PDA16 Scope Application Operator's Manual

DynamicSignals LLC
900 North State Street
Lockport, Illinois 60441-2200
USA

Tel: (815) 838-0005
Fax: (815) 838-4424
Web: http://www.signatec.com
Web: http://dynamicsignals.com

Copyright © 2012

All Rights Reserved

Revision 1.3 – 04/04/2012

TABLE OF CONTENTS

1 OVERVIEW 1
1.1 Requirements 1
1.2 Running the PDA16 Scope Application 1
1.3 Virtual PDA16 Devices 2
1.4 Remote PDA16 Devices 2
1.4.1 Setting up the PDA16 Server 2
1.4.2 Setting Up the PDA16 Scope Application as a Client 3
1.5 Startup Files 3
2 PDA16 SCOPE APPLICATION USER INTERFACE 4
2.1 Overview 4
2.2 The Control Panel 5
2.2.1 Acquire Data 5
2.2.2 Acquire Continuously 6
2.2.3 Record Data 6
2.2.4 Enter Standby Mode 6
2.2.5 Issue Software Trigger 6
2.3 The Settings Panel 6
2.3.1 The Data Tab 7
2.3.2 The Sampling Rate Tab 9
2.3.3 The Trigger Tab 10
2.3.4 The Status Tab 12
2.3.5 The Timestamps Tab 14
2.3.6 The FPGA Processing Tab 15
2.3.7 The SAB Tab 16
2.3.8 The Data Plotting Tab 17
2.4 The Scope Panel 17
2.4.1 Mouse Controls 18
2.4.2 Keyboard Controls 18
2.5 PDA16 Scope Application Main Menu Options 19
2.5.1 The File Menu 19
2.5.2 The View Menu 20
2.5.3 The Hardware Menu 22
2.5.4 The Tools Menu 24
2.6 Recording PDA16 Data 24
2.6.1 Device and Destination File Selection 25
2.6.2 Recording Type 26
2.6.3 Recording Duration 26
2.6.4 Recording Options 27
2.6.5 Running a Recording 27
2.7 Destination File Advanced Options 27
2.7.1 Append data if it already exists 27
2.7.2 Deinterleave this data into separate files 28
2.7.3 Span data over multiple, statically sized files 28
2.7.4 Leave room for application-specific data 28
2.7.5 Save data as text 28

2.7.6 Don't try to use fast, unbuffered IO for binary files 28
2.7.7 Convert data to signed format before writing 29
2.7.8 Save PDA16 timestamps to external file (*.pda16ts) 29
2.7.9 Signatec Recorded Data Context (SRDC) Information 29
2.8 FFT Analysis of PDA16 Data 30
2.8.1 Frequency and Time Domain Statistics 31
2.8.2 Frequency Domain Plots 31
2.8.3 FFT Options Dialog 32
3 APPENDIX A – REVISION HISTORY 1

1 OVERVIEW

The PDA16 Scope Application ('Scope App') is an application that is installed as part of the standard PDA16 software installation on Microsoft Windows platforms. This application is used to control one or more Signatec PDA16 data acquisition PCI/PCI-X expansion boards. The PDA16 is a dual channel, 16-bit data acquisition card with acquisition rates up to at least 160MHz.

The PDA16 Scope Application is a virtual oscilloscope application that allows the operator to view or edit all PDA16 hardware settings as well as record and display PDA16 acquisition data. For many users, the Scope App is the primary software tool used to record PDA16 acquisition data.

The goal of this document is to provide information for operational use of the PDA16 Scope Application. This manual is not a PDA16 hardware reference manual. For more specific details on the PDA16 hardware and operation consult the PDA16 Operator's Manual. The PDA16 Operator's Manual is installed in the "Documentation" folder of the PDA16 installation folder (C:\Program Files\Signatec\PDA16 by default). There may also be a link in your Start Menu under All Programs → Signatec → PDA16.

1.1 Requirements

Refer to the PDA16 Operator's Manual for hardware and system requirements for running PDA16 hardware.

The PDA16 Scope Application is dependent on core libraries and components installed by the PDA16 software setup.

1.2 Running the PDA16 Scope Application

There are several ways to start the PDA16 Scope Application:

- Via the shortcut installed on the Desktop.
- Via the Start Menu (Start → All Programs → Signatec → PDA16 → PDA16 Scope Application.
- Directly: C:\Program Files\Signatec\PDA16\PDA16Scope.exe

When the Scope Application starts, it will automatically connect to all local PDA16 devices. Note that when the application starts up the PDA16 hardware is not accessed. This allows the Scope Application to run without directly affecting any PDA16 operation that may be in progress with other software.

If no physical PDA16 devices are detected in the system[1], then a single 'virtual PDA16' device will be used. A virtual device is an imaginary device that is not connected to any real hardware. It's implemented mainly as a software debugging tool, but also serves to show what the software can do when the hardware is present.

The Scope Application uses an MDI interface so each managed PDA16 device will have its own window.

The PDA16 Software Installation associates a few different file types with the PDA16 Scope Application. When activated from Windows Explorer, the PDA16 Scope Application will automatically be loaded with the specified file as a startup file. The Startup Files section describes these file types in more detail.

[1] If you have PDA16 devices in your system but none are detected by the Scope Application then it's likely that the PDA16 device driver is missing or has not been installed. Consult the PDA16 Operator's Manual for details on how to install the PDA16 driver.

1.3 Virtual PDA16 Devices

When the Scope Application starts up, if no physical PDA16 devices are detected, then the application will create a single ‘virtual’ device. A virtual device is an imaginary device that is not connected to any real hardware. Most application operations may be performed, but since there’s no real hardware being controlled it isn’t very exciting.

The virtual device mechanization is mainly used as a software debugging tool for application development. It can be used to help differentiate between software and hardware/firmware problems.

When a virtual device is used the board’s name will be prefaced with “Virtual”. If you have a PDA16 board plugged into your system, yet the PDA16 Scope Application indicates virtual hardware in use; this would indicate that your computer system has not detected the PDA16 hardware.

1.4 Remote PDA16 Devices

The PDA16 Scope Application has the ability to connect to PDA16 devices that reside on remote, non-local systems. If you plan to run the PDA16 Scope Application on the same computer that the board is installed in; you do not need to follow any of the steps in sections 1.4x. However, if you wish to use the PDA16 Scope Application on a remote computer, then the information in sections 1.4x are important to follow.

The PDA16 Scope Application software uses local TCP/IP services to connect to the remote PDA16 server. With the exception of a few small user-interface differences, controlling remote PDA16 devices isn’t any different from controlling normal, local PDA16 devices.

In order to work with a remote PDA16 device two pieces of software must be used: the client and the server. The client software is the PDA16 Scope Application. The server software is the Signatec Service Manager, which must be installed, along with the normal PDA16 software, on the server system (where the PDA16 hardware resides).

1.4.1 Setting up the PDA16 Server

In order to run a PDA16 server, the Signatec Service Manager (“Service Manager”) software must be installed. At the time of this writing, this software is not part of the standard PDA16 software installation. It should be included in the ‘SigSvcMgr’ directory of the installation media. This software should also be available on the Signatec website.

The server software will need to be installed on the remote machine that will host the PDA16 board(s). Also note that the standard PDA16 installation software must also be installed on the remote machine.

The Service Manager is designed to host arbitrary devices/services. As such, it implements service requests by deferring the calls to respective service-specific libraries. Before the Service Manager can host a PDA16 service, it must be properly connected to the PDA16-specific library. The Service Manager obtains this information from a small XML file that resides in the application installation directory (C:\Program Files\Signatec\SigServerMgr by default). This configuration file is always named SigSvcCfg.xml. Here’s an example configuration file that defines a PDA16 service:

```
<?xml version="1.0" encoding="utf-8"?>
<SigServiceCfg>
  <Service Index="0"
   Port="0">C:\Program Files\Signatec\PDA16\PDA16.dll</Service>
</SigServiceCfg>
```

If the PDA16 service is not already installed on your server system you will need to add the bold section to your own SigSvcCfg.xml file. Note that the path above is the path used for default installations; if you've installed to a non-default folder you may need to alter the path accordingly.

Once the configuration data has been specified, services can begin by starting the Service Manager and selecting "Startup Services" on the main form. Service Manager has options to automatically start when the system starts up.

The Service Manager will report if there was a problem starting up any services. Also, all currently running services are displayed on the main form.

1.4.2 Setting Up the PDA16 Scope Application as a Client

Assuming that the Signatec Service software is properly configured and running; connecting to it is simple:

- In the File menu, select 'Connect to Remote PDA16…'
- On the Connect to Remote PDA16 Device dialog, input the IP address of your PDA16 server. (Tip: The Service Manager software displays its IP address on the main window.)
- Use the default service port, 3490
- Check the 'Connect to all devices on server' to automatically connect to all PDA16 devices on the server.
- Select OK to connect.

You can use the Test Server button on the dialog to see if any PDA16 devices are running on the server identified by the current parameters. If any devices are on that server, they will be added to the device combo box.

Once you've connected to the remote PDA16 you can use it just like you would any local PDA16 device. In fact, as far as the Scope Application is concerned, it doesn't really know if it's connected to a local or a remote PDA16 device. The small, but important, exception to this is that all paths for data files for use with a remote PDA16 are assumed to be server-specific paths. This means that if you want to record acquisition data to file "C:\data\MyRecording.rd16" on a remote PDA16, that path must be a valid path on the *server* machine.

File browsing dialogs for remote PDA16 devices will browse on the server, not the client.

1.5 Startup Files

The PDA16 Scope Application understands various file types that are used when operating PDA16 devices. When passed on the command line, these files will automatically be opened up and applied to any or all PDA16 devices. When one of these known files is specified on the command line, this is referred to as a *startup file*.

During the PDA16 Software Installation, these known file types are associated with the PDA16 Scope Application such that when they are activated (e.g. double-clicked) the PDA16 Scope Application will be opened up with that file. This has the same effect as starting the PDA16 Scope Application with the startup file on the command line.

During PDA16 Scope Application initialization, after all PDA16 devices have been connected to, the application will check to see if a startup file was specified. If so, and there's only a single PDA16 device, it will apply to that device. If more than one PDA16 device is present, a user-interface will be provided to select which PDA16 device(s) the startup file should apply to.

