# Total Phase Data Center Software

TOTAL PHASE

PHASE

Supported products:

Data Center Software
Manual v6.00
December 16, 2011

## Features

- Non-intrusive Super-speed USB Monitoring (5 Gbps)
- Non-intrusive High-speed USB Monitoring (480 Mbps)
- Non-intrusive Full-speed USB Monitoring (12 Mbps)
- Non-intrusive Low-speed USB Monitoring (1.5 Mbps)
- Non-intrusive $I^2C$ Monitoring (up to 4 MHz)
- Non-intrusive SPI Monitoring (up to 24 MHz)
- Monitor transmissions in real time as they appear on the bus
- Basic and Complex Matching Trigger and Filter System
- Extensive real-time filters
- Repetitive packet compression
- Bit-level timing from 100 to 20 ns resolution
- Windows, Linux, and Mac OS X compatible


## Summary

The Total Phase Data Center™ software package is a graphical user interface to the Beagle series of protocol analyzers. The Data Center application provides access to all the features of the Beagle analyzers. Developers can capture, display and filter USB, $I^2C$, and SPI serial bus data in real time.

## Contents

## List of Figures

## List of Tables

# 1 Overview

The Total Phase Data Center application is a graphical user interface for the Beagle series of protocol analyzers. Data Center software provides access to all the features of the Beagle analyzers. Developers can capture, display and filter USB, $I^2C$, and SPI serial bus data in real-time.

This software manual will introduce and explain how to use the Data Center application. For specific or technical information about the Beagle protocol analyzers, please refer to the Beagle Protocol Analyzer Data Sheet which can be downloaded from the Total Phase website - www.totalphase.com.

## 1.1 Changes in version 6.00

**SPI**

- Aardvark XML export bug fix.

**CAN**

- Added full CAN/Komodo support.

## 1.2 Changes in version 5.20

**General**

- Added timestamp LiveFilter.

**$I^2C$**

- Added Aardvark XML export.

**SPI**

- Added Aardvark XML export.

**USB**

- Added USB 3.0 packet truncation.
- Extended USB 3.0 complex match system.
- Extended USB 3.0 error detection to include framing errors and EDB.

## 1.3 Changes in version 5.14

**General**

- Added search history to LiveSearch.
- Added Recent Files to the File menu.

**USB**

- Added an option to group bulk endpoint packets.
- Minor bug fixes.

## 1.4 Changes in version 5.13

**General**

- Improved LiveFilter when filtering for a particular device.
- Fixed some instabilities.

## 1.5 Changes in version 5.12

**General**

- Fixed some instabilities.

## 1.6 Changes in version 5.11

**General**

- Minor bug fixes.

**I$^2$C/SPI**

- Added real-time I$^2$C and SPI statistics.

## 1.7 Changes in version 5.10

**General**

- Minor bug fixes.

**USB**

- Added LTSSM View.
- Improved video class decoding.

## 1.8 Changes in version 5.03

**General**

- Minor bug fixes.

**USB**

- Added a firmware update notifier for the Beagle USB 5000 SuperSpeed Protocol Analyzer.
- Expanded real-time USB3.0 statistics.
- Added quick filters in the statistics pane.

## 1.9 Changes in version 5.02

In addition to the features listed below, the Total Phase Data Center software is now available as a 64-bit application for Mac OS X.

**General**

- Added ability to goto a specific record index.
- Added option to auto connect when a single analyzer is attached.
- Minor bug fixes.

**USB**

- Improved managed configurations functionality.

## 1.10 Changes in version 5.01

**General**

- Fixed an issue related to loading/saving large captures.
- Minor bug fixes.

**USB**

- Improved functionality of the statistics pane.

## 1.11 Changes in version 5.00

**General**

- Added the Capture Control window.
- Expanded timestamp reporting capabilities.
- Improved filtering interface for faster results.
- Minor bug fixes.

**USB**

- Added support for the Beagle USB 5000 Protocol Analyzer.
- Added ability to save, edit, and apply device configurations.
- Added real-time packet, transfer, and error statistics.

## 1.12 Changes in version 4.30

In addition to the new features listed below, the Total Phase Data Center software is now available as a 64-bit application for Linux and Windows, which allows for a larger capture buffer and more records to be captured.

