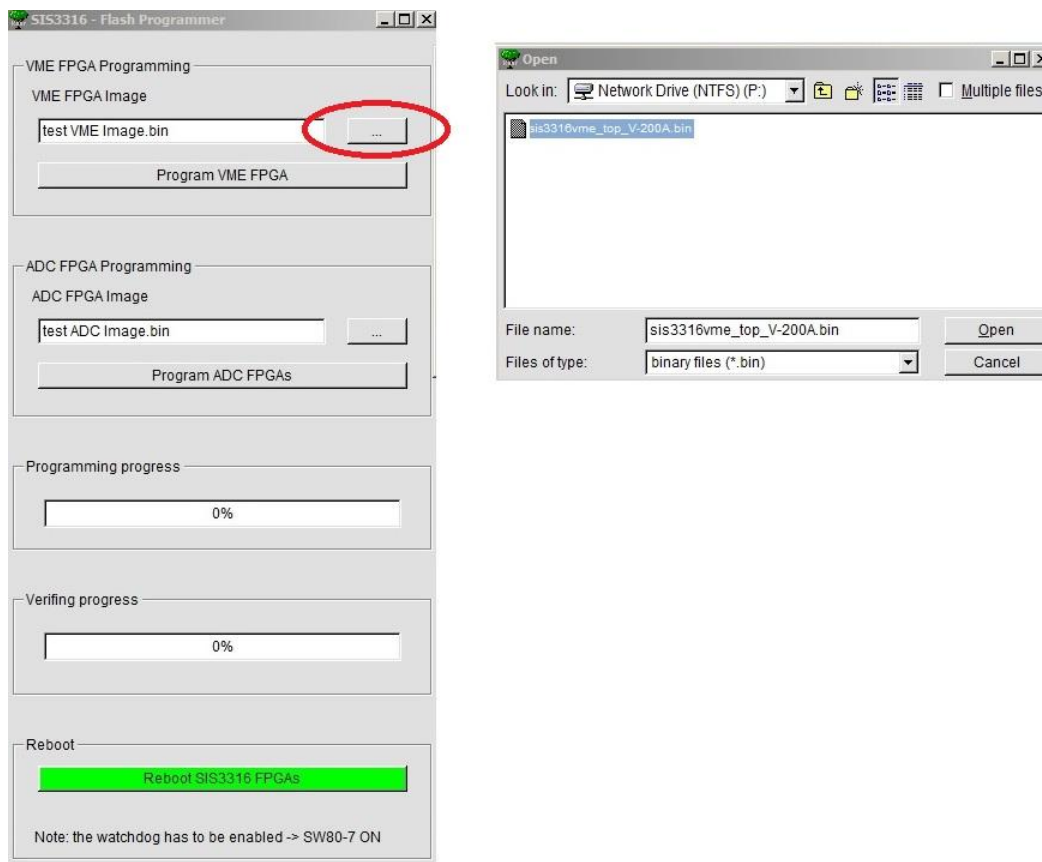


The “FPGA Programming Menu” canvas will appear.

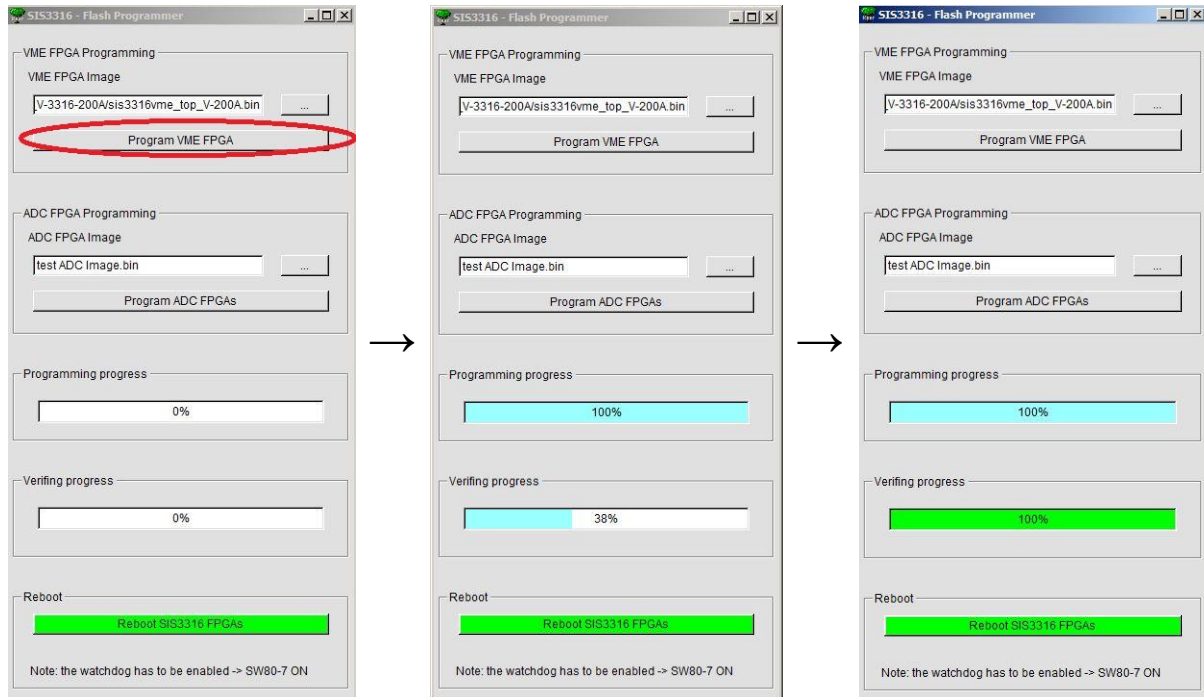
#### 3.1.1 VME FPGA

Select “...” button (marked in red) to get a browser window to search for and select the VME FPGA image file (\*.bin).

Note: be sure to load a VME FPGA image file!



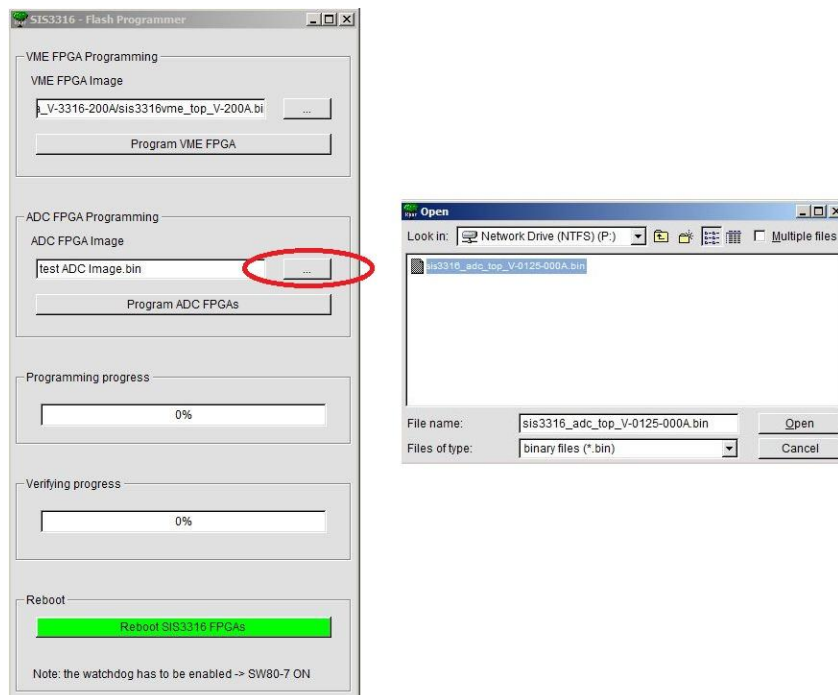
Click on “Program VME FPGA” button (marked in red).  
Programming and verification will take a couple of minutes.