The currently associated file types are listed in the following table:

File Type	Extension	Notes
PDA16 Hardware Settings Files	.pda16set	Files of this type contain previously saved hardware settings. Since loading these settings will affect the hardware, the user will be prompted if they want to continue before the hardware settings are applied.
PDA16 Firmware Update Files	.pda16fw	Files of this type contain PDA16 firmware updates. The firmware update is not immediate; users will have the ability to cancel out.
PDA16 Recorded Data Files	.rd16	Files of this type contain previously recorded data. When loaded into the Scope Application, they are added as a data source in the Scope Panel.

2 PDA16 SCOPE APPLICATION USER INTERFACE

2.1 Overview

The Scope Application may be used to modify any of the various hardware settings that affect how the PDA16 behaves. These settings are distributed over the tabbed view at the bottom of the main device form. These settings are described in the subsequent sections.

For all hardware settings, whenever an item in the user-interface is selected, the hardware is immediately updated with the new setting. When editing a user interface item with an edit control (e.g. the segment size parameter on the Trigger tab) the operator will need to press Enter in order for the modification to have effect. On success, the status bar will indicate the successful update. On error, a message box with details on the error will be displayed.

A screenshot of the application is shown in the figure below. The main interface is divided into three resizable panels. These panels are briefly described here in 2.1, with detailed information provided in subsequent sections.

The top panel is the 'Scope' panel and is used to display PDA16 acquisition data, data recording snapshots, and previously recorded data files. The Scope panel is capable of simultaneously displaying multiple data channels.

The bottom-left panel is the 'Control' panel and is used to start and stop data acquisitions and recordings. The options on this panel have to do with controlling the PDA16 operating mode.
The bottom-right panel is the 'Settings' panel and is a tabbed view with various views that control the various PDA16 hardware settings. Certain tabs are only displayed if the underlying PDA16 device supports those features. Not all boards support FPGA Processing for example, so that tab will only be displayed if the underlying PDA16 supports FPGA Processing.

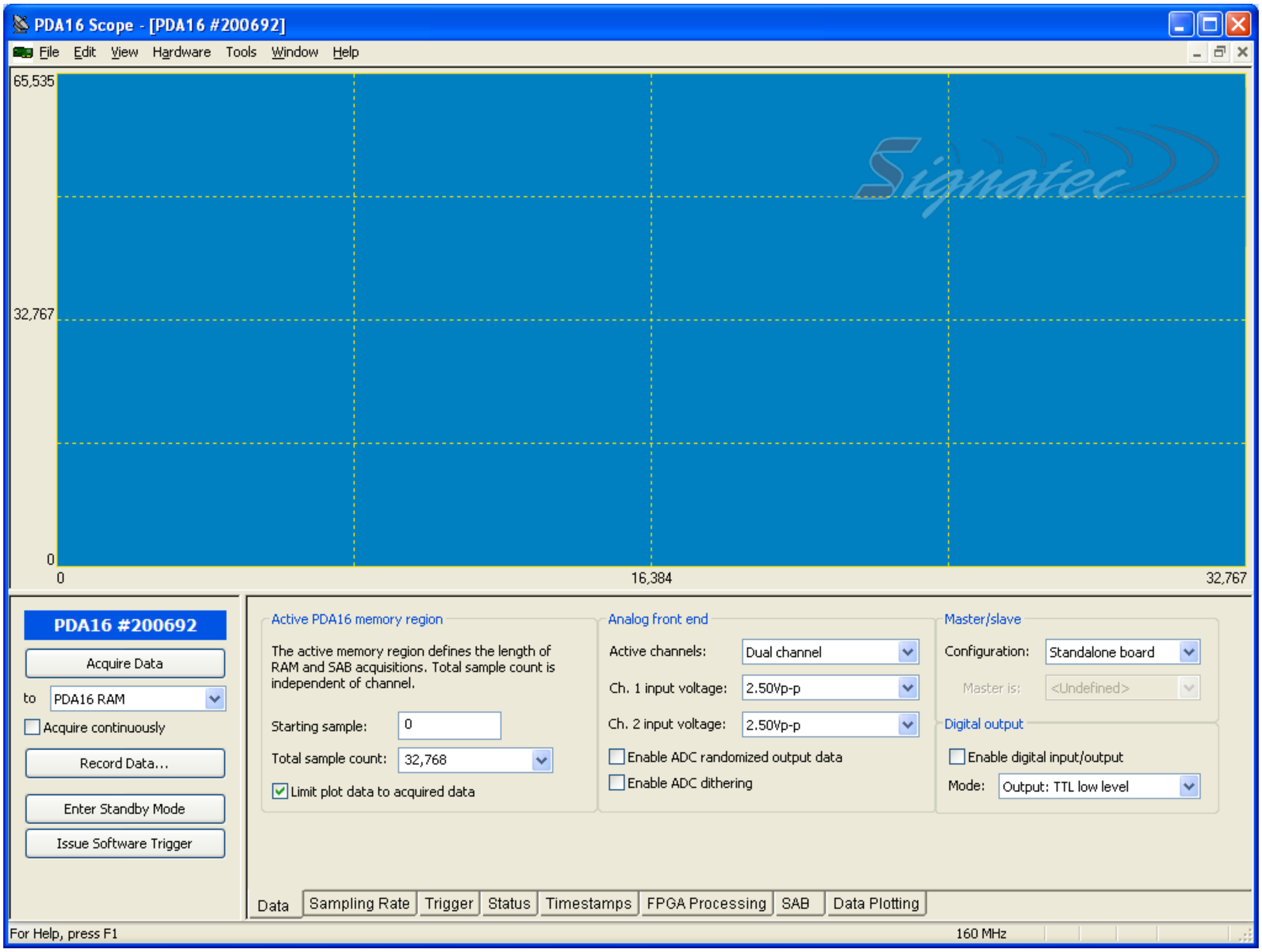

Figure 2-1: PDA16 Scope Application User Interface

2.2 The Control Panel

The Control panel is the bottom-left panel of the main application window. This is the main interface for controlling PDA16 operations like starting/stopping data acquisitions and recordings.

2.2.1 Acquire Data

Clicking this button will do a single data acquisition to the entity specified in the combo box under the button. This will commonly be the PDA16 RAM. (If no combo box is visible then PDA16 RAM will be the target object). The length of the acquisition is defined by the Active Memory region settings on the Data tab of the Settings Area.

When the data acquisition completes, a notification is displayed in the status bar and the plot area is updated to display fresh data. Note that the plot area can display data from the entire PDA16 RAM, which may or may not contain the newly acquired data. Checking the "Limit plot data to acquired data" checkbox on the Data tab will only display newly acquired data and ignore the unused portion of the PDA16 RAM.

If the acquisition does not complete within a short time (a few hundred milliseconds or so) the user interface will update to indicate that the acquisition is pending completion. At this point the Acquire Data button will turn into a ‘Cancel Acquisition’ button. Also, certain hardware settings that cannot be modified while an acquisition is in progress will become disabled.

Hint: If an acquisition does not complete (e.g. because the board is still waiting for a trigger event) you can click the ‘Issue Software Trigger’ button to force the PDA16 to trigger and start the data collection.

Related PDA16 library functions: AcquireToBoardRamD16, AcquireToSabD16, TransferSampleRamToSabD16

2.2.2 Acquire Continuously

If this checkbox is selected when the Acquire Data button is selected, the application will go into a continuous mode where after the data acquisition completes and the data is displayed, another acquisition will begin immediately thereafter. In this mode, the ‘Acquire Data’ button will be renamed to ‘Cancel Continuous Mode’. Pressing this button will stop the continuous data acquisition.

2.2.3 Record Data

Clicking the ‘Record Data…’ button will bring up the Record PDA16 Data window. This window can be used to control data acquisition recordings for one or more PDA16 devices.

See Recording PDA16 Data section for details on this window.

2.2.4 Enter Standby Mode

This button is mainly for troubleshooting purposes. Clicking this button will place the PDA16 device into the Standby operating mode. This will also have the effect of cancelling the current data acquisition or data transfer, even if another thread or process is controlling the card. If the Scope Application (or custom PDA16 software) ever becomes unresponsive during a recording or acquisition (this should not happen with release firmware) you can open up a new instance of the Scope Application and select this button to abort the pending operation that the stuck application is waiting for.

Related PDA16 library functions: SetOperatingModeD16

2.2.5 Issue Software Trigger

Clicking the ‘Issue Software Trigger’ button will issue a software-generated trigger event to the PDA16. This is useful for forcing data collection to begin in situations where a proper trigger is not available.

If a software trigger is issued when segmented trigger mode is being used, the software-generated trigger will trigger all subsequent segments.

Related PDA16 library functions: IssueSoftwareTriggerD16

2.3 The Settings Panel

This panel contains various tabs that contain interfaces for viewing/modifying the various PDA16 hardware settings. Except where explicitly noted, whenever a setting is changed in the user interface it is applied immediately to the underlying hardware. On success, the application status bar will be updated to indicate that the hardware was updated. On error, the operator will be notified via a message box.

2.3.1 The Data Tab

This Data tab contains settings that control how much data to acquire for RAM acquisitions as well as analog front end settings.

2.3.1.1 Starting Sample

The starting sample setting defines the PDA16 RAM sample index at which data will be written for subsequent RAM acquisitions.

This value will be aligned down to a multiple of 4 samples and will be clipped to the largest valid starting sample: 268,435,452.

Dual-channel data is stored in RAM in a channel-interleaved format (Ch 1, Ch 2, Ch 1, Ch 2 …). When dealing with dual channel data, the RAM address of Ch. 1 sample *N* would be 2**N* and the RAM address of Ch. 2 sample *N* would be 2**N* + 1.

Unlike most all other hardware settings, this setting is not applied immediately to the hardware after being edited by the operator. The setting will be applied prior to the next data acquisition operation.

Related PDA16 library functions: SetStartSampleD16

2.3.1.2 Total Sample Count

The Total Sample Count setting defines the length, in samples, of subsequent RAM acquisition operations. This count is independent of channel count, so for a total sample count of *T*:

- A single channel acquisition will result in an acquisition of *T* samples.
- A dual-channel acquisition will result in an acquisition of *T* samples, or *T*/2 samples per channel.

This value should be a multiple of 16. This value will be clipped to the largest valid sample count value: 1,073,741,808. (Note that this is larger than the PDA16 RAM size; larger acquisition sizes can be used with SAB acquisitions.)

When 0 is specified as the total sample count, the PDA16 Scope Application interprets this as a ‘transfer-only’ request. This means that no new data acquisition will take place but the Scope area will be updated to display current PDA16 RAM content.