**General**

- Added Block View.
- Added ability to export record data to a binary file.

**USB**

- Added time-ordered packet view.
- Added option to keep individual data-less sequences.
- Improved USB Video Class (UVC) support:
  - Added decoding of UVC headers.
  - Added grouping of video frames.

- The following classes have been added:
  - Audio 2.0
  - USB Attached SCSI (UAS)

## 1.13 Changes in version 4.20

The following features have been added to the Total Phase Data Center software.

**General**

- Added prompt configuration to preferences.
- Added quick filters to record window context menu.
- Added option to mirror column layout on CSV export.
- Added option to start in minimized state.
- Improved the capture limit by basing the limit on virtual and physical memory.
- Minor bug fixes.

## 1.14 Changes in version 4.10

The following features have been added to the Total Phase Data Center software.

**General**

- Added the ability to configure the timestamp precision of the Transaction Window.
- Added support for Japanese.
- Minor bug fixes.

**USB**

- The following Communications Device Class (CDC) subclasses have been added:
  - Network Control Model (NCM)
  - Mobile Direct Line Model (MDLM) Semantic Models:
    - Early Network Control Model (ENCM)

## 1.15 Changes in version 4.00

The following features have been added to the Total Phase Data Center software.

**General**

- Added the ability to insert comments.
- Added the ability to be notified when updates to the software are available.
- Reinstated the instantaneous and inter-transaction bandwidth reporting.
- Removed the 4 billion record limit.
- Added support for German.
- Minor bug fixes.

**USB**

- The following classes have been added:
  - Printer
  - Video

   - Still Image
   - Communications Device Class (CDC)
     * Public Switched Telephone Network (PSTN)
     * Ethernet Control Module (ECM)
     * Wireless Mobile Communications Devices (WMC)
   - Device Firmware Upgrade (DFU)

## 1.16 Changes in version 3.50

The following new features have been added to the Total Phase Data Center software.

**General**

- Added support for French.
- Minor bug fixes.

**USB**

- Added support for class-level decoding. The following classes are supported:
  - Audio v1.00
  - Hub
  - Human Interface Devices (HID)
  - Mass Storage (Bulk Only)
  - Standard Device Requests

## 1.17 Changes in version 3.10

The following new features have been added to the Total Phase Data Center software.

**General**

- Added support for circular capture buffer.
- Added support for exporting in CSV format.
- Added support for additional languages.
- Minor bug fixes.

## 1.18 Changes in version 3.00

The Total Phase Data Center application has been migrated to a new platform. Many features have been added to make the interface more informative and easier to navigate.

**Notable New Features**

- USB packets are now grouped into transactions in a hierarchical view in real-time.
- The Info pane displays detailed information about the selected record, including parsed packet information displayed in multiple formats.
- The Bus pane lists devices that have been detected on the bus being monitored. For USB captures, any enumeration information that was seen during the capture is also represented in detail.

- A command line console, with built-in history, logs all actions performed in the software. The user can then repeat or modify previous actions and create batch scripts that are loaded on startup.

MDIO support is no longer available in version 3.00 of the Data Center software. Additionally, a few features from the previous version of the Data Center application will be temporarily unavailable as they are migrated to the new platform. If you require these features, download version 2.20 of the Data Center application from the Total Phase website, www.totalphase.com.

## 1.19 Changes in version 2.20

The following new features have been added to Total Phase Data Center software.

**General**

- Added support for 64-bit Windows. Note that Data Center will run on 64-bit systems as a 32-bit application.

## 1.20 Changes in version 2.19

The following new features have been added to Total Phase Data Center software.

**General**

- Added support for Intel versions of Mac OS X 10.4 Tiger and 10.5 Leopard.

**USB**

- Made improvements to OTG event detection.
- Resume bus events are now displayed with a duration.
- Fixed the timing of high-speed suspend events.
- Fixed a bug with the Setup Data Details.
- Minor bug fixes.

## 1.21 Changes in version 2.10

The following new features have been added to Total Phase Data Center software.