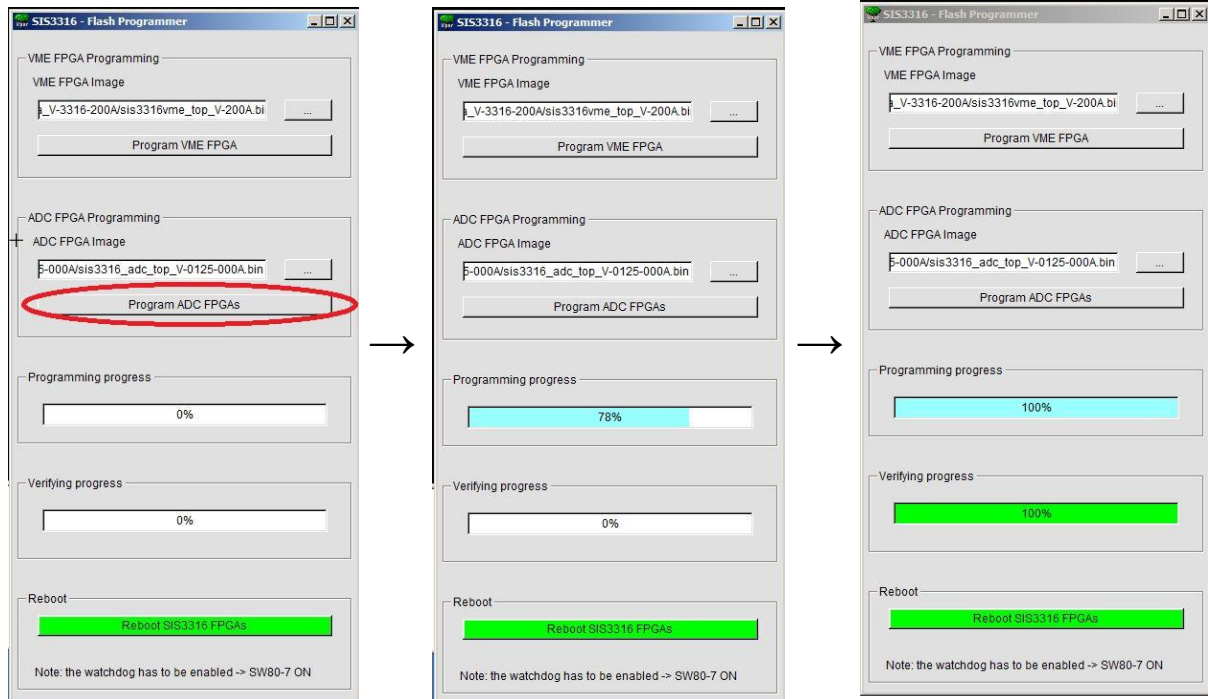
The 100% and the green color of the Verifying progress bar indicates that the programming of the VME FPGA image into the Flashprom has completed successfully.

### 3.1.2 ADC FPGA

Select “...” button (marked in red) to get a browser window to search for and select the ADC FPGA image file (\*.bin).



Click on “Program ADC FPGA” button (marked in red).  
Programming and verification will take a couple of minutes.



The 100% and the green color of the Verifying progress bar indicates that the programming of the VME ADC Image into the Flashprom has completed successfully.

### 3.1.3 FPGA reboot

The FPGAs have to be rebooted after the programming of the image files into the Flashproms was done properly. The reboot can be done either by a power down / power up cycle of the module or Click on “Reboot SIS3316 FPGAs” button (marked in red).



**Note:** Refer to the JTAG firmware installation procedure as described in the SIS3316 user manual in case of corrupted/erased Flashprom contents.

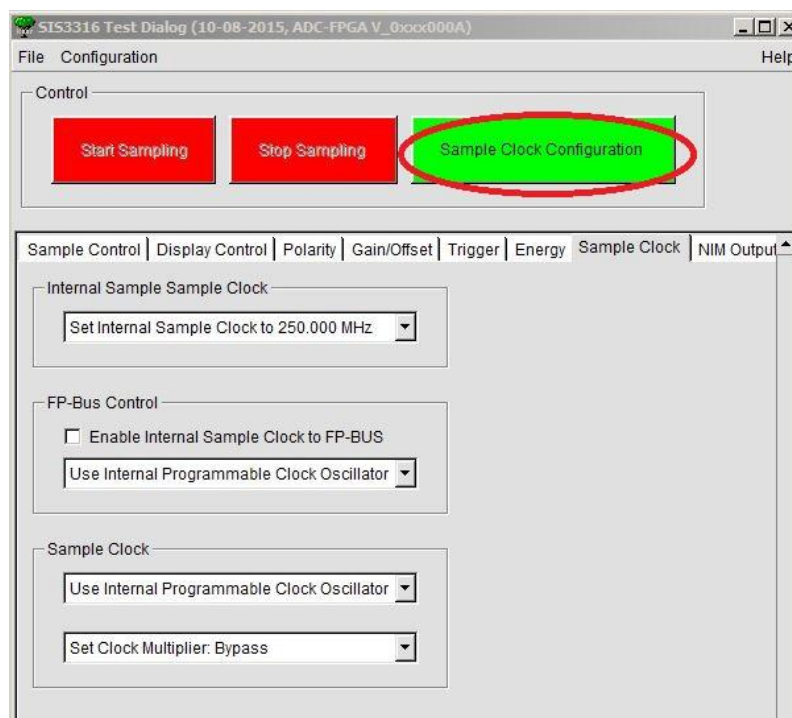


### 3.2 SIS3316 Test Menu

Select “SIS3316 Test” in the menu bar and select “Test1”.