Related PDA16 library functions: SetSampleCountD16

2.3.1.3 Limit Plot Data to Acquired Data

If this checkbox is selected then the Scope panel will only display the newly acquired data. When unchecked, the Scope panel will allow for panning through the entire PDA16 RAM.

2.3.1.4 Active Channels

This item selects which channels will be digitized by subsequent data acquisitions.

When ‘Dual channel’ is selected the PDA16 will digitize the incoming data on the Channel 1 and Channel 2 input connectors. Data will be captured a channel-interleaved format: Ch. 1, Ch. 2, Ch. 1, Ch. 2…

When ‘Single channel – Ch. 1’ is selected the PDA16 will digitize the incoming data on the Channel 1 input connector.

Related PDA16 library functions: SetActiveChannelsD16

2.3.1.5 Channel Input Voltage

This setting controls the input voltage range for the channel inputs. Each channel can have its own input voltage range.

Related PDA16 library functions: SetVoltRangeCh1D16, SetVoltRangeCh2D16

2.3.1.6 Enable ADC Randomized Output Data

This setting enables/disables ADC randomized output data. When enabled, this setting may reduce amplitude of frequency spurs.

Related PDA16 library functions: SetAdcRandomizedOutputDataEnableD16

2.3.1.7 Enable ADC Dithering

This setting enables/disables ADC dithering. When enabled, this setting may reduce low signal amp frequency spurs.

Related PDA16 library functions: SetAdcDitheringEnableD16

2.3.1.8 Master/Slave Configuration

This setting selects the PDA16's master/slave configuration. This feature is only relevant when multiple PDA16 devices are being used. In a master/slave setup, one master PDA16 device provides the clock and trigger to one or more slave PDA16 devices. This allows all slave devices to acquire synchronously with the master.

The master and all slaves must be physically connected via a master/slave cable.

IMPORTANT: The operator must ensure that there is never more than one master device in any master/slave group. Having more than one master will mean that multiple devices are driving the master/slave bus and this can damage the master devices.

When a board is selected as a slave, the 'Master is' combo box should be set with the slave's master device. Doing this will allow the Scope Application to make sanity checks for certain operations.

Related PDA16 library functions: SetMasterSlaveConfigurationD16

2.3.1.9 Digital Output

These controls affect the behavior of the PDA16 digital IO feature. The PDA16 has a digital IO SMA connector that can be configured to drive out certain signals, or accept certain input signals. The 'Enable digital input/output' check box is used to enable the digital IO port. The actual output or input function is selected with the 'Mode' combo box.

Currently defined digital IO modes are:

- Output: TTL low level – Outputs a TTL low-level signal (0V).
- Output: Synchronized trigger – Outputs the trigger signal, synchronized to the acquisition clock.
- Output: ADC clock – Outputs the acquisition clock
- Output: 3.3V – Outputs a 3.3V signal

- Input: Timestamp request – Used with the 'timestamp per digital IO pulse' timestamp mode. An input pulse will result in the PDA16 generating a timestamp.

Related PDA16 library functions: SetDigitalOutputModeD16, SetDigitalOutputEnableD16

2.3.2 The Sampling Rate Tab

This tab contains settings that affect the PDA16 acquisition clock, which defines the sampling rate.

2.3.2.1 Clock Source

This setting determines which clock will be used to clock the acquisition data. The PDA16 has an internal clock implemented by a voltage controlled oscillator and can also be clocked by an external device.

Related PDA16 library functions: SetAdcClockSourceD16

2.3.2.2 Internal/External Clock Rate

When the internal clock is selected as the clock source, the 'Internal clock rate' setting is used to select the desired acquisition rate. This can be any value from 45 MHz to 128 MHz and most* any value from 128 MHz to 170 MHz and will be aligned to a 20 KHz boundary.

*There are a few 'hole' frequency ranges that cannot be reached with the internal clock:

- 128.92 to 129.82
- 140.63 to 142.80
- 154.70 to 158.66

When the external clock is selected as the clock source, the 'External clock rate' setting is used to tell the software the external clock rate. This can be any rate from 1 MHz to 170 MHz.

IMPORTANT: It's important for the operator to ensure that this setting is kept up to date with the external clock rate. The underlying software needs this information to properly configure the PDA16 for normal operation. If the actual external clock is much slower or faster than what is specified by this setting then the firmware may not be able to synchronize properly which can result in bad acquisition data or acquisitions not finishing.

Related PDA16 library functions: SetInternalClockRateD16, SetExternalClockRateD16

2.3.2.3 Divide External Clock To

This setting is used to select the clock division, if any, to apply to the external clock.

Related PDA16 library functions: SetClockDividersD16, SetClockDivider1D16, SetClockDivider2D16

2.3.2.4 Reference Clock

This setting is used to select the internal clock reference. By default, a 10MHz internal reference is used as the reference clock. This internal reference clock is accurate to better than 5ppm. An external 10MHz reference clock can also be used.

Related PDA16 library functions: SetAdcClockReferenceD16

2.3.2.5 Disable Digital Clock Module

This is an advanced setting and should only be used when explicitly told by Signatec. By default, the PDA16 library automatically manages the digital clock module (DCM) selection depending on various parameters like clock source, frequency, and channel count. Checking this checkbox will result in the DCM being disabled.

Unlike most all of the other controls on this form, this option does not correspond directly to a PDA16 hardware setting. Because of this, checking this item will not have an effect on other processes that may be using the PDA16. This setting will not be persisted when hardware settings are saved or loaded. Lastly, this setting will be lost when the PDA16 Scope Application closes.

Related PDA16 library functions: SetManualDcmDisableD16

2.3.2.6 Post-ADC Emulated Clock Division

This setting controls a feature of the PDA16 that emulates clock division by throwing away samples after conversion. This setting can be used with either the internal or external ADC clock.

Related PDA16 library functions: SetPostAdcClockDividerD16

2.3.3 The Trigger Tab

This tab contains settings that affect the PDA16 trigger. These settings relate how data is collected relative to external events.

In the context of a PDA16, a trigger is an event that commences data collection during a data acquisition operation. When the PDA16 is put into one of the acquisition operating modes, the PDA16 will begin digitizing immediately, but acquisition data will not be stored until the card receives a trigger event.

2.3.3.1 Trigger Source

This setting controls the PDA16 trigger source, which defines where trigger events originate. There are two types of trigger sources: internal or external.

When the external trigger is selected then a trigger event is defined by a TTL pulse delivered to the PDA16 via the external trigger connector on the back of the card. (The external trigger input is the third SMA connector from the top and should be labeled.) Further, depending on the state of the External Trigger Direction setting, the trigger can take place on the rising or falling edge of the pulse. The default setting, positive-going, means that a trigger event will be generated on the rising edge of the external TTL signal.

The PDA16 can also be configured to use one of two internal trigger sources: Internal Channel 1 or Internal Channel 2. Both internal sources operate the same way; the only difference between the two is the data channel that is monitored. When using an internal trigger source, a trigger event occurs when the channel's ADC value crosses a particular trigger level in a particular direction. Further, the PDA16 implements two independent trigger level/direction pairs: A and B. If you only need one trigger-level, you can disable one by setting its level to 0.

As an example, suppose the trigger source is Internal Ch. 1, Trigger Level A is 12345, and Trigger Direction A is positive-going. When the board is placed into acquisition mode, it begins digitizing immediately but does store sample values until a trigger is detected. In this particular example a trigger condition will be met when the current ADC value is greater than or equal to the trigger level value (12345) and the previous ADC value is less than the trigger level (i.e. crossing the trigger level in a positive-going fashion.)

Related PDA16 library functions: SetTriggerSourceD16

2.3.3.2 Trigger Mode

The PDA16 supports two types of triggering modes. The trigger mode defines how much data will be acquired for each trigger event.

The default trigger mode is Post Trigger mode. In this mode, when a trigger event is detected, the PDA16 will digitize samples until the total number of samples (over all channels) has been acquired. The total number of samples is defined by the Total Samples setting. This mode is used to acquire a single stream of continuous data.

The other trigger mode is Segmented Trigger mode. In this mode, when a trigger event is detected, the PDA16 will digitize a static number of samples defined by the Segment Size setting. Then the PDA16 will rearm and wait for another trigger. This is repeated until the total number of samples acquired over all segments hits the Total Samples count. This mode is used to acquire multiple discrete segments of continuous data.

Related PDA16 library functions: SetTriggerModeD16

2.3.3.3 Segment Size

This setting is only relevant when using Segmented trigger mode. This setting defines the number of samples that will be acquired per trigger event. This value will be aligned down to an even sample count.

Related PDA16 library functions: SetSegmentSizeD16

2.3.3.4 Pre-Trigger Sample Count

The Pre-Trigger Samples setting allows the board to keep a specified number of samples that occurred prior to the trigger event. This feature allows the operator to look 'back in time' at data prior to the trigger event.

The pre-trigger sample count and trigger delay sample count settings are mutually exclusive.

Related PDA16 library functions: SetPreTriggerSamplesD16

2.3.3.5 Trigger Delay Sample Count

The Trigger Delay Samples setting allows the board to ignore a number of samples after a trigger event. This moves the collected data 'forward in time' after the trigger event.

The pre-trigger sample count and trigger delay sample count settings are mutually exclusive.

Related PDA16 library functions: SetTriggerDelaySamplesD16

2.3.3.6 Internal Trigger Levels

The trigger level settings are used when one of the internal trigger sources are selected. The PDA16 implements two independent trigger level/direction pairs for internal trigger sources. This allows the operator to trigger on data that may pulse in either direction.
The trigger level value represents the ADC value at which, when crossed in the specified direction, will result in a trigger event being generated. See Trigger Source for an example.

The PDA16 Scope Application has two methods of setting the trigger level. The slider control can be used when an exact value isn't necessary. Alternately, clicking the '…' button will open up a window that will allow the operator to specify a specific trigger level value.

Related PDA16 library functions: SetTriggerLevelAD16, SetTriggerLevelBD16

2.3.3.7 Trigger Direction

The trigger direction settings are used to define the direction that defines a trigger event. For the internal trigger directions, A and B, the direction defines the direction that the trigger level must be crossed. For the external trigger, the direction defines whether a trigger is defined by the positive-going (rising) or negative-going (falling) edge of the trigger pulse.

Related PDA16 library functions: SetTriggerDirectionAD16, SetTriggerDirectionBD16, SetTriggerDirectionExtD16

2.3.4 The Status Tab

This tab contains an interface to read PDA16 hardware status items. In addition, version and configuration information for various PDA16 hardware/firmware/software entities are displayed on this tab.