**General**

- Added the ability to insert comments during a capture.
- Native file format (.bgl) updated to store comments. Old files can be opened but new files will not open in previous versions of the software.
- New bandwidth and data payload measurement between packets.
- Instantaneous bandwidth displayed during capture.
- On Windows systems, the saturation dialog now stays accessible even if application is minimized and later unminimized.
- Fixed an issue with save and export dialogs affecting certain Linux systems.

**$I^2C$**

- A parsing issue with 10-bit $I^2C$ transactions has been resolved.

**USB**

- Beagle USB 12 Analyzer now displays bit-stuff errors.
- USB descriptors are now properly parsed when requested by the OS X operating system.
- Improved display of Setup packets.
- BOS and OTG descriptors are now parsed
- Beagle USB 480 Analyzer reports OTG events

## 1.22 Changes in version 2.00

The following new features have been added to Total Phase Data Center software.

**General**

- Delta time displayed between the selected transaction and the transaction under the mouse pointer.
- **Find** tool can search for text in info, event, and collapsed transactions.
- Option to increase incoming data buffer size to reduce possibility of losing capture data.
- Beagle analyzer hardware and firmware version displayed in status bar.
- Native file format (.bgl) updated to store additional information. Old files can be opened but new files will not open in previous versions of the software.

**I$^2$C**

- Added parsing support for I$^2$C combined format transactions.
- XML batch support export option added to replay I$^2$C transactions using an Aardvark I$^2$C/SPI Host Adapter and the Aardvark Control Center software.

**SPI**

- XML batch support export option added to replay SPI transactions using an Aardvark I$^2$C/SPI Host Adapter and the Aardvark Control Center software.

**USB**

- CRC and SOF frame number errors reported in error column.
- Find can search for specific PIDs by name.
- USB packet group filtering option layout improved.
- Filter option added to filter out event transactions.
- Details mode replaces Descriptor mode to display information about collapsed packets and events, in addition to the descriptor display.
- Support for Beagle USB 480 protocol analyzer added with the following features:
  - Real-time capture of High-speed, Full-speed, and Low-speed USB traffic.
  - Displays the chirp events in the High-speed negotiation process, including identification of potential "Tiny J/K" events.
  - Option to lock capture speed or to allow the Beagle USB 480 Protocol Analyzer to automatically detect the bus speed.
  - Capture USB traffic in real-time with or without hardware buffer overflow protection, or run a delayed-download capture to minimize traffic on the bus while capture is occurring.
  - Option to suppress common packet groups, such as SOF, IN+NAK, and PING+NAK in the hardware.

  - Support for up to 4 digital input lines in the capture.
  - Configure the 4 digital output lines to toggle on capture start, active packet, on specific PIDs and device and endpoint address, and on specific data patterns.
  - Ability to disable packet collapsing.
  - Option to filter out packets generated by the Beagle USB 480 Protocol Analyzer.

## 1.23 Changes in version 1.30

The following new features have been added to Total Phase Data Center software.

**General**

- On Windows systems, requires version 1.1.0.0 of the Beagle analyzer USB driver. Refer to the Beagle analyzer datasheet for instructions on upgrading the USB driver.
- Native file format (.bgl) updated to include status information. Old files can be opened but new files will not open in previous versions of the software.
- Abridged transaction timestamp is displayed in transaction tables.
- Abridged transaction duration is displayed in transaction tables.
- Full transaction timestamps and full duration times displayed in timing detailed views.
- Filtering based on transaction duration added.
- Partial byte information and other errors displayed in transaction windows.
- Clear command asks for confirmation of command.
- Progress bar appears when loading .bgl files.
- Saturation dialog in Windows platforms always stays on top of main window.
- Export while in protocol specific tab now only exports transactions of that protocol.
- Informational headers added to text and CSV export formats.
- CSV formats updated with new columns in transaction tables.

**I$^2$C**

- Start and stop conditions displayed in the transaction table.
- Packet detailed view added with start and stop conditions.
- Asterisk will be displayed in the address field of the transaction window if the address byte was NACK'ed.

**SPI**

- Combined MOSI/MISO detailed view added.
- Text export now displays MOSI data properly in timing table.

**MDIO**

- MDIO capture and filtering support added.

**USB**

- Parsing of setup packets improved.

## 1.24 Changes in version 1.20

The following new features have been added to Total Phase Data Center software.