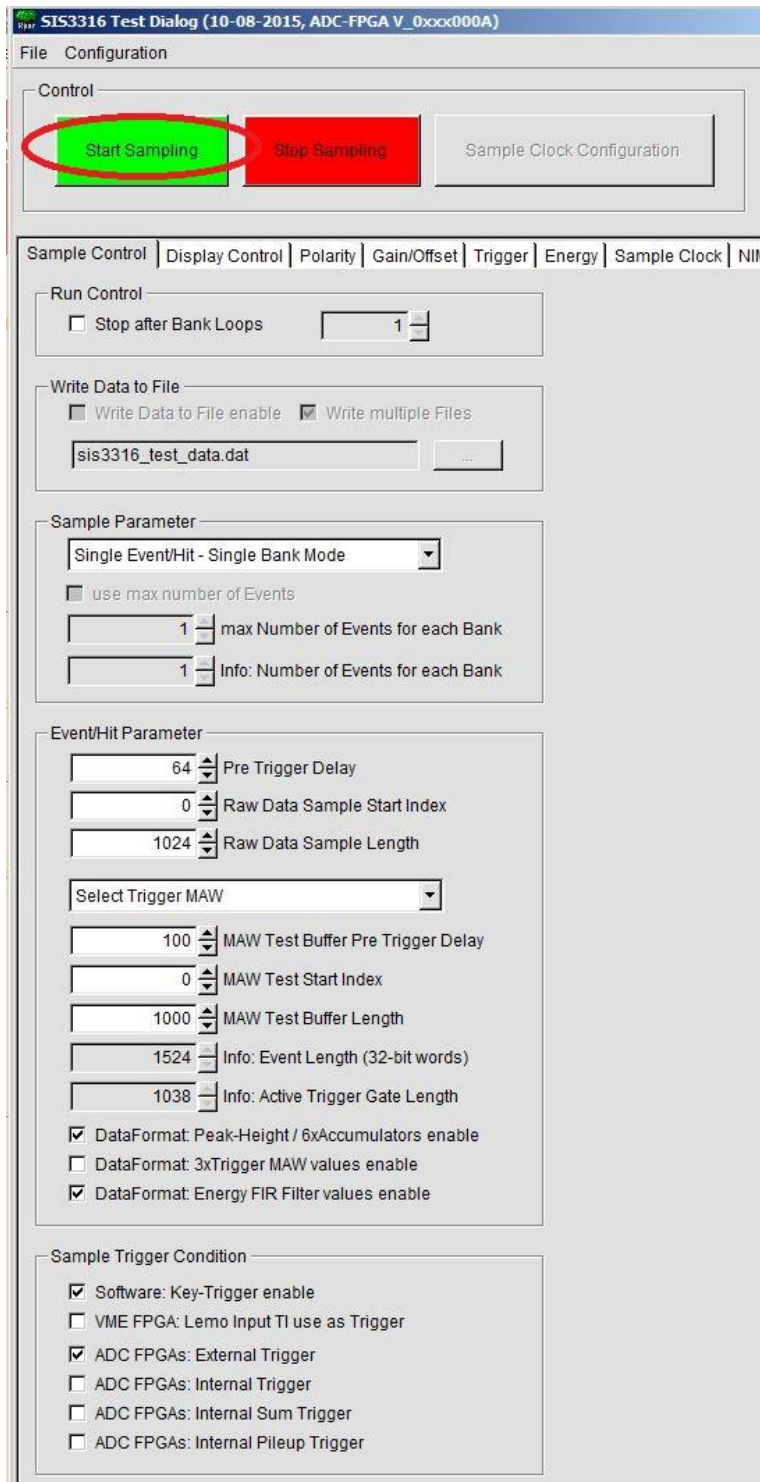
The “SIS3316 Test Dialog” canvas will appear with “Sample Clock” tab frame.



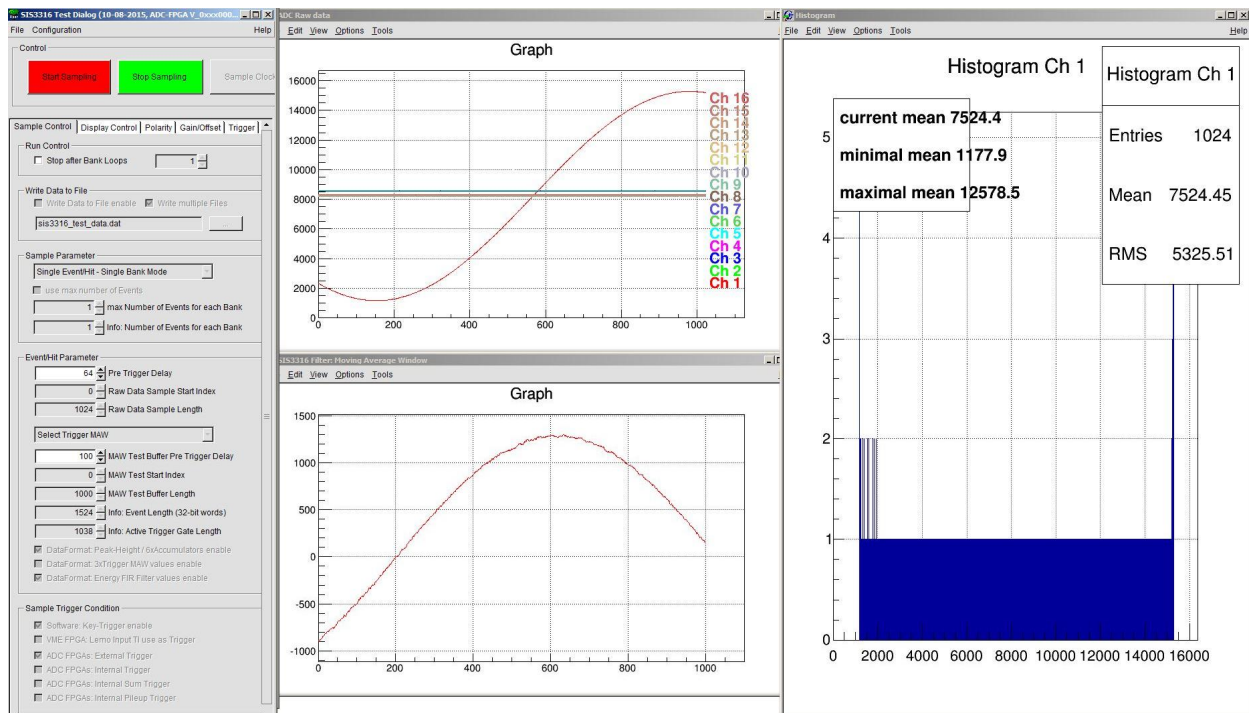
Define the required sample clock and click on “Sample Clock Configuration” button (marked in red). The SIS3316 will be reset and the sample clock will be configured.



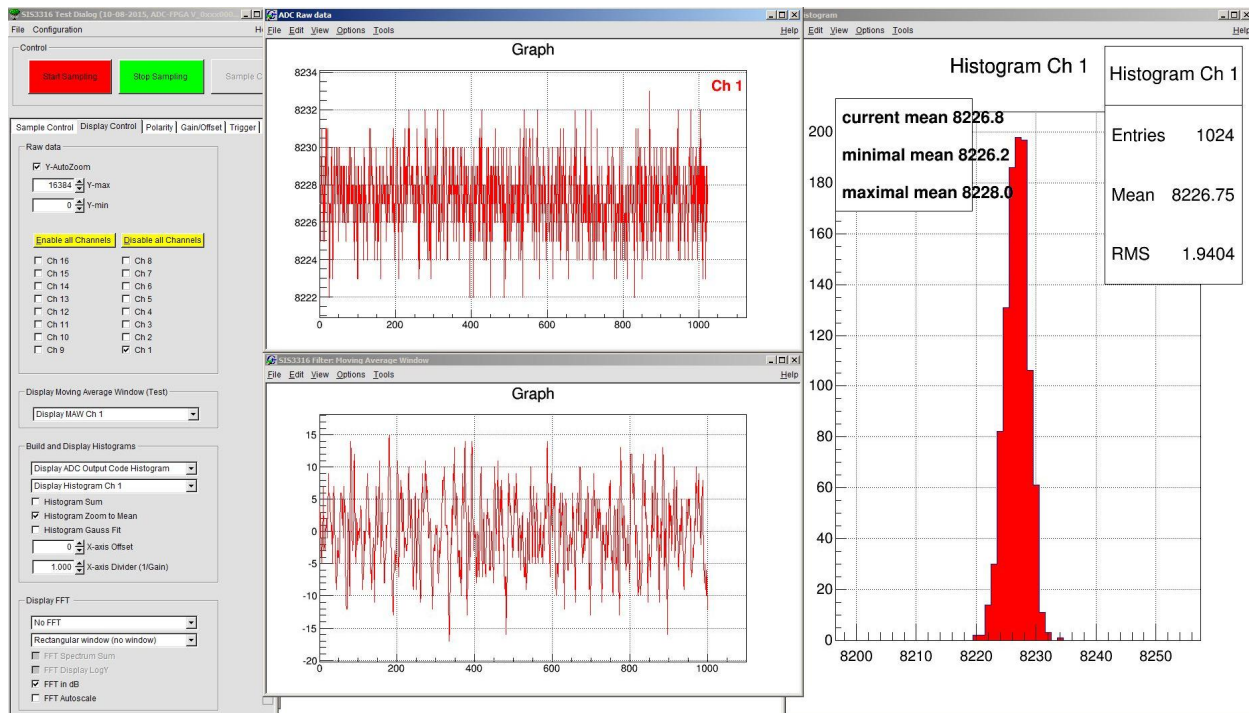
The “SIS3316 Test Dialog” canvas will appear with “Sample Control” tab frame. Click on “Start” button (marked in red). The SIS3316 will start sampling and displaying the signals of all channels in the “ADC Raw data” canvas, the Trigger MAW trapezoidal of channel 1 in the “SIS3316 Filter: Moving Average Window” canvas and the ADC output code of channel 1 in the “Histogram” canvas.