2.3.4.1 PLL Lock Status

This item displays the state of the phase lock loop (PLL) lock status and is only relevant when using the internal ADC clock. A locked PLL means that the acquisition clock is stable.

Related PDA16 library functions: GetPllLockStatusD16

2.3.4.2 RAM FIFO

This item displays the state of the RAM FIFO and is only relevant when acquiring in the RAM-buffered PCI acquisition mode. If this flag is set then the host PC could not keep up with the acquisition data rate. This can happen when the PC is saving acquisition data to disk but the underlying file system cannot write data as fast as the PDA16 is acquiring it.

Related PDA16 library functions: GetFifoFullFlagD16

2.3.4.3 Timestamp FIFO

This item displays the state of the timestamp FIFO and is only relevant when timestamp data is being generated by the PDA16 firmware. This item can be in one of three states:

- Empty – No timestamps are in the timestamp FIFO.
- Not Empty – Timestamp FIFO contains timestamps
- Full – Timestamp FIFO is full; timestamp data may have been lost

Related PDA16 library functions: GetTimestampOverflowFlagD16, GetTimestampAvailabilityD16

2.3.4.4 Refresh Status Flags

Clicking this button will result in the PDA16 status hardware register being read which will result in all items in the PDA16 status flags group being updated.

2.3.4.5 Version Information

This section displays the version numbers of various PDA16 software, firmware and hardware components.

The general format used for version numbers by Signatec is:
Major.*Minor*.*Sub-Minor*.*Package*

Where:

- *Major* is the major version number of the entity. This is usually only incremented when the underlying entity goes through a major change.
- *Minor* is the minor version number of the entity. The minor version number is usually incremented whenever the underlying entity is changed.
- *Sub-Minor* is the sub-minor version of the entity is usually used to indicate pre-release state.
- *Package* is the package version and is only incremented when the underlying entity has not changed, but has been rebuilt or repackaged.

PDA16 firmware – This is the overall PDA16 firmware package number. This is the version number that is referred to in documentation. This item can be updated by uploading PDA16 firmware.

PCI – This is the version of the PCI or PCI-X portion of the PDA16 firmware. The version is displayed for informational purposes only.

SAB – This is the version of the SAB portion of the PDA16 firmware. The version is displayed for informational purposes only.

PDA16 hardware – This is the hardware revision of the current PDA16. This version is assigned during PDA16 hardware initialization and can only be updated by Signatec.

PDA16 driver – This is the version of the underlying PDA16 kernel-mode driver. This is the sole software entity that communicates directly with the PDA16 hardware. This item can be changed by installing newer PDA16 product software.

PDA16 library – This is the version of the main PDA16 user-mode shared library. This is the primary interface to the underlying PDA16 device driver. This item can be changed by installing newer PDA16 product software.

Software release – This is the version of current PDA16 product software installation. This item can be changed by installing newer PDA16 product software.

Related PDA16 library functions: GetVersionTextD16, GetLibVersionD16, GetDriverVersionD16, GetFirmwareVersionD16, GetSabFirmwareVersionD16, GetPciFirmwareVersionD16, GetHardwareRevisionD16, GetSoftwareReleaseVersionD16

2.3.4.6 Hardware Configuration

This section displays the properties of various PDA16 read-only hardware characteristics.

Channel type – This is channel type. As of this writing, there is only one type: “AC-coupled”

Channel filter – This is the type of filter installed for the specified channel. At the time of this writing there are four standard options for channel filter:

- None – No filter is installed
- Low pass – 120 MHz
- Low pass – 210 MHz
- Custom

Master/slave – This is the master/slave hardware configuration. PDA16 devices can be hardware modified to have better performance in master/slave setups. In this case, this item will display the master/slave configuration. Devices with a master/slave hardware modification can still operate as a standalone board. If this item is 'Unspecified' then it has not been specifically modified for master or slave operation and can be configured as either. (See Master/Slave Configuration)

PCI type – This is the PCI bus type that the PDA16 is configured for and can be one of the following:

- PCI-64 (33 MHz)
- PCI-X (100 MHz)
- PCI-X (66 MHz)

The PCI bus type can be reconfigured by the user but is not recommended unless explicitly requested by Signatec. For a best fit between performance and compatibility; PCI-X 66 MHz is the default setting when shipping a PDA16 independent from a Signatec computer system. For turnkey systems created by Signatec; PCI-X 100 MHz or PCI-64 33 MHz might be selected depending on the system level requirements. For continuous recordings to the PC at 160 MHz per channel for both channels; PCI-X 100 MHz will be required to sustain the 640 MB/s data recording rate. See Uploading PDA16 Firmware.

SAB present – This item displays whether or not the Signatec Auxiliary Bus (SAB) is present on the PDA16. If present, the underlying FPGA type that implements SAB logic will be display.

Ethernet present – This item displays whether or not Ethernet is supported for the PDA16. Standard PDA16 boards do not support Ethernet.

Related PDA16 library functions: GetChannelImplementationD16, GetBoardFeaturesD16,

2.3.5 The Timestamps Tab

This tab contains settings that are used to control the PDA16 timestamp features. This tab is only displayed for PDA16 devices in which the firmware supports timestamps.

The PDA16 can be configured to generate timestamps for certain external events. A timestamp is a 64-bit unsigned value that represents a number of clock ticks. Upon transition from Standby operating mode to any acquisition operating mode, the timestamp counter will reset to 0 and increment once per acquisition clock cycle.

As timestamps are generated (defined by the Timestamp Mode), they are inserted into the Timestamp FIFO. Software can read timestamps from this FIFO as the acquisition progresses or after the acquisition completes.

The PDA16 Scope Application can be configured to read timestamp data during acquisition recordings.

Related PDA16 library functions: GetTimestampFifoDepthD16

2.3.5.1 Timestamp Mode

This item is used to set the PDA16 timestamp mode. The timestamp mode determines how the PDA16 generates timestamps. Currently defined timestamp modes are:

- No timestamps – The PDA16 will not generate timestamps

- Timestamp per segment – The PDA16 will generate a timestamp at the start of each segment acquired while in segmented trigger mode.
- Timestamp per external trigger – The PDA16 will generate a timestamp for each pulse received on the external trigger.
- Timestamp per digital IO pulse – The PDA16 will generate a timestamp for each pulse received on the digital IO connector. In order for this timestamp mode to function, the digital IO mode must be selected as 'Input: Timestamp request' and digital IO must be enabled. This mode is useful if your time-stamp pulse is independent of the trigger signal.

Related PDA16 library functions: SetTimestampModeD16

2.3.5.2 Reset Timestamp FIFO Now

Clicking this button will result in the immediate resetting of the timestamp FIFO. After this operation the timestamp FIFO will be empty. This operation will have no effect on the timestamp counter.

Related PDA16 library functions: ResetTimestampFifoD16

2.3.5.3 Reset Timestamp Counter Now

Clicking this button will result in the immediate resetting of the timestamp counter back to zero. This operation will have no effect on the timestamp FIFO.

Related PDA16 library functions: ResetTimestampCounterD16

2.3.5.4 Drain Timestamp FIFO Now

Clicking this button will drain all timestamps from the timestamp FIFO by pulling out all timestamps until the FIFO is empty. The total number of timestamps removed from the FIFO will be displayed to the operator.

Related PDA16 library functions: GetTimestampFifoDepthD16, GetTimestampAvailabilityD16, ReadTimestampDataD16

2.3.5.5 Save Timestamp Data to File

Clicking this button will read and save all timestamps to the specified file. If the 'Save as text' option is checked then timestamps will be written as new-line delimited text to the specified file. If not checked, then timestamp data will be written as binary data. There is no header or context information, the first 8 bytes are the first timestamp, the next 8 bytes are the next timestamp, and so on.

Note: In continuous record mode (or PCI Buffered Acquisition mode), the PDA16 Scope Application can automatically read and save timestamp data during the recording. See Destination File Advanced Options.

Related PDA16 library functions: GetTimestampFifoDepthD16, GetTimestampAvailabilityD16, ReadTimestampDataD16

2.3.6 The FPGA Processing Tab

This tab contains settings that are used to control FPGA processing features of the PDA16. This tab is only displayed for PDA16 device in which FPGA processing is available.

2.3.6.1 Enable FPGA Processing

Check this option to enable FPGA processing. The actual processing performed is dependent on the underlying PDA16 firmware. This feature allows customers with FPGA development packages for the PDA16 to use the PDA16 Scope Application as a run-time test environment for their custom developed logic.

Related PDA16 library functions: SetBoardProcessingEnableD16

2.3.6.2 FPGA Processing Parameters

This section is used to specify arguments to the FPGA processing parameters. There are 65536 16-bit registers [0, 65535] that are used to specify various parameters to the FPGA processing. The actual interpretation of these values is dependent on the underlying PDA16 firmware. This feature allows customers with FPGA development packages for the PDA16 to use the PDA16 Scope Application as a convenient PC application interface for writing dynamically to their own custom logic registers.

Related PDA16 library functions: SetBoardProcessingParamD16

2.3.7 The SAB Tab

This tab contains settings that are used to control Signatec Auxiliary Bus (SAB) settings. The Signatec Auxiliary Bus is an external bus that is independent of the system PCI/PCI-X bus that can be used to connect a PDA16 to one or more external devices. The PDA16 can acquire data directly to a peer device over the SAB bus.

2.3.7.1 SAB Board Number

This item selects the Signatec Auxiliary Bus (SAB) board number for the PDA16. Each device on a SAB bus must be assigned a unique non-zero number. A SAB number of zero (the default) means that the device will not be used on the SAB. The actual value that's selected isn't important as long as its non-zero and unique among all other devices on the local SAB.

Related PDA16 library functions: SetSabBoardNumberD16

2.3.7.2 SAB Configuration

This item selects the Signatec Auxiliary Bus (SAB) configuration and can be any of the following:

- 64-bit – The PDA16 will use all 64-bits of the SAB bus for SAB data acquisitions or transfers.
- 32-bit (lower) – The PDA16 will use only the lower 32-bits of the SAB bus for SAB acquisitions or transfers.
- 32-bit (upper) – This SAB configuration is currently not supported, but may be implemented in a future PDA16 firmware. In this configuration, the PDA16 will use only the upper 32-bits of the SAB bus for SAB acquisitions or transfers.