#### General

- Clear added to the menu to clear the capture contents while preserving the file path.
- Data pattern filter added to allow for filtering of transactions based upon transaction data content.
- Open, Save, and Export dialogs remember the directory of the last file worked upon.

#### I²C

- Target power is no longer enabled upon connection to a Beagle I²C/SPI/MDIO analyzer.
- Ability to filter on more than one device address.

#### USB

- Ability to filter on more than one device address.
- Ability to filter on more than one endpoint address.

### 1.25 Changes in version 1.10

The following new features have been added to Total Phase Data Center software.

#### Bug Fixes

- Libpango has been updated and fixed for International versions of Windows.
- Selected index is preserved when applying new filter settings.
- Empty USB configuration and empty USB interface descriptors are properly parsed.
- Consistent time scale is used across all bytes of a given transaction.

#### General

- Maximum and minimum index numbers can be set in the filters to see a particular range of transactions.
- Filtering by length no longer requires a default value and can be left blank.
- Find arbitrary hex value or ASCII data patterns in a transaction. Matching data patterns are highlights in the Data view.
- Application saturation dialog window now includes a bar graph that indicates the application load.
- The Connection Dialog window can be refreshed by clicking the refresh button.
- Beagle analyzers that are in use by other applications are listed as unavailable in the Connection Dialog window.
- When exporting "-export" is added to exported file name by default.
- Comma separated values (*.csv) file added as an export option.

#### I²C

- Filter transactions based on the read/write bit.

#### USB

- Grouped IN, OUT and SETUP packets are color coded.
- Added a CRC column in the transaction window. The CRC can be viewed in standard bit order or in reversed bit order (for CATC compatibility).

# 2 Quick Start

## 2.1 Capturing traffic

The general flow for capturing traffic is the same with any Beagle analyzer (whether it be the USB 5000, USB 480, USB 12, or I$^2$C/SPI analyzer) or Komodo interface with the following caveats.

When monitoring USB, it is best to attach the Beagle analyzer's analysis port and start the capture before the attaching target device port. This allows the Beagle Analyzer to capture the descriptor information that is communicated at device connection.

For all other protocols, the capture can be started before or during the presence of traffic on the target bus. If traffic is already present on the bus, the first packet may appear corrupted or incomplete since the Beagle analyzer or Komodo interface may start monitoring traffic midway into a transmission.

These are the basic steps for capturing data with a Beagle analyzer or Komodo interface. For more detailed information, please refer to the specific sections in this manual.

1. Install the Total Phase USB driver and Data Center software.
2. Plug in the Beagle analyzer or Komodo interface into the analysis computer.
3. Attach the Beagle analyzer or Komodo interface to the bus under test. (For USB, do not attach the target device until the capture been started. The target host port, however, can be connected now.)
4. Launch the Data Center application.
5. Click the (**Connect to Analyzer**) button in the toolbar and connect to an available analyzer or interface.
6. Click on the (**Device Settings**) button in the toolbar and make sure the correct capture protocol is selected in the pull-down list. Set any other device settings as appropriate.
7. Click on the (**Capture Settings**) button in the toolbar. Set the capture settings as appropriate.
8. Ensure the Protocol Lens is set to the appropriate protocol (CAN, I$^2$C, SPI, or USB).
9. Click the (**Run Capture**) button in the toolbar to start the capture.
10. For USB, connect the target device to the Beagle USB 12/480/5000 analyzer. Some spurious events may appear. There is no need to be alarmed since these events correspond to electrical noise created during the physical connection event.
11. As traffic is seen on the bus, it will be displayed in real-time in the Transaction window.

The capture can be stopped at any time by clicking the  (**Stop**) button. The captured data can be filtered during or after the capture. The captured data or a filtered view can be saved as a binary *.tdc file for future analysis.

# 3 Getting Started

## 3.1 Requirements

#### Overview

The following sections describe the minimum system requirements to run the Data Center software. Be sure the device driver has been installed before plugging in the Beagle analyzer or Komodo interface. Refer to the Beagle Protocol Analyzer Datasheet for Komodo CAN Duo Interface Datasheet additional information regarding the driver and compatibility.