A sine signal is connected to channel 1.



No signal is connected (open) to channel 1.



Click on “Stop” button to stop sampling.

## 3.2.1 Tab Frames

### 3.2.1.1 Sample Control

Sample Control | Display Control | Polarity | Gain/Offset | Trigger | Energy | Sample Clock | NIM Outputs |

Run Control

- ☒ Stop after Bank Loops

Write Data to File

- ☒ Write Data to File enable ☐ Write multiple Files
- 

Sample Parameter

- 
- ☒ use max number of Events
- max Number of Events for each Bank
- Info: Number of Events for each Bank

Event/Hit Parameter

- Pre Trigger Delay
- Raw Data Sample Start Index
- Raw Data Sample Length
- 
- MAW Test Buffer Pre Trigger Delay
- MAW Test Start Index
- MAW Test Buffer Length
- Info: Event Length (32-bit words)
- Info: Active Trigger Gate Length
- ☒ DataFormat: Peak-Height / 6xAccumulators enable
- ☐ DataFormat: 3xTrigger MAW values enable
- ☒ DataFormat: Energy FIR Filter values enable

Sample Trigger Condition

- ☒ Software: Key-Trigger enable
- ☐ VME FPGA: Lemo Input TI use as Trigger
- ☒ ADC FPGAs: External Trigger
- ☐ ADC FPGAs: Internal Trigger
- ☐ ADC FPGAs: Internal Sum Trigger
- ☐ ADC FPGAs: Internal Pileup Trigger

### 3.2.1.2 Display Control

Sample Control | Display Control | Polarity | Gain/Offset | Trigger | Energy | Sample Clock | NIM Outputs

Raw data

☐ Y-AutoZoom

16384 Y-max

0 Y-min

**Enable all Channels** **Disable all Channels**

<input checked="" type="checkbox"/> Ch 16	<input checked="" type="checkbox"/> Ch 8
<input checked="" type="checkbox"/> Ch 15	<input checked="" type="checkbox"/> Ch 7
<input checked="" type="checkbox"/> Ch 14	<input checked="" type="checkbox"/> Ch 6
<input checked="" type="checkbox"/> Ch 13	<input checked="" type="checkbox"/> Ch 5
<input checked="" type="checkbox"/> Ch 12	<input checked="" type="checkbox"/> Ch 4
<input checked="" type="checkbox"/> Ch 11	<input checked="" type="checkbox"/> Ch 3
<input checked="" type="checkbox"/> Ch 10	<input checked="" type="checkbox"/> Ch 2
<input checked="" type="checkbox"/> Ch 9	<input checked="" type="checkbox"/> Ch 1

Display Moving Average Window (Test)

Display MAW Ch 1

Build and Display Histograms

Display ADC Output Code Histogram

Display Histogram Ch 1

☐ Histogram Sum

☐ Histogram Zoom to Mean

☐ Histogram Gauss Fit

0 X-axis Offset

1.000 X-axis Divider (1/Gain)

Display FFT

No FFT

Rectangular window (no window)