Related PDA16 library functions: SetSabConfigurationD16

2.3.7.3 SAB Clock

This item selects the SAB clock to use for SAB acquisitions or transfers and can be any of the following:

- Use acquisition clock – This clock should be selected when doing a SAB acquisition. With this selection, the main acquisition clock will be used to strobe the SAB bus.
- 62.5 MHz (Transfer) – The 62.5 MHz clock will be used to strobe the SAB bus; this should only be used with SAB transfer operations. This is the maximum rate when Signatec's PMP1000 is receiving the data.

- 125 MHz (Transfer) – The 125 MHz clock will be used to strobe the SAB bus; this should only be used with SAB transfer operations.

This item is only relevant when the PDA16 is sending data over the SAB.

Related PDA16 library functions: SetSabClockD16

2.3.8 The Data Plotting Tab

This tab contains settings that affect how data is displayed in the Scope panel.

2.3.8.1 Zoom To Selection

This button will be enabled when an extended selection is made in the plot area. When this button is clicked it will zoom in horizontally such that the selected region fills the plot area.

2.3.8.2 Clear Scope Data

Clicking this button will remove all plot sources from the plot area. This includes displayed PDA16 RAM data, recording snapshots, and external files.

2.3.8.3 Add Source

Clicking this button will allow the operator to select a data file to plot in the Scope panel. When a file is selected, the Scope Application will check to see if a corresponding SRDC file is available and if present, data type and channel count will be obtained from the SRDC data.

2.3.8.4 Auto Set

Clicking this button will attempt to automatically adjust vertical scaling such that all data fits into the plot area.

2.3.8.5 Plot Source List

This list control lists all current plot sources. The checkboxes are used to toggle individual channel visibility.

2.3.8.6 Source Info

This button is enabled when a plot source is selected in the plot source list control. Clicking this button will open up a property page that displays information on the currently selected data source.

2.3.8.7 Go To…

Clicking this button will open up the Go to Sample or Segment in Plot window. This is equivalent to selecting "Go to Sample…" in the View menu.

2.4 The Scope Panel

The Scope panel is where PDA16 acquisition data is displayed. This can be data that is currently in the PDA16 RAM, a data snapshot from the current data recording, or data previously saved to a file. The plot area can display multiple channels of independent or interleaved data of varying sample sizes (8-, 12-, 14-, 16-, 32- bit) and types (signed, unsigned, floating point) simultaneously.

At the time of this writing, the units displayed for the data are sample number (horizontally) and sample value (vertically). A future version of the application may extend this to support time and/or voltage units.

The 'Data Plotting' tab in the Settings area controls many of the data plotting parameters. This includes scaling parameters, channel visibility, and channel source information. In addition, the mouse may be used to alter the horizontal and vertical scaling of the data as well as panning through the data.

If the plot area is zoomed in sufficiently, data points will be discretely visible. When in this mode, holding the mouse pointer over the data point should result in a tooltip with the channel identifier and sample value being displayed in a tooltip window.

Plotted data is read-only; there are no facilities to modify data with this interface.

2.4.1 Mouse Controls

The mouse may be used to zoom and pan plotted data. In this section 'plot area' represents the blue area where data is actually plotted and 'outside plot area' represents the margins outside of the blue data plot area.

Left-click in plot area: Set the main selection point to the sample position closest to where the mouse was clicked. This will also result in a vertical trace being displayed in the plot area. The sample position will also be marked under the trace in the lower horizontal axis label area.

Left-click outside plot area: This will remove the current selection

Shift left-click: Extends the selection from the main selection point to the sample position closest to where the mouse was clicked. This will result in a second vertical trace being displayed as well as the highlighting of all samples (inclusively) between the two selection points.

Mouse wheel in plot area: Rolling the mouse wheel will zoom in/out horizontally on the data under the sample position closest to the mouse position. (Note: Plot area must have mouse focus for this to work so if mouse wheel has no effect, click once somewhere on the Plot area pane to give it focus.)

Ctrl + mouse wheel in plot area: Rolling the mouse wheel while the Control button is held down will zoom in/out vertically on the data.

Right-drag in plot area: Holding down the right mouse button and dragging will horizontally pan data left or right depending on the direction of the drag. (Note: Right-drag panning doesn't work as well when you're zoomed in real tight on the data. In this case, use the scroll bar.)

Ctrl + right-drag in plot area: Holding down the Control key while right dragging will allow data to be vertically panned up or down.

2.4.2 Keyboard Controls

When the scope control has keyboard focus, certain keys have an effect on the scope control:

Left/Right Arrow: Scroll one 'line' left/right. This is equivalent to clicking the arrow button on the scroll bar.

Page Up/Down: Scroll one page up/down. This is equivalent to clicking in the 'open' area of a scroll bard.

Home: Scroll all the way back to the beginning of the data.

End: Scroll all the way to the end of the data.

2.5 PDA16 Scope Application Main Menu Options

2.5.1 The File Menu

2.5.1.1 New PDA16 Device

This item will only be enabled if there are local PDA16 devices that are not already open in the application.

Selecting this item will open up a new local PDA16 instance.

2.5.1.2 Connect to Remote Device

This item will open up a user interface that can be used to connect to a PDA16 on a remote computer. See Remote PDA16 Devices.

2.5.1.3 Close Device

Select this item to close the current PDA16 device. If an acquisition or recording is currently in progress it will be stopped. Opening and closing devices is more relevant for remote device connections. Users that run the PDA16 Scope Application on a machine that contains local PDA16 boards will most likely never have a need to use this feature.

2.5.1.4 Open and Apply Board Settings

Select this item to load PDA16 hardware settings previously saved with the Save Board Settings menu item. Loading board settings will implicitly put the board into Standby mode.

Related PDA16 library functions: LoadSettingsFromFileXmlD16

2.5.1.5 Save Board Settings

Select this item to save all PDA16 hardware settings to a file that can be opened and applied to the hardware at a later time.

Related PDA16 library functions: SaveSettingsToFileXmlD16

2.5.1.6 Save Board Data

Selecting this item will open the 'Save PDA16 RAM Data to File' dialog. This dialog can be used to save data in PDA16 RAM or the current recording snapshot to a file on the local computer.

Data in PDA16 RAM – The region specified by the starting address and sample count will be transferred from the PDA16 RAM to the specified file on the host computer.

- The Use Current button will use the active memory region currently specified on the Data tab.
- The All RAM button will select the entire PDA16 RAM
- The Scope Data button will select the RAM region that is currently displayed in the Scope panel.

Current data snapshot – This item is only enabled when a recording snapshot is present in the Scope panel. When this item is checked the application will save the recording snapshot to the specified file.

Destination File – The pathname of the destination file. The application can save PDA16 data in a variety of ways; see Destination File Advanced Options.

Related PDA16 library functions: ReadSampleRamFileFastD16, ReadSampleRamFileD16

2.5.1.7 Plot Data in Scope

Selecting this option will allow the operator to select a file to be displayed in the Scope panel. This is equivalent to selecting the 'Add Source…' button on the Data Plotting tab.

2.5.2 The View Menu

2.5.2.1 Frequency Domain Window

Selecting this item will toggle the visibility of the Frequency Domain window. See FFT Analysis of PDA16 Data for details on this window.

2.5.2.2 Device Driver Statistics Window

Selecting this item will toggle the visibility of the PDA16 Driver Statistics window. This window displays some PDA16 statistics collected by the PDA16 device driver. An instance of this window is shown below.

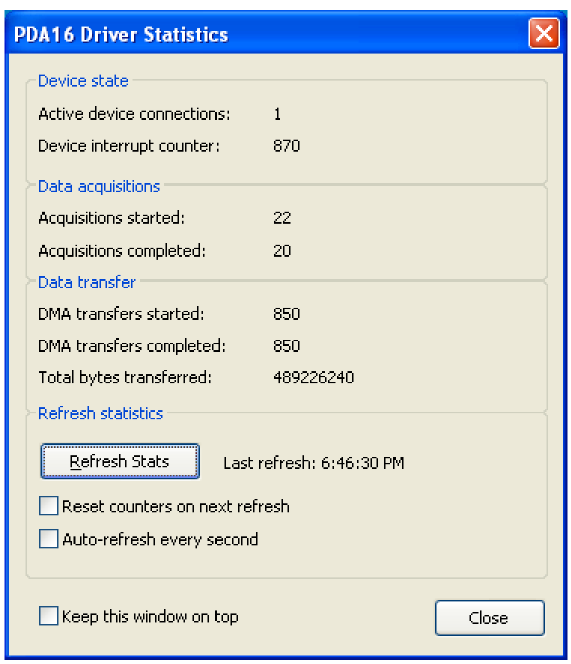

Figure 2-2: PDA16 Device Driver Stats Window

The Refresh Stats button will query the PDA16 driver for fresh data. It is safe to query driver statistics while the PDA16 is in operation.

These statistics are for the PDA16 device in general and are aggregated from all threads and processes. This includes custom PDA16 software written by end users. This makes the data displayed on this window useful for troubleshooting PDA16 problems. Using this window you can determine if data is currently being moved, if transfers or acquisitions are not completing, etc.

Active device connections – This is the total number of active PDA16 device connections. (The number of open handles to the underlying PDA16 device driver.)

Device interrupt counter – This is the total number of times that the PDA16 device driver interrupt handler is invoked. In most cases, this is the total number of times that the PDA16 has interrupted the system. If the underlying interrupt line is being shared with another device, this value may be greater than the total number of PDA16 interrupts.

Acquisitions started – This is the total number of acquisitions started. This includes both RAM and RAM buffered PCI acquisitions.

Acquisitions completed – This is the total number of acquisitions completed. Or more specifically, this is the total number of 'Samples Complete' interrupts generated by the PDA16. Free-run acquisitions (e.g. RAM buffered PCI acquisitions) will never generate a Samples Complete interrupt.

DMA transfers started – This is the total number of DMA transfers started. DMA transfers are the primary method of obtaining PDA16 acquisition data. DMA transfers are used to obtain data in acquisition recordings as well as obtain data for display in the Scope panel.

DMA transfers completed – This is the total number of DMA transfers completed.

Total bytes transferred – This is the total number of bytes transferred (via DMA) between the PDA16 and host PC.

Reset counters on next refresh – If this checkbox is checked when the Refresh Stats button is checked then the driver will zero out all of the counters (except for the active connection count). The stats are zeroed after they are copied for the refresh operation.

Auto-refresh every second – If this checkbox is checked then the stats will be automatically refreshed every second.

Keep this window on top – If this checkbox is selected then the Driver Stats window will be a topmost window which means other windows (even from other applications) will not obscure it.