#### Hardware

- Intel or AMD processor running at a minimum speed of 2.0 GHz
- 512 MB of physical RAM, (2 GB of physical RAM is recommended for USB 3.0 captures)
- 1 GB of hard disk space
- High-speed USB port

#### Windows

Data Center software is compatible with Windows XP (SP2 or later, 32-bit and 64-bit), Windows Vista (32-bit and 64-bit), and Windows 7 (32-bit and 64-bit). Windows 2000 and legacy 16-bit Windows 95/98/ME operating systems are not supported. The software is provided as a 32-bit or 64-bit application.

#### Linux

Data Center software has been designed for Red Hat Enterprise Linux 5 with integrated USB support. Kernel 2.6 is required. The software is provided as a 32-bit or 64-bit application.

#### Mac OS X

Data Center software is compatible with Intel versions of Mac OS X 10.5 Leopard, 10.6 Snow Leopard, and 10.7 Lion. The software is provided as a 32-bit or 64-bit application.

## 3.2 USB Driver

Please refer to the Beagle analyzer datasheet or Komodo interface datasheet for instructions regarding installing and uninstalling the Beagle analyzer USB driver.

## 3.3 Installing Data Center Software

The Data Center software package is a self-contained application. All DLLs and support files that are required to run the Data Center software are bundled into a single directory hierarchy. No additional DLLs need to be installed into the core operating system directories (e.g. `c:\Windows\`).

This makes installing the software as easy as unarchiving the software zip package into the directory of your choice.

To install the Data Center application:

1. Download the latest version of the software from the Total Phase website.
2. Unzip the zip archive to your desired location.

   Please make sure that the directory structure is preserved when unzipping the zip archive. The application will fail to launch if the directory structure is not preserved.

## 3.4 Uninstalling Data Center Software

Since the Data Center application is self-contained, there is no need to "Uninstall" it. To remove the application from your machine, you need only delete the directory where the application resides. There is no further action required to remove the software from the system.

## 3.5 Overview of the Beagle Protocol Analyzers

This is a brief introduction to the Beagle Protocol Analyzers. More detailed information can be found in the Beagle Protocol Analyzer Datasheet.

### Beagle USB 5000 SuperSpeed Protocol Analyzer

The Beagle USB 5000 SuperSpeed Protocol Analyzer is a high-performance analyzer for monitoring super-, high-, full-, and low-speed USB traffic.

Please note the following performance issues.

- The maximum cable length for USB 3.0 is not specified and is dependent on gauge of the wires used and the overall quality of the cable. Shorter cables with a wider gauge are better. For this reason, it is strongly recommended that short cables are used to ensure good signal integrity between the target host and the target device.
- Given the speeds of USB 3.0, it is not possible to passively monitor the USB 3.0 bus. Consequently, the USB 3.0 data stream needs to be regenerated to send to the target receiver. It is important to note that the latency of this regeneration is only 1ns and that the USB 3.0 signal is not retimed.

#### Front Panel

On the front of the Beagle USB 5000 analyzer (Figure 1) are a number of ports and LED indicators.

#### Analyzer Power

The Beagle USB 5000 analyzer power indicator is integrated into the Total Phase logo. When the analyzer is powered, the circles in the Total Phase logo will be illuminated.

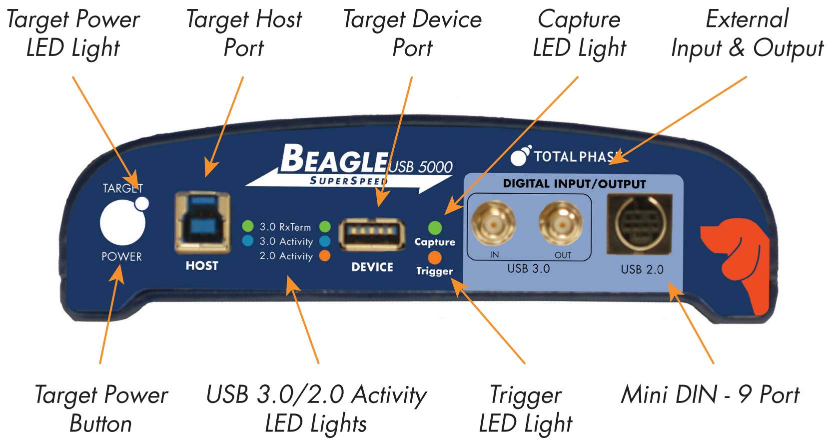



***Figure 1:*** *Beagle USB 5000 Protocol Analyzer - Front Panel*

**Target Power**

The Target Power indicator consists of two elements: the large white circle is a button with the LED indicator in the upper right corner. When $V_{DD}$ is active, the white LED will be on. $V_{DD}$ can be disconnected between the target host and target device by pressing and holding the Target Power button.