☐ FFT Spectrum Sum

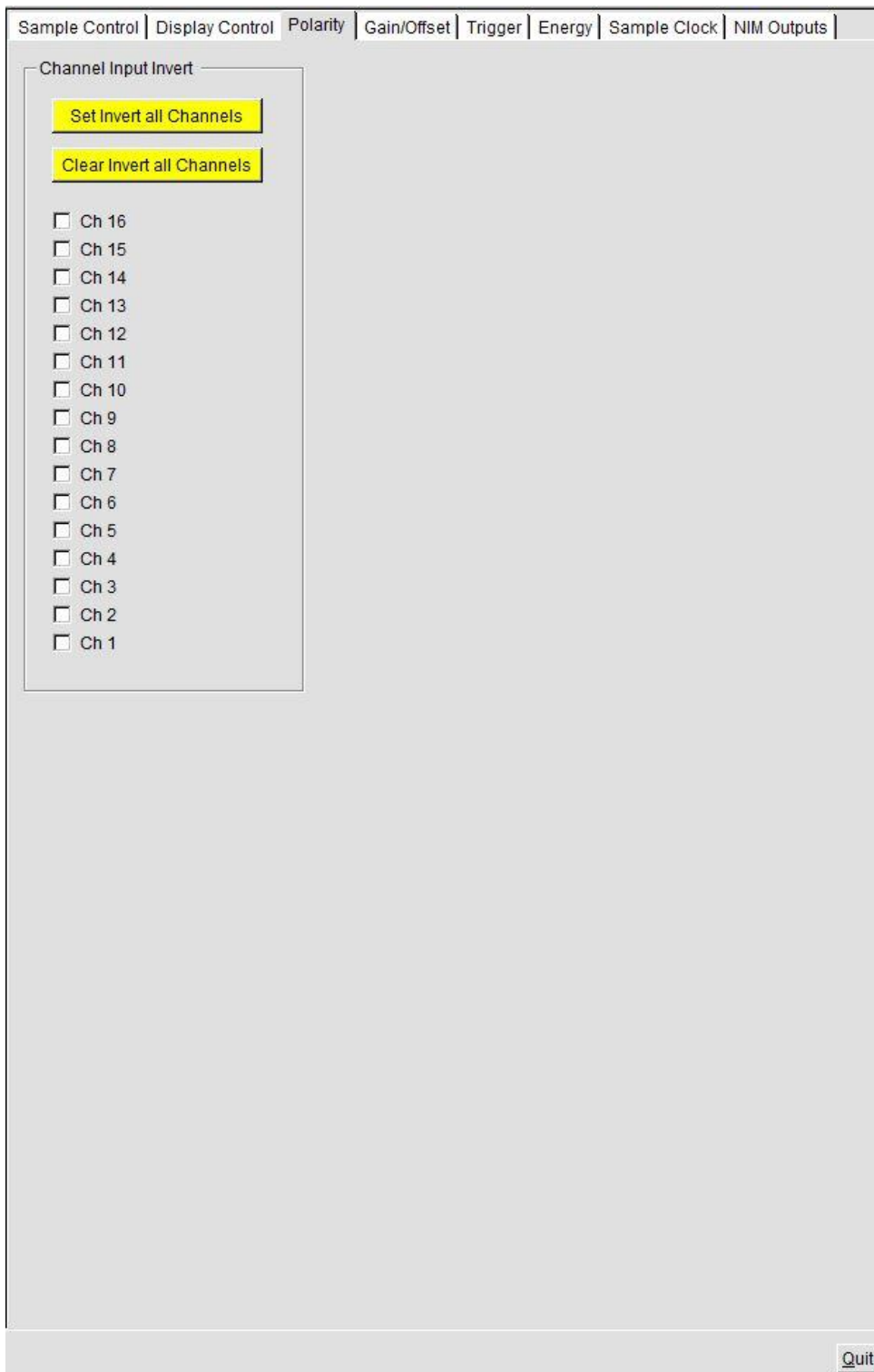
☐ FFT Display LogY

☒ FFT in dB

☐ FFT Autoscale

Quit

### 3.2.1.3 Polarity



The screenshot shows the 'Polarity' tab of the 'sis3316\_root\_gui' software. The interface has a menu bar at the top with the following tabs: 'Sample Control', 'Display Control', 'Polarity' (selected), 'Gain/Offset', 'Trigger', 'Energy', 'Sample Clock', and 'NIM Outputs'. Below the menu bar, on the left side, is a panel titled 'Channel Input Invert'. This panel contains two yellow buttons: 'Set Invert all Channels' and 'Clear Invert all Channels'. Below these buttons is a list of 16 channels, each with a checkbox and a label: 'Ch 16', 'Ch 15', 'Ch 14', 'Ch 13', 'Ch 12', 'Ch 11', 'Ch 10', 'Ch 9', 'Ch 8', 'Ch 7', 'Ch 6', 'Ch 5', 'Ch 4', 'Ch 3', 'Ch 2', and 'Ch 1'. All checkboxes are currently unchecked. The main area of the window is a large, empty gray rectangle. In the bottom right corner of the window, there is a 'Quit' button.



### 3.2.1.4 Gain/Offset

Sample Control	Display Control	Polarity	Gain/Offset	Trigger	Energy	Sample Clock	NIM Outputs
<div><div><div>Channel Input 50 Ohm Termination</div><div><div>Set 50 Ohm Termination all Channels</div><div>Clear 50 Ohm Termination all Channels</div></div><div><div><input checked="" type="checkbox"/> Ch 16<input checked="" type="checkbox"/> Ch 8 <input checked="" type="checkbox"/> Ch 15<input checked="" type="checkbox"/> Ch 7 <input checked="" type="checkbox"/> Ch 14<input checked="" type="checkbox"/> Ch 6 <input checked="" type="checkbox"/> Ch 13<input checked="" type="checkbox"/> Ch 5 <input checked="" type="checkbox"/> Ch 12<input checked="" type="checkbox"/> Ch 4 <input checked="" type="checkbox"/> Ch 11<input checked="" type="checkbox"/> Ch 3 <input checked="" type="checkbox"/> Ch 10<input checked="" type="checkbox"/> Ch 2 <input checked="" type="checkbox"/> Ch 9<input checked="" type="checkbox"/> Ch 1</div><div>checked 50 Ohm else 1K Ohm</div></div></div><div><div>Channel Input Range</div><div><div>Set Input Range 0 all Channels</div><div>Clear Input Range 0 all Channels</div></div><div><div><input checked="" type="checkbox"/> Ch 16<input checked="" type="checkbox"/> Ch 8 <input checked="" type="checkbox"/> Ch 15<input checked="" type="checkbox"/> Ch 7 <input checked="" type="checkbox"/> Ch 14<input checked="" type="checkbox"/> Ch 6 <input checked="" type="checkbox"/> Ch 13<input checked="" type="checkbox"/> Ch 5 <input checked="" type="checkbox"/> Ch 12<input checked="" type="checkbox"/> Ch 4 <input checked="" type="checkbox"/> Ch 11<input checked="" type="checkbox"/> Ch 3 <input checked="" type="checkbox"/> Ch 10<input checked="" type="checkbox"/> Ch 2 <input checked="" type="checkbox"/> Ch 9<input checked="" type="checkbox"/> Ch 1</div><div>checked 5V else 2V</div></div></div></div> <div><div>Channel DAC Offset</div><div><div><div><div>32768</div><div>▲▼</div></div>Ch 16</div><div><div>32768</div><div>▲▼</div></div>Ch 15</div><div><div>32768</div><div>▲▼</div></div>Ch 14</div> <div><div>32768</div><div>▲▼</div></div> Ch 13							

32768

▲▼

32768

▲▼

32768

▲▼

32768

▲▼

32768

▲▼

Ch 8

32768

▲▼

32768

▲▼

32768

▲▼

32768

▲▼

32768

▲▼

32768

▲▼

32768

▲▼

☐ automatical Increment Dac value Test enable

ADC SPI Settings for all channels

ADC chip full scale 2.00V input range

▼

SIS3316 Tap Delay Settings for all channels

1002

▲▼

Tap Delay (hex)

Quit

don't change it (test feature)

## 3.2.1.5 Trigger

Sample Control | Display Control | Polarity | Gain/Offset | **Trigger** | Energy | Sample Clock | NIM Outputs |

FIR Filter Trigger Settings for all channels

4 Trigger Out Pulse Length

8 Trigger Gap

4 Trigger Peaking

100 Trigger Threshold (adc value !)

0 HE-Trigger Threshold (adc value !)

☐ High Energy Suppress Trigger mode

Select -CFD function enable with 50% crossing-

Trigger Enable

Enable all Channels

Disable all Channels

☐ Ch 16 ☐ Ch 8

☐ Ch 15 ☐ Ch 7

☐ Ch 14 ☐ Ch 6

☐ Ch 13 ☐ Ch 5

☐ Ch 12 ☐ Ch 4

☐ Ch 11 ☐ Ch 3

☐ Ch 10 ☐ Ch 2

☐ Ch 9 ☐ Ch 1

☐ Ch\_Sum 1 to 4

☐ Ch\_Sum 5 to 8

☐ Ch\_Sum 9 to 12

☐ Ch\_Sum 13 to 16

Pileup Settings for all channels

0 Pileup Length

0 Re-Pileup Length

Internal trigger and High-Energy trigger routing selection to the VME FPGA

Select -Internal Trigger- as -Internal Trigger- to VME FPGA

Select -Internal HE-Trigger- as -Internal HE-Trigger- to VME FPGA

Quit

Trigger Threshold (adc\_value !):

What does “adc values!” mean in this context ?

The FIR filters (Trigger and Energy) are building the complete sum of the adc data words over the Peaking Time. No “averaging” are performed! The denotation of **MAW** is caused by history and it is to equate with **MW** in this document!

Therefore the Trigger Threshold depends on the Trigger Peaking value, also. The software multiplies the defined value in the “Trigger Threshold” entry box with the value of the “Trigger Peaking” entry box.

The result will be written to Trigger Threshold registers.

**Note:** the “sis3316\_root\_gui” program does not support parameters like “Trigger Threshold” for the individual channel, while the SIS3316 firmware does support individual parameters for each channel!