2.5.2.3 Go to Sample

Selecting this option will display the Go to Sample or Segment Window. This window is a modeless dialog that can be used to navigate to specific offsets in the plot area. Further, if the scope source is segmented (e.g. data acquired using the segmented trigged mode) the window can be used to jump to specific data segments. An instance of this window is shown below:

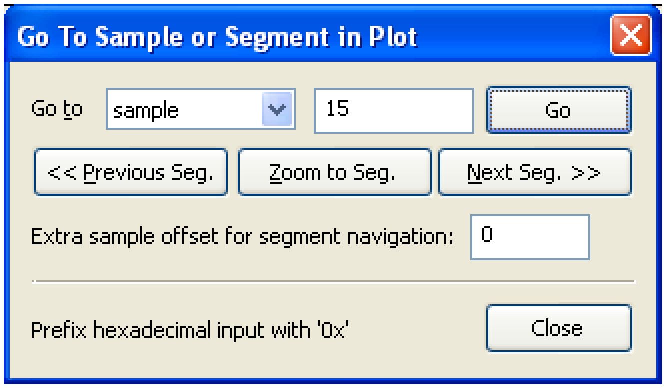

Figure 2-3: Go To Sample or Segment Window

Go to – This line is used to select the sample or segment to navigate to. The combo box allows for sample or segment selection. The edit box contains the sample/segment (whichever is selected in the adjacent combo box) to jump to. Both samples and segment counts begin at 0. Decimal input is assumed. Hexadecimal input is allowed, but the number must be prefixed with '0x'.

Go – Click this button to jump to the desired sample or segment.

Previous Seg. – Clicking this button will the plot to the previous data segment. This button is only enabled when the scope contains segmented data. When this button is clicked the *Go to* portion of the user interface is updated to display the new segment index.

Next Seg. – This item is just like the *Previous Seg* button but moves forward through segments instead of backward.

Zoom to Seg. – Clicking this button will adjust the horizontal scaling such that a segment fits in exactly one page. This item is only enabled when the scope contains segmented data.

Extra sample offset for segment navigation – If this item is non zero it can be used to offset the actual sample jump when navigating between segments. A negative value will display points before the segment and a positive value will display points after the segment. If zero, no offset is applied and the leftmost sample will be the first sample of the segment.

2.5.3 The Hardware Menu

2.5.3.1 Set Power-Up Defaults

Selecting this item will reset all PDA16 hardware settings to their default value. This will implicitly put the board into Standby mode which will cancel any current acquisitions or transfers.

For nearly all hardware settings, the default setting is the 'zero' value. For combo-box items this will be the first selection in the drop down list. An exception to this is that trigger levels are set to midscale.

Related PDA16 library functions: SetPowerupDefaultsD16

2.5.3.2 Rewrite Hardware Settings

Selecting this item will rewrite all PDA16 hardware settings back to the hardware.

Related PDA16 library functions: RewriteHardwareSettingsD16

2.5.3.3 Copy Settings from Other PDA16

Selecting this item will allow the operator to copy the hardware settings of another PDA16 device.

Related PDA16 library functions: CopyHardwareSettingsD16

2.5.3.4 Refresh Local Settings from Driver

Selecting this item will have the PDA16 Scope Application update its current hardware settings from the PDA16 driver cache. This will result in no hardware access; only the driver's hardware register cache is consulted. Often this is all that's necessary to get the current hardware settings. Since all hardware access goes through the PDA16 driver, it will always know the state of the hardware registers. This of course excludes status registers which can change at any time.

Related PDA16 library functions: RefreshLocalRegisterCacheD16

2.5.3.5 Refresh Local Settings from Hardware

Selecting this item will have the PDA16 Scope Application update its current hardware settings by reading the PDA16 hardware registers.

Related PDA16 library functions: RefreshLocalRegisterCacheD16

2.5.3.6 Uploading PDA16 Firmware

This item is used to upload PDA16 firmware. Signatec periodically updates the PDA16 firmware to add new features or correct minor bugs. This updated firmware can usually be downloaded from the Signatec website and comes in the form of a .pda16w file.

Note: Firmware uploading for remote devices is not supported.

Selecting this item will open up a standard file dialog that the operator can use to select the firmware file to upload. Once selected, the Upload PDA16 Firmware dialog is displayed.

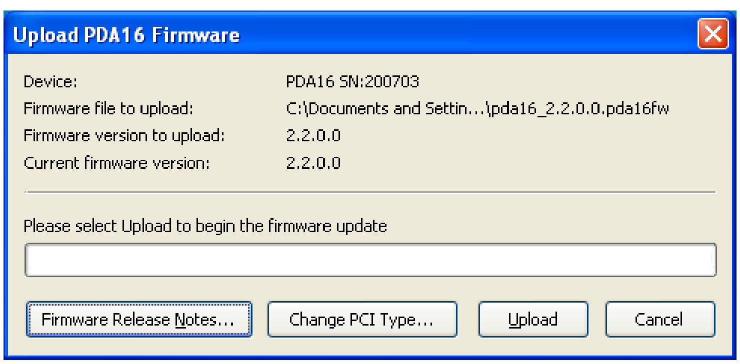

Figure 2-4: Upload PDA16 Firmware

Firmware Release Notes – Clicking on this button will display the PDA16 firmware release notes if any are available.

Change PCI Type – Clicking this button will open up a window that allows the operator to change the PCI bus configuration. This is an advanced operation and should only be performed when explicitly requested by Signatec.

Upload – Clicking this button will begin the firmware uploading process. The entire operation can take a few minutes.

IMPORTANT: Do not interrupt a firmware upload once it has begun. If the firmware upload process is not allowed to finish, the firmware may be lost and the card will not be recognized by the system. In this case, the board will need to be returned to Signatec so that it can have its firmware uploaded via an alternate method that requires additional hardware.

NOTE: During the firmware upload process there may be times where the progress does update for a while. This is normal. During these long waits, a countdown timer will be displayed on the progress window.

Related PDA16 library functions: UploadFirmwareD16

2.5.4 The Tools Menu

This menu contains a few miscellaneous utility items that operate on PDA16 data.

2.5.4.1 Data Conversion: Convert Data to WAV format

This item is used to generate a waveform audio format (*.wav) file from data in a PDA16 acquisition data file.

Selecting this item will open up a standard file dialog that is used to browse for the source PDA16 acquisition data file (*.rd16). The PDA16 Scope Application will pull additional information (channel count, sampling rate, etc) from the corresponding auxiliary SRDC file.

The output will be a WAV file with the same pathname as the source file but with a '.wav' file extension.

2.5.4.2 Recorded Data Details

This item is used to view details about a PDA16 acquisition data file. These details are obtained from the auxiliary SRDC file that Signatec software can automatically generate. These details include things like channel count, sampling rate, input voltage range, operator notes, etc.

Related PDA16 library functions: OpenSrdcFileD16, GetRecordedDataInfoD16

2.6 Recording PDA16 Data

The PDA16 Scope Application has the ability to record acquisition data from one or more PDA16 devices to a file. Data recordings are controlled by clicking the 'Record Data…' button in the Control panel, which will open the 'Record PDA16 Data' window.

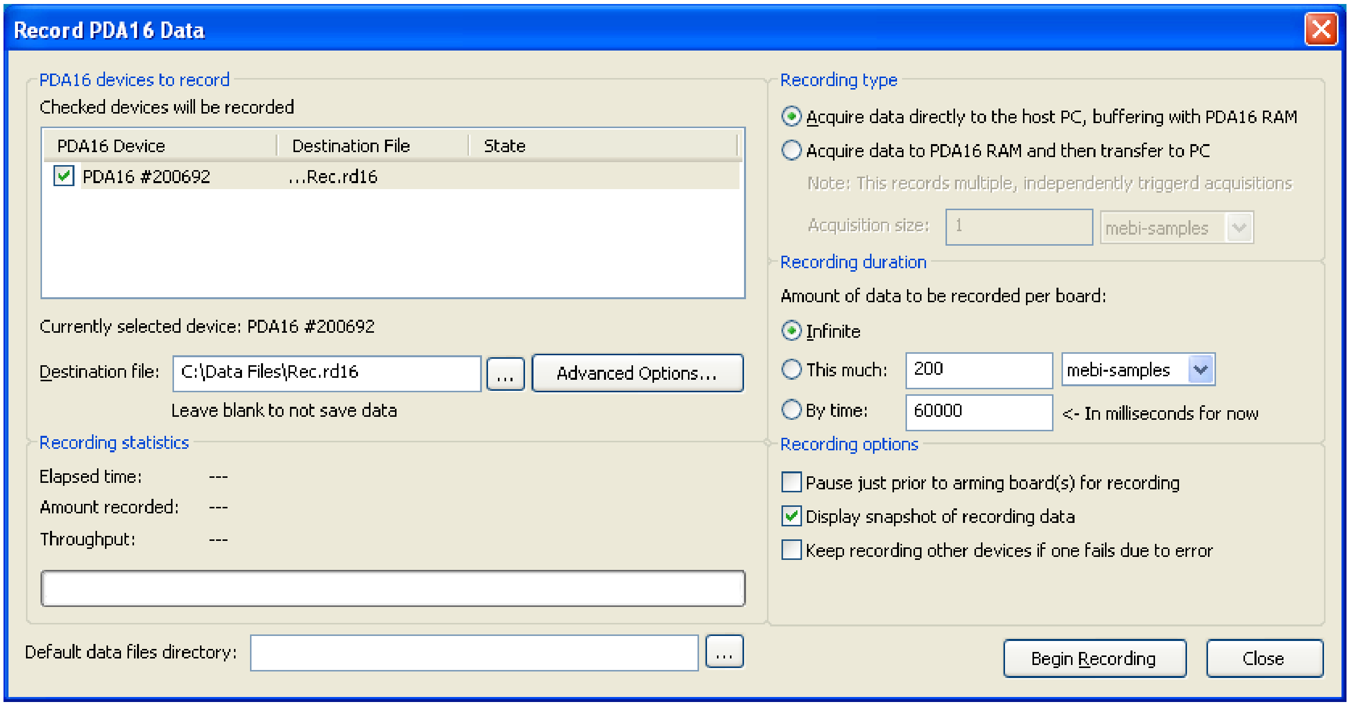

Figure 2-5: Record PDA16 Data Window

2.6.1 Device and Destination File Selection

To select the device(s) to record, check the desired device in the list control. To specify a destination file for the recorded data, select the device in the list control and specify the pathname in the 'Destination file' edit box. (The … button will open up a file dialog to browse for the destination file.)