**Target Host and Target Device**

The Target Host port is a SuperSpeed USB B receptacle and the Target Device port is a SuperSpeed USB A receptacle. Both receptacles can accept either a USB 2.0 or USB 3.0 cable. To capture USB 3.0 data, a USB 3.0 cable must be used.

**Activity Indicators**

**USB 3.0 RxTerm**

The RxTerm LEDs are illuminated when the proper USB 3.0 receiver termination is detected on the respective link. The left LED corresponds to the upstream link into the host and the right LED corresponds to the downstream link into the device.

**USB 3.0 Activity**

The USB 3.0 Activity LEDs are illuminated when there is USB 3.0 bus activity and a data capture is active. The LED blink speed is proportional to the amount of USB 3.0 traffic on the bus. If the analyzer is not capturing data, the LEDs will not be active even if there is USB 3.0 traffic on the bus. The left LED corresponds to the upstream activity and the right LED corresponds to the downstream activity.

**USB 2.0 Activity**

The USB 2.0 Activity LED is illuminated when there is USB 2.0 bus activity and the data capture is active. The LED blink speed is proportional to the amount of USB 2.0 traffic on the bus. If the analyzer is not capturing data, the LED will not be active even if there is USB 2.0 traffic on the bus.

**Capture**

The Capture LED indicator will be illuminated when a capture is active. Once the capture has ended, the Capture indicator will continue to blink while data is being transferred to the analysis computer. The Capture LED will turn off once the data transfer is complete.

#### Trigger

The Trigger LED indicator will be illuminated once the trigger occurs. The indicator will remain active until all the data has been downloaded to the analysis PC.

#### External Inputs and Outputs

The Beagle USB 5000 analyzer features two separate sets of external inputs and outputs.

#### USB 3.0 Digital Input and Output

The USB 3.0 Digital Input and Output are the two SMA connectors located on the front panels.

**WARNING: The USB 3.0 Digital Input and Output are only rated for 1.8V.** The USB 3.0 input and output of the Beagle USB 5000 analyzer have been optimized for maximum edge performance at 125 MHz. **Applying signals with higher voltage will damage your analyzer and void the warranty.**

#### USB 2.0 Digital Input and Output

The USB 2.0 Digital Inputs and Outputs are available through the Mini-DIN9 port.

#### Back Panel

On the back of the Beagle USB 5000 analyzer (Figure 2) is the power switch, power receptacle, and downlink port.

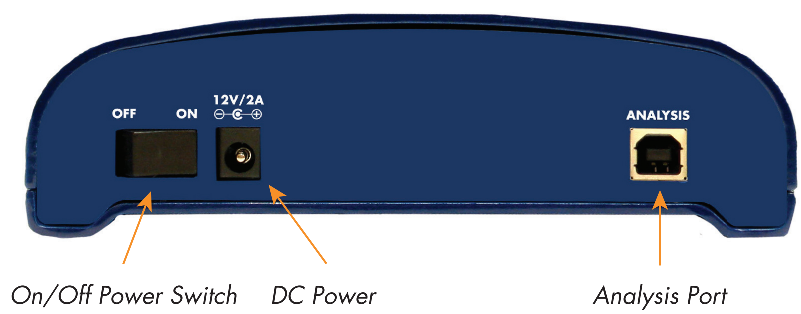



***Figure 2:*** *Beagle USB 5000 Protocol Analyzer - Back Panel*

#### Analysis

The Analysis port is a high-speed USB downlink and must be connected with a standard USB 2.0 cable to the Analysis computer running the Data Center Software.