### 3.2.1.6 Energy

Sample Control | Display Control | Polarity | Gain/Offset | Trigger | **Energy** | Sample Clock | NIM Outputs

FIR Filter Energy Settings for all channels

Energy Peaking

Energy Gap

Energy Decay Tau table

Energy Decay Tau factor

Energy Additional Average factor

Energy Pickup Index

Accumulator Settings for all channels

Start Index of Accumulator 1

Length of Accumulator 1

Start Index of Accumulator 2

Length of Accumulator 2

Start Index of Accumulator 3

Length of Accumulator 3

Start Index of Accumulator 4

Length of Accumulator 4

Start Index of Accumulator 5

Length of Accumulator 5

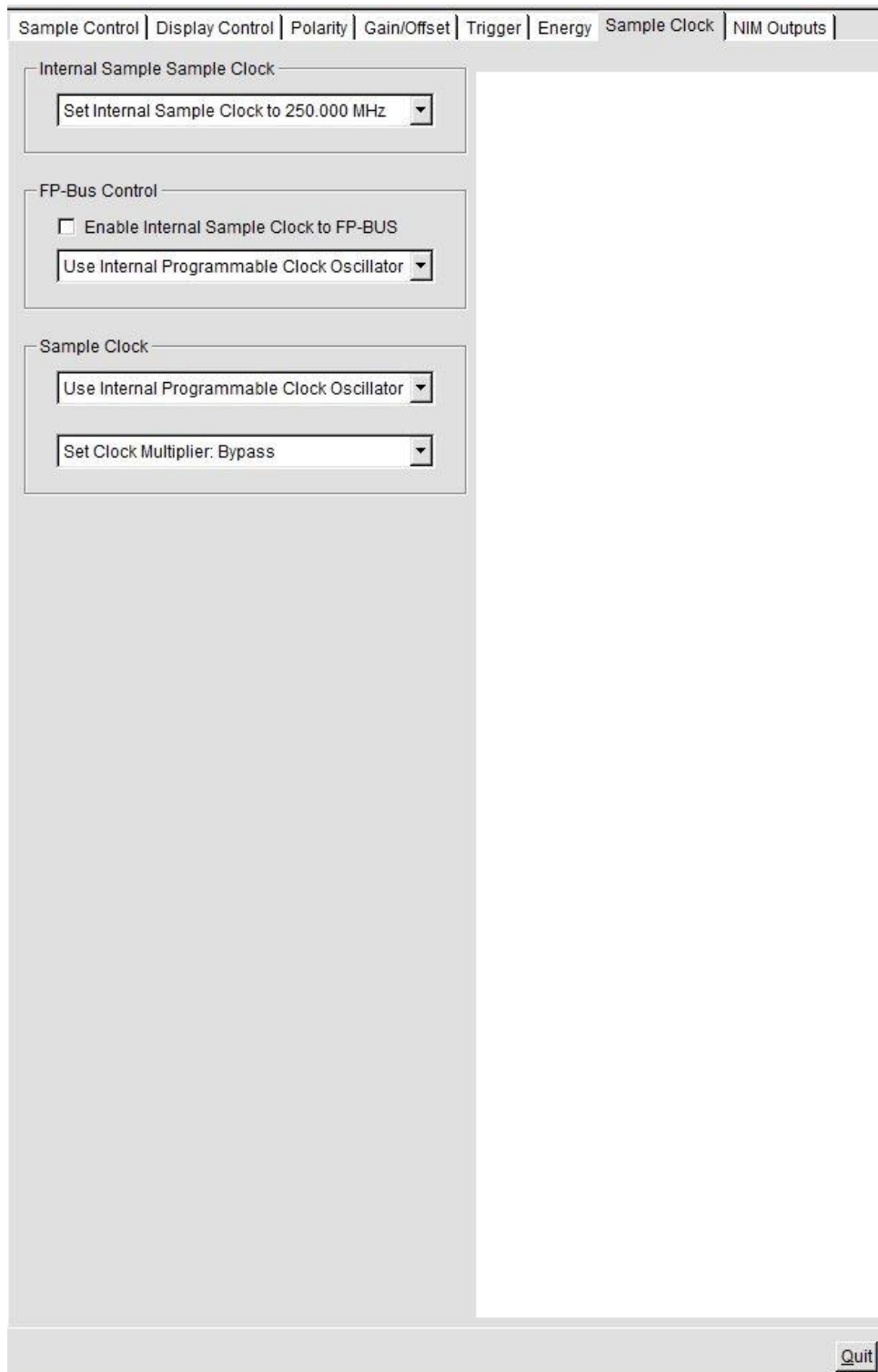
Start Index of Accumulator 6

Length of Accumulator 6

Quit

**Note:** the “sis3316\_root\_gui” program does not support individual parameters like “Energy Peaking” for the individual channel, while the SIS3316 firmware does support individual parameters for each channel!

### 3.2.1.7 Sample Clock

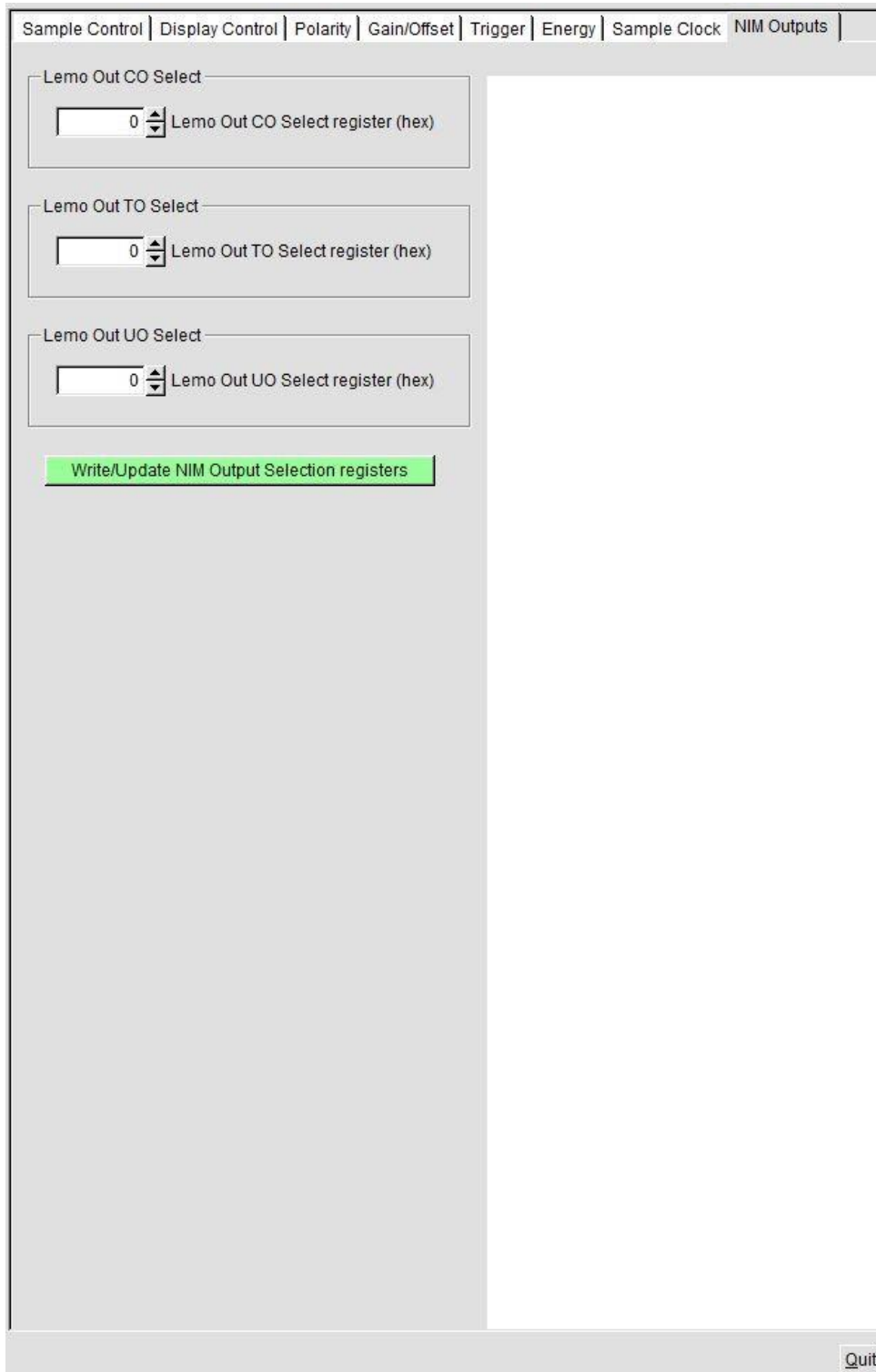


The screenshot shows the 'Sample Clock' configuration window of the sis3316\_root\_gui. The window has a title bar and a menu bar with the following items: Sample Control, Display Control, Polarity, Gain/Offset, Trigger, Energy, Sample Clock (highlighted), and NIM Outputs. The main content area is divided into three sections:

- Internal Sample Clock:** Contains a dropdown menu with the text 'Set Internal Sample Clock to 250.000 MHz'.
- FP-Bus Control:** Contains a checkbox labeled 'Enable Internal Sample Clock to FP-BUS' (which is unchecked) and a dropdown menu with the text 'Use Internal Programmable Clock Oscillator'.
- Sample Clock:** Contains two dropdown menus. The first has the text 'Use Internal Programmable Clock Oscillator'. The second has the text 'Set Clock Multiplier: Bypass'.

A 'Quit' button is located in the bottom right corner of the window.

### 3.2.1.8 NIM Outputs



The screenshot shows the 'NIM Outputs' tab of a software interface. The tab bar at the top includes 'Sample Control', 'Display Control', 'Polarity', 'Gain/Offset', 'Trigger', 'Energy', 'Sample Clock', and 'NIM Outputs'. The main area contains three sections for selecting output registers:

- Lemo Out CO Select**: A numeric input field with '0' and a spin button, followed by the text 'Lemo Out CO Select register (hex)'.
- Lemo Out TO Select**: A numeric input field with '0' and a spin button, followed by the text 'Lemo Out TO Select register (hex)'.
- Lemo Out UO Select**: A numeric input field with '0' and a spin button, followed by the text 'Lemo Out UO Select register (hex)'.

Below these sections is a green button labeled 'Write/Update NIM Output Selection registers'. At the bottom right of the window is a 'Quit' button.

### 3.2.2 Save/Load Configuration parameters

Most of the “sis3316\_root\_gui” parameters can be saved to a configuration file and can be loaded from a configuration file.

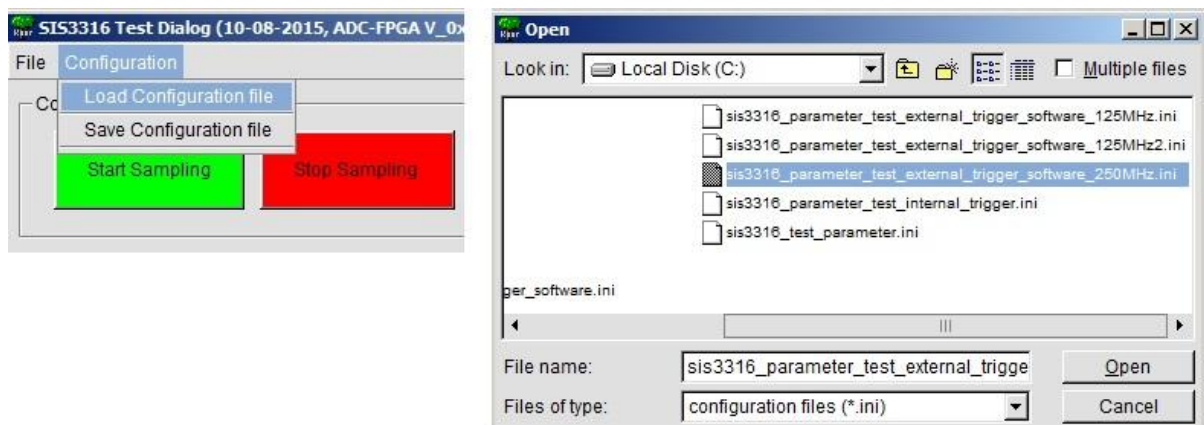
The parameter definition can be found in the header file “`get_configuration_parameters.h`”.

For example:

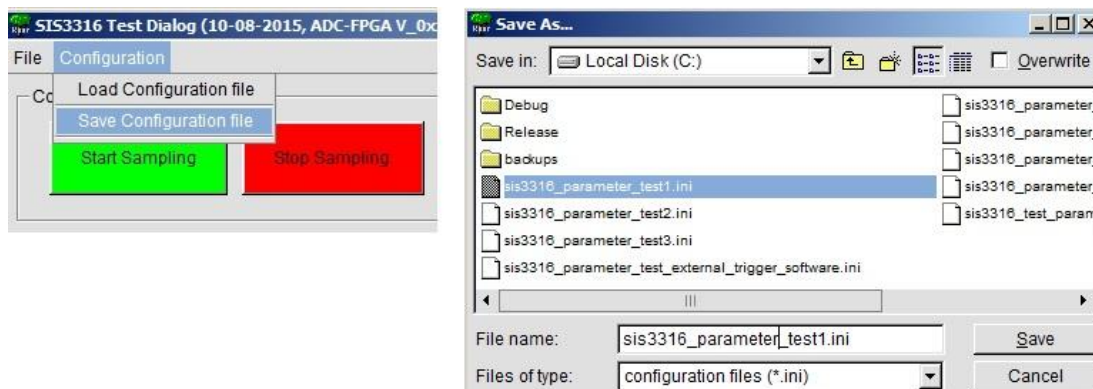
```
#define TEXT_PARAMETER_TRIGGER_COND_SOFTWARE_KEY "PARAMETER_TRIGGER_COND_SOFTWARE_KEY"
#define TEXT_PARAMETER_TRIGGER_COND_VME_LEMO_TI "PARAMETER_TRIGGER_COND_VME_LEMO_TI"
#define TEXT_PARAMETER_TRIGGER_COND_ADC_EXTERNAL "PARAMETER_TRIGGER_COND_ADC_EXTERNAL"
#define TEXT_PARAMETER_TRIGGER_COND_ADC_INTERNAL "PARAMETER_TRIGGER_COND_ADC_INTERNAL"
#define TEXT_PARAMETER_TRIGGER_COND_ADC_INTERNAL_SUM "PARAMETER_TRIGGER_COND_ADC_INTERNAL_SUM"
#define TEXT_PARAMETER_TRIGGER_COND_ADC_INTERNAL_PILEUP "PARAMETER_TRIGGER_COND_ADC_INTERNAL_PILEUP"

#define TEXT_PARAMETER_CHANNEL_POLARITY_INVERT "PARAMETER_CHANNEL_POLARITY_INVERT"
#define TEXT_PARAMETER_CHANNEL_RANGE_2V "PARAMETER_CHANNEL_RANGE_2V"
#define TEXT_PARAMETER_CHANNEL_50OHM_TERMINATION_DISABLE "PARAMETER_CHANNEL_50OHM_TERMINATION_DISABLE"
#define TEXT_PARAMETER_CHANNEL_ADC_OFFSET "PARAMETER_CHANNEL_ADC_OFFSET"
```

Select “Configuration” in the menu bar and select “Load Configuration file” to get a browser window to search for and select a configuration file (\*.ini).