Remote PDA16 Devices: Note that if the underlying PDA16 device is remote (on another computer) that pathnames specified are relative to the remote server system, not the host system. Recording remote devices will result in data being saved on the remote system, not the host system. Also, clicking on the browse ('…') button for a remote device will open up an emulated file dialog that will allow for browsing on the remote server system.

If no destination file is specified then acquisition data will not be saved. This can be useful in determining if the host system can sustain recordings of a particular acquisition rate without considering the underlying file IO limitations.

The native file format for recorded PDA16 acquisition data is the RD16 file format. The RD16 moniker is derived from "Raw Data 16-bit". RD16 files are identified by the '.rd16' file extension. RD16 files are binary files that contain only acquisition data. There is no file header or additional information in the file. The first two bytes of the file are the first data sample. This simple file format has two big advantages. First, it's very fast to write these files since data is written to the file exactly as it is received from the PDA16. If the underlying file system (e.g. a high-speed RAID system) can keep up with the data rate, data can be streamed from the PDA16 card to the file at the full acquisition rate. The second advantage is that this file format is very generic which makes it easy for other software to use the data. This includes custom software, or other software environments like Matlab or LabVIEW.

The main disadvantage of the RD16 file format is that no context information is stored in the file and the details of the data may not be apparent. Important details such as channel count, voltage range selection and sampling rate are unknown by looking at the raw data alone. To get around this problem, Signatec software can also be configured to generate a small auxiliary context file that sits in the same directory as the RD16 file. This auxiliary file can contain information such as channel count, input voltage range, sampling rate, source board, operator notes, or any other user-defined data. These auxiliary files are referred to as Signatec Recorded Data Context (SRDC) files and are identified by the .srdc file extension. SRDC files are written in XML format and are easily read by any XML-aware software.

See the PDA16 Operator's Manual for more details on SRDC file contents.

See Destination File Advanced Options for alternate methods of saving recording data, including the enabling of SRDC file generation.

2.6.2 Recording Type

The PDA16 Scope application supports two types of data recordings which are described in the following sections.

2.6.2.1 Acquire data directly to host PC, buffering with PDA16 RAM

This type of recording, also called a RAM-buffered PCI recording, is used to record one long continuous stream of data (or one long continuous stream of discrete data segments if in segmented mode) in which the PDA16 RAM is used to buffer the data during recording. This is the most common type of data recording.

In RAM-buffered PCI recording, the underlying destination disk drive(s) must be able to sustain the acquisition data rate. If the acquisition rate is too fast (or the disk write speed is too slow) then the PDA16 RAM FIFO will overflow and data will be lost. In this case, the PDA16 software will catch this condition and end the recording with an error message.

2.6.2.2 Acquire data to PDA16 RAM and then transfer to PC

This type of recording is used to record a series of non-contiguous acquisitions. First, new data is acquired to RAM and then when the acquisition has completed, the data is transferred to the host system. It's important to note here that while data is being transferred to the system no new data is being acquired.

This type of recording is good for recording at the full acquisition rate which may be too fast for a RAM-buffered PCI recording.

2.6.3 Recording Duration

The duration of the recording is determined by the settings in the 'Recording duration' group box. There are currently three types of duration supported.

- Infinite – Recording goes on indefinitely until manually stopped by the operator.
- This much – Records the amount of data specified, per-board. (Note: gibi-, mebi-, and kibi- prefixes denote 1073741824 (2^{30}), 1048576 (2^{20}), and 1024 (2^{10}) respectively.)
- By time – Converts the given time into an equivalent sample count (which is a function of acquisition rate) and then records that much data. It should be noted that this time is entire time of recorded data, not including the time to wait for a trigger event.

Regardless of the recording duration type selected, a recording may be manually stopped by clicking the 'Stop Recording' button the Recording window.

2.6.4 Recording Options

2.6.4.1 Pause just prior to arming board(s) for recording

Checking this item will result in the Scope Application displaying a message box after setting everything up for the recording and just prior to actually arming the boards for recording.

2.6.4.2 Display data snapshot of recording data

Checking this item will result in a snapshot of the current recording data being displayed in the plot area. Currently, an 8192 point snapshot will be obtained roughly once a second. A future version of the application may allow for further customization of the data snapshot.

2.6.4.3 Snapshot Options

Clicking this button will open up the Recording Snapshot Options dialog. This dialog can be used to control the size and frequency of the recording data snapshot.

2.6.4.4 Keep recording other devices if one fails due to error

If this item is checked then the recording will continue if one device fails in a multiple device recording. By default, if one device fails, the entire recording is halted.

2.6.4.5 Default data files directory

This item specifies the default directory that will be used for data files when no path is specified.

2.6.5 Running a Recording

Once all recording parameters have been set, a recording is started by clicking the ‘Begin Recording’ button. The Scope application and underlying PDA16 library/driver code will manage all aspects of the recording.

The ‘Recording statistics’ group contains current recording statistics including progress (for concrete-length recordings) and data throughput. Individual device throughputs are displayed in the device list control.

Related PDA16 library functions: CreateRecordingSessionD16, ArmRecordingSessionD16, GetRecordingSessionProgressD16, GetRecordingSnapshotD16, AbortRecordingSessionD16, DeleteRecordingSessionD16

2.7 Destination File Advanced Options

2.7.1 Append data if it already exists

If this item is selected then data will be appended to the end of the file destination file if it already exists. By default, files are overwritten if they exist.

2.7.2 Deinterleave this data into separate files

If this item is selected then the data will be de-interleaved into two separate files. Channel 1 data will be written to the primary pathname specified on the parent window. Channel 2 data will be written to the file specified in the adjacent edit box. The '…' button will open up a file dialog to browse for the destination file.

Note that that software is responsible for de-interleaving the data. This means that if data is being saved as part of a data acquisition recording, this option may decrease the overall data throughput. The processing involved in de-interleaving data at high rates can be very demanding. Signatec cannot guarantee that de-interleaved data recordings can be sustained on all recording platforms and for all data recording rates.

2.7.3 Span data over multiple, statically sized files

This item is used to span acquisition data over multiple files instead of a single file. This can be useful in situations where multiple, smaller files are preferred over a single large file. When this item is checked, the specified destination file name is used as a template. The software will take the template filename and append a '_#' (number) for each file generated where # is an incrementing value starting with 0.

The size of the file segment is defined by the 'Static file size' item. For acquisition recordings, it is recommended to use an integer number of mebi-samples (1 mebi-sample = 2^{20} = 1048576 samples) and to not create new data files anymore frequent than about once per gigabyte when recording at very high data rates.

2.7.4 Leave room for application-specific data

This option applies when writing binary files.

If this item is checked then a number of bytes will be set aside for user-defined header data. The bytes are reserved by advancing the file pointer prior to writing file data; this will result in zeroes being written. Other software can then write application-specific data to the front of the file at a later time.

If the 'Skip even when appending data to an existing file' option is selected then the user-defined header data will always be reserved, regardless of whether a new file is being created or an existing file is being appended to. By default, if this item is not checked, the user-defined data region will only be used when creating a new file.

2.7.5 Save data as text

If this item is checked then data will be saved in ASCII text format instead of the default binary format. This option can be demanding on the CPU and might not be sustainable for high data rate recordings.

When saving as text, sample values will be written to the file in decimal format, one sample per line. That is, each sample will be newline ("\r\n") delimited.

Use hexadecimal output – If this item is checked then samples will be written in hexadecimal.

Assumed dual channel data; column per channel – If this item is checked then data will be written two samples per line: *Channel 1* <tab> *Channel 2* <newline>.

2.7.6 Don't try to use fast, unbuffered IO for binary files

If this item is checked then the software will not try to use unbuffered file writes. Unbuffered file writes can be performed faster than normal, buffered writes at the expense of having to align file write sizes and offsets to integer multiples of the underlying file system's sector size. By default, the software will analyze all file write parameters and determine if unbuffered writes are allowed and, if so, will use unbuffered writes.

2.7.7 Convert data to signed format before writing

If this item is checked then all PDA16 data will be converted to signed format prior to writing.

At the time of this writing, the PDA16 can only acquire unsigned data, so this conversion is done entirely in the software. Enabling this feature for acquisition recordings can result in decreased data throughput.

2.7.8 Save PDA16 timestamps to external file (*.pda16ts)

If this item is checked then PDA16 timestamps will be read and saved in an external file during recording. Each timestamp is a 64-bit unsigned value that represents the number of acquisition clock ticks that has elapsed since the board was armed for acquisition. The timestamp mode determines how the PDA16 will generate timestamps.

By default, the name of the timestamps file will be the filename of the recording destination file, but with a 'pda16ts' file extension. The '.pda16ts' file format is a simple binary file format that contains only the raw timestamp data; there is no header or any other additional information in the file.

Save timestamps as newline-delimited text (.pda16ts.txt)* – If this checkbox is selected then timestamps will be written as text instead of the default binary format. The format of the file is one decimal timestamp per line. The default filename will be the filename of the recording destination file, but with a '.pda16ts.txt' extension.

Use timestamp FIFO overflow marker – If this checkbox is selected then software will insert a special timestamp FIFO overflow marker into the timestamp data if the timestamp FIFO overflows at any point during the recording. The timestamp marker is two consecutive timestamp values of 17,433,700,174,704,734,704 (or F1F0F1F0F1F0F1F0 in hexadecimal). When the timestamp FIFO is detected as full, all timestamps the FIFO is emptied of timestamps and then the FIFO overflow marker is inserted into the timestamp stream.

Override timestamp file pathname – This item allows the operator to override the default timestamp pathname. Selecting the adjacent ellipses ('…') button will open a file dialog.

Abort and fail recording if timestamp FIFO overflows – If this item is checked then the recording will be halted with an error if the timestamp FIFO ever goes full during the recording.

2.7.9 Signatec Recorded Data Context (SRDC) Information

SRDC files are described in the Recording: Device and Destination File Selection section.

Note: The SRDC options below are not mutually exclusive.

Save details in auxiliary Signatec Recorded Data Context (.srdc) file – If this checkbox is selected then SRDC information will be written to an external file. The pathname of the SRDC file will be the full pathname of the recording destination file appended with a '.srdc' extension. If, for example, the recording destination file is named "C:\My Data\Recording.rd16" then the generated SRDC file will be "C:\My Data\Recording.rd16.srdc".

Embed details in alternate file stream of output file(s) – If this checkbox is selected then SRDC information will be written into an alternate file stream (AFS) of the recording destination file. The SRDC file stream will be named 'SRDC'. If, for example, the recording destination file is named "C:\My Data\Recording.rd16" then the SRDC stream will be in "C:\My Data\Recording.rd16:SRDC".