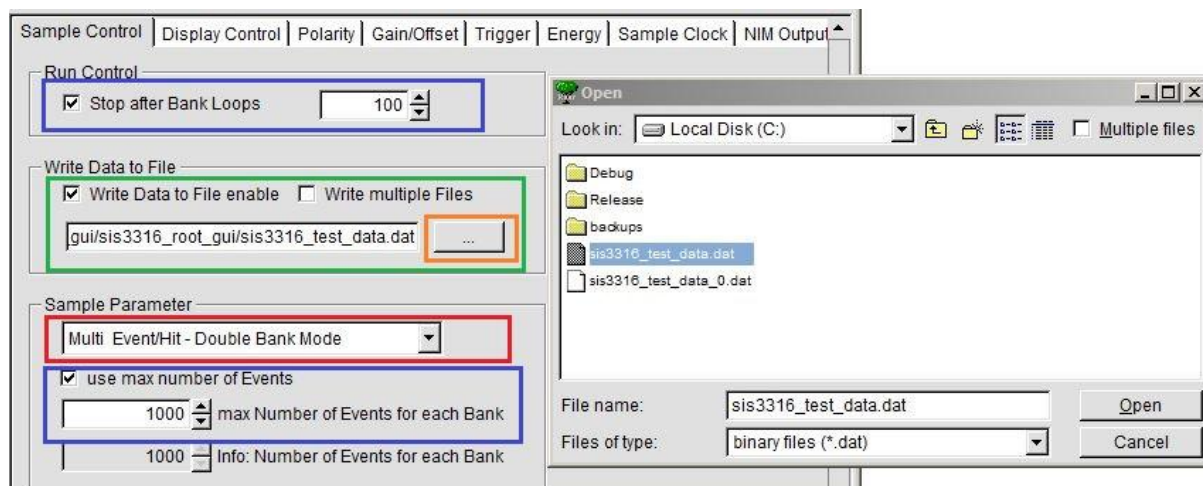
Select “Configuration” in the menu bar and select “Save Configuration file” to get a browser window to search for and select or define a configuration file name (\*.ini).



### 3.2.3 Write Data File Format

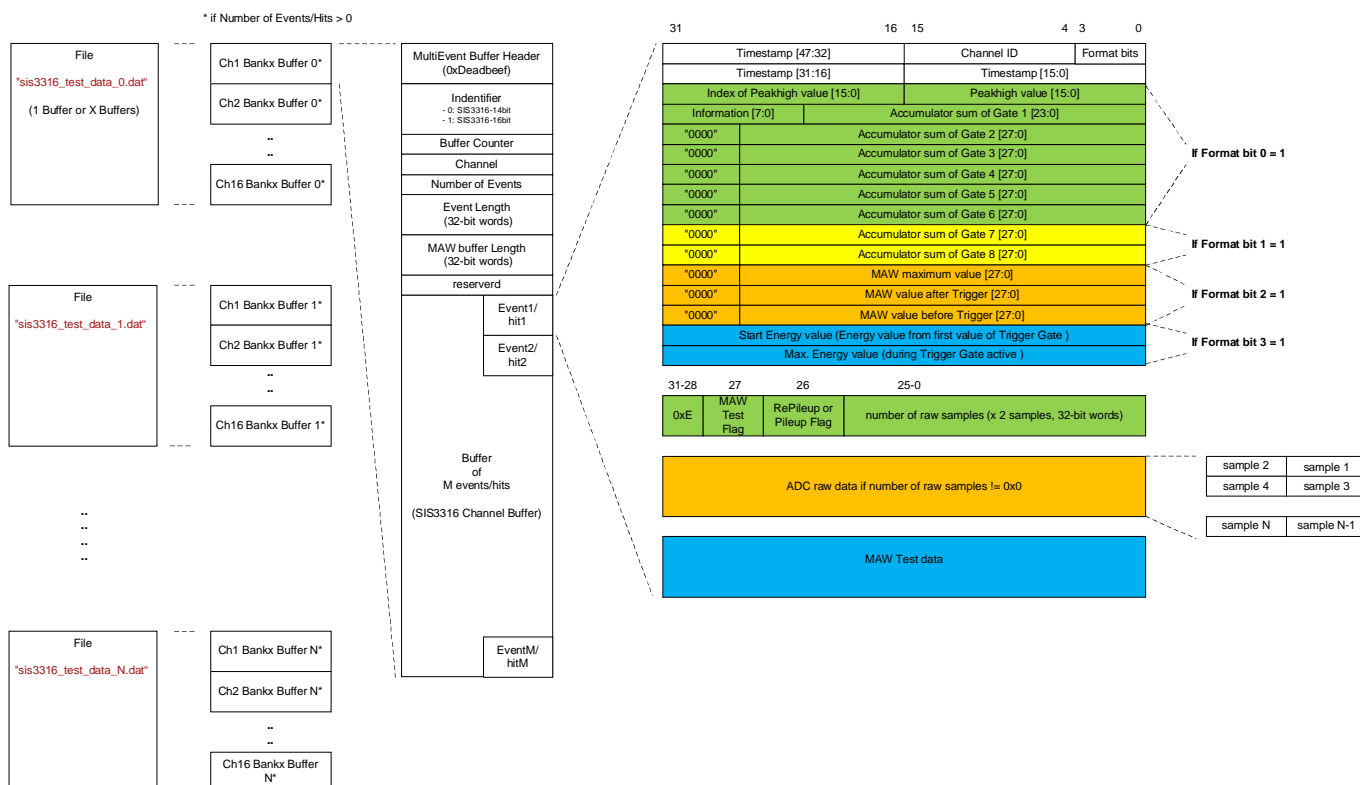
It is possible to write the sampled data to a file with the “Multi Event/Hit – Double Bank Mode” (marked in red). The check button “Write Data to File enable” has to be set (marked in green). Select “...” button (marked in orange) to get a browser window to define or select a file name.

The program will save 100 “banks” with each 1000 Hits/Events (marked in blue).



Depending on the check button “Write multiple Files” the 100 bank buffers will be saved to one or to 100 files.

#### File Data Format:



## 4 Index

ADC FPGA 20  
base address 17  
build  
    Eclipse 15  
    shell 16  
Build the program sis3316\_root\_gui 6  
CERN 4  
Communication Interface 5  
Configuration parameters 33  
data format 34  
DHCP 18  
Eclipse 15  
    build 15  
**Eclipse 3.6.1** 13  
Ethernet  
    Interface 18  
FFT 4  
File Format 34  
FIR 4  
FPGA  
    ADC 20  
    VME 19  
FPGA Programming 19  
FPGA reboot 21  
Graphical Interface 4  
GUI 4  
image  
    VME FPGA 19  
interface  
    Ethernet 18  
    VME 17  
introduction 4  
IP address 18  
JTAG 21  
Linux 13  
Name-Server 18  
NIM 32  
Operating System 4  
outputs  
    NIM 32  
ROOT 4  
Run the program sis3316\_root\_gui 17  
SBC 5  
Scientific Linux 4  
shell  
    build 16  
    run 16  
Test Menu 22  
VC10 Build project 11  
VC10 Include path settings 8  
VC10 Library path settings 10  
VC10 Project Properties 8  
VC10 Run project 12  
verification 20, 21  
VME FPGA 19  
VME interface 17  
Windows 4  
Windows 7 / Windows 8 5, 6