Alternate file streams are a feature of the NT File System (NTFS), which allow alternate, independent streams of data to coexist in a file. These alternate streams are analogous to multiple files being contained in a directory. Consult Microsoft documentation for more details on Alternate File Streams.

NOTE: If a recording data file that contains an SRDC file alternate file stream is moved to a file system (e.g. a USB flash drive, or moving a file across a network) that does not support NTFS, then the SRDC data will be lost.

Operator notes – This is an optional note that will be saved in the SRDC file. This is intended for user-defined notes that are relevant to the recording.

2.8 FFT Analysis of PDA16 Data

The PDA16 Scope Application has the ability to do FFT analysis on PDA16 (or file) data. FFT operations are performed on data sources that are currently displayed in the Scope panel and are displayed on the Frequency Domain (FFT) window. The FFT window is displayed by selecting the 'Frequency Domain Window' option in the View menu. Alternately, pressing Ctrl + Shift + F will display the FFT window. A sample FFT window is shown in the figure below.

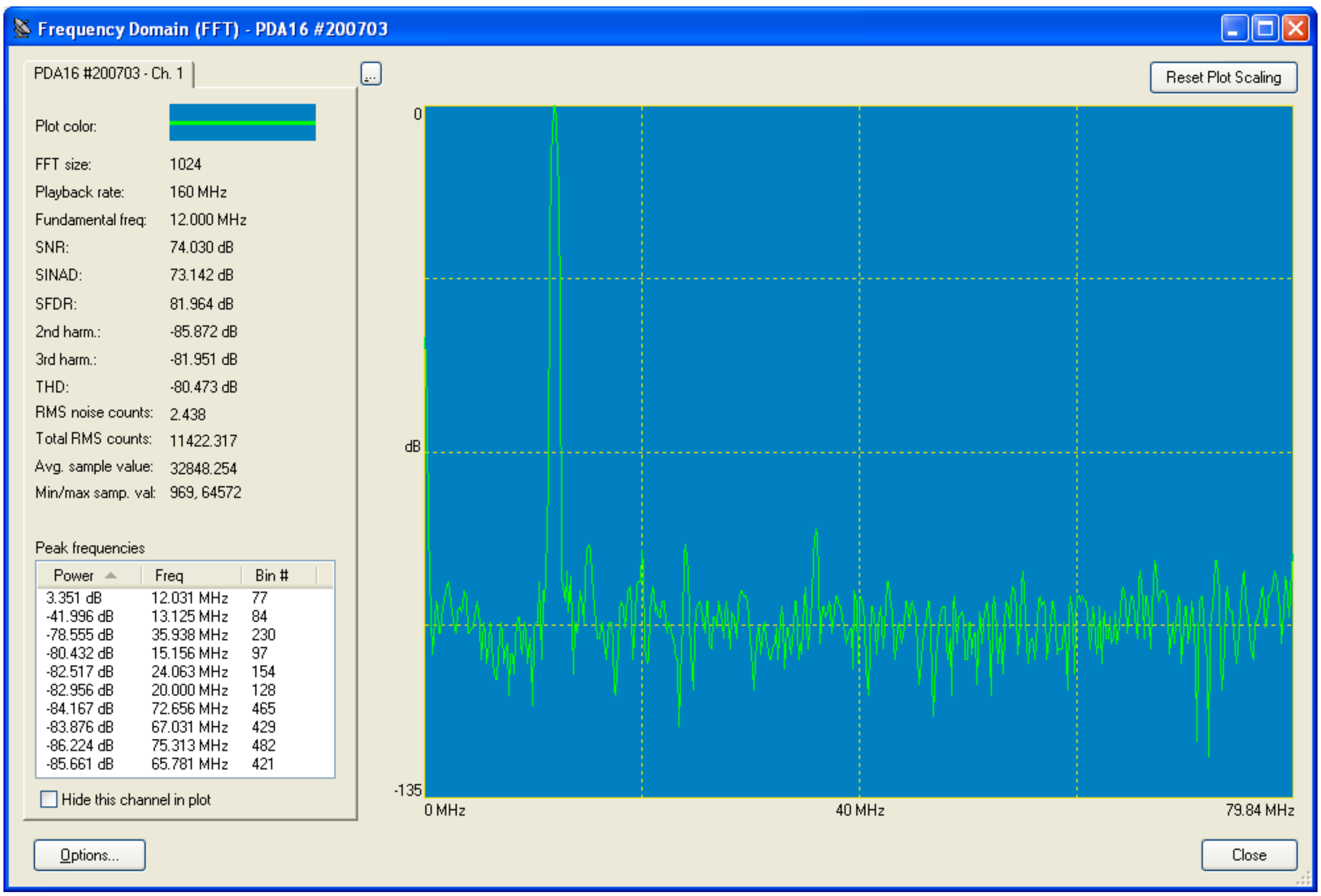

Figure 2-6: Frequency Domain (FFT) Window

The FFT window is divided up into two regions. The left side is a tab control that contains various FFT and time-domain statistics. There is one tab for each channel of data. The right side contains the frequency domain plots of all channels.

2.8.1 Frequency and Time Domain Statistics

FFT Size – The number of time-domain samples used in the FFT operation. The FFT size can be changed on the FFT Options dialog.

Acquisition rate – This is the underlying acquisition rate, in MHz, used to obtain the time domain data. When the PDA16 is the data source, the current acquisition rate is used. When an external file is the data source, this information is obtained from SRDC data. If acquisition rate information is not available this item will be 0.

Fundamental frequency – This is the frequency component with the highest power. In order for this to be calculated properly, the underlying acquisition rate must be known.

SNR – This is the signal-to-noise ratio and is a ratio of the signal power to the noise power.

SINAD – This is the signal to noise and distortion ratio and is a ratio of the signal power to the sum of the noise, 2^{nd}, and 3^{rd} harmonic power.

SFDR – This is the spurious free dynamic range and is a ratio between the fundamental power and the next highest spur power.

2^{nd} harmonic – This is the power of the second harmonic.

3^{rd} harmonic – This is the power of the third harmonic.

THD – This is the total harmonic distortion and is a ratio of the sum of the 2^{nd} and 3^{rd} harmonics to the fundamental signal power.

RMS noise counts – This is the root mean square of the noise power.

Total RMS counts – This is the root mean square of the signal power.

Average sample value – This is the average time-domain sample value in the data used for the FFT operation.

Minimum/maximum sample value – This is the minimum and maximum time-domain sample values from the data used for the FFT operation.

Peak Frequencies – This list tracks the highest power spurs in the current frequency domain data. Double clicking on any of these items will move the selection in the frequency domain plot to that peak.

Hide this channel in plot – If this item is checked then the currently selected data channel (relative to the tab) will be hidden from the plot.

FFT Options – Selecting this button will open the FFT Options dialog.

2.8.2 Frequency Domain Plots

The frequency domain data plots can be zoomed and panned like the time domain data in the Scope panel of the main application window.

Reset Plot Scaling – Selecting this button will reset the horizontal and vertical scaling of the frequency domain data.

2.8.3 FFT Options Dialog

This dialog contains various options that affect FFT operations. An instance of this dialog is shown in the figure below.

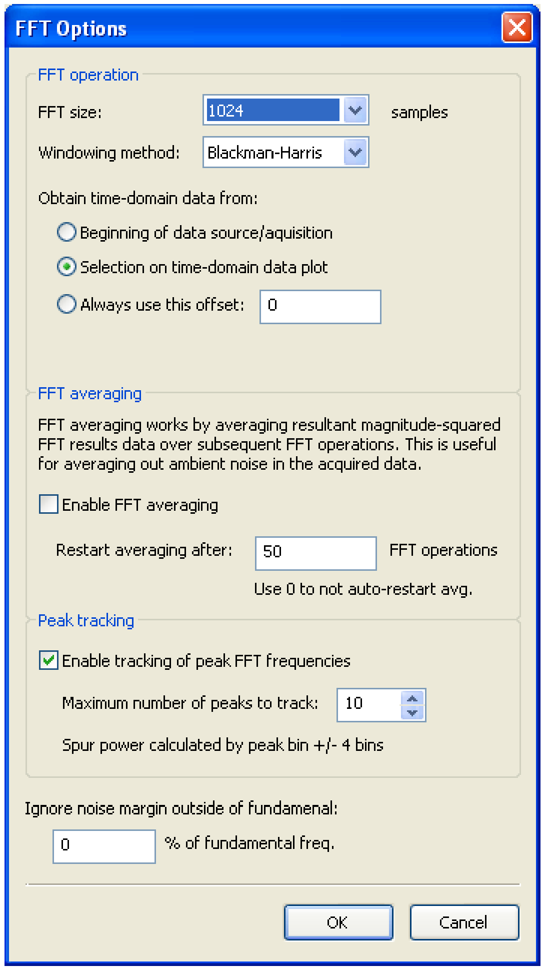

Figure 2-7: FFT Options Dialog

FFT Size – This item controls the size of the FFT operation.

Windowing method – This item defines how the time-domain data will be windowed prior to the FFT operation.

Obtain time-domain data from – This item controls where the source time-domain data will be obtained.

- Beginning of data source/acquisition – Time-domain data is obtained from the start of channel's data. For PDA16 sources this will be the start of the acquisition or recording snapshot. For file sources this will be the start of the time-domain data in the file.
- Selection on time-domain plot – Time-domain data is obtained starting at the location of the current selection in the time-domain data plot in the Scope panel. If no selection has been made, the start of the data source will be used.
- Always use this offset – Time-domain data will be obtained from the specific given offset into the data source.

Enable FFT averaging – When this item is checked, the PDA16 Scope Application will average and track the resultant magnitude-squared FFT results. As FFT operations are performed, this will average out any ambient noise in the frequency-domain.

When enabled, this will result in a new tab being added to the Frequency Domain (FFT) window and a new plot added.

Restart averaging after – This item specifies the number of FFT operations after which FFT averaging will be reset. If this value is 0 then the average will never reset.

Enable tracking of peak FFT frequencies – When enabled, the PDA16 Scope Application will track a number of the highest energy spurs. These peaks are displayed in the 'Peak Frequencies' list of each channel's tab. The number of peaks to track is defined by the adjacent edit control.

3 APPENDIX A – REVISION HISTORY

Revision 1.2

- This appendix added
- Added documentation for Startup Files
- Added documentation for Keyboard controls in scope

Revision 1.3

- Updated manual cover page for DynamicSignals contact information