#### Power

The Beagle USB 5000 analyzer includes a 36 W AC power adapter. To ensure proper operation, the Beagle analyzer must be powered on before any devices are connected to the analyzer.

The DC connector has positive-polarity barrel plug with dimensions of 5.5 mm x 3.5 mm x 9.5 mm.

### Beagle USB 480 Protocol Analyzer

The Beagle USB 480 analyzer is a compact device for monitoring high-, full-, and low-speed USB traffic.

On one side of the Beagle USB 480 monitor is a single USB-B receptacle. This is the **Analysis** side (Figure 3). This port connects to the analysis computer that is running the Beagle Data Center application or custom application.

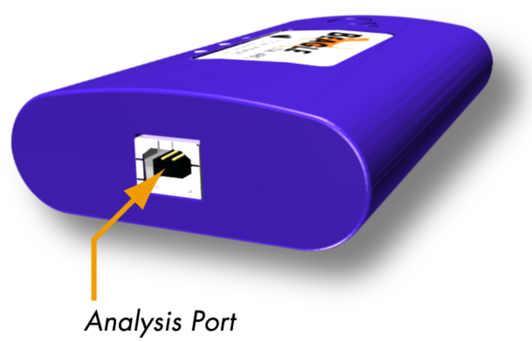



***Figure 3:*** *Beagle USB 480 Protocol Analyzer - Analysis Side*

Please note the following performance issues:

- Use of USB ports that are mounted directly onto the motherboard is highly recommended. Ports that are not mounted directly can cause noise and sync errors due to poor quality of cables and connections.
- For best performance, it is recommended that the Beagle USB 480 analyzer be connected to its own USB host controller. All other USB devices should be connected to separate controllers.
- If only one USB host controller is available, it is still possible to use the Beagle analyzer effectively. Please refer to the Beagle Protocol Analyzer Datasheet (Device Operation section) and later sections of this manual for information on those operating modes.

The opposite side is the **Capture** side (Figure 4), and it contains a USB-A and USB-B receptacle. These are used to connect the target host computer to the target device. The target host computer can be the same computer as the analysis computer. However, for more performance critical applications, separate target host and analysis computers may be necessary.

The **Capture** side acts as a USB pass-through. In order to remain within the USB 2.0 specifications, no more than 5 meters of USB cable should be used in total between the target host computer and the target device.


***Figure 4:*** *Beagle USB 480 Protocol Analyzer - Capture Side*

The **Capture** side also includes a mini-DIN 9 connector which serves as a connection to the digital inputs and outputs. The pinout of the connector is documented in the Beagle protocol analyzer datasheet.

The top of the Beagle USB 480 Protocol Analyzer has three LED indicators as shown in Figure 5. The green LED serves as an Analysis Port connection indicator. The green LED will be illuminated when the Beagle analyzer has been correctly connected to the analysis computer and is receiving power from USB. The amber LED serves as a Target Host connection indicator. The amber LED will be illuminated when the target host computer is connected to the analyzer. Finally, the red LED is an activity LED. Its blink rate is proportional to the amount of data being sent across the monitored bus. If no data is seen on the bus, but the capture is active, the activity LED will simply remain on.

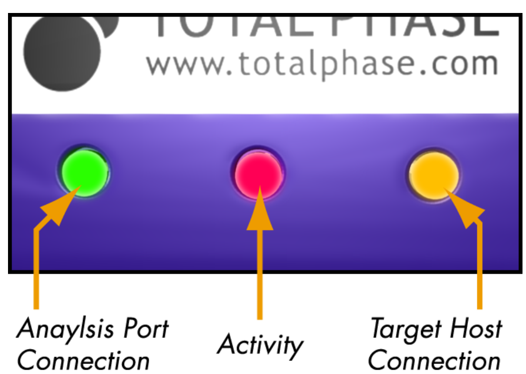



***Figure 5:*** *Beagle USB 480 Protocol Analyzer - LED Indicators*

Please check all the connections if the either of the connection LEDs fail to illuminate after the Beagle USB 480 analyzer has been connected to the analysis computer or the target host computer.

### Beagle USB 12 Protocol Analyzer

The Beagle USB 12 analyzer is a compact device for monitoring full and low-speed USB traffic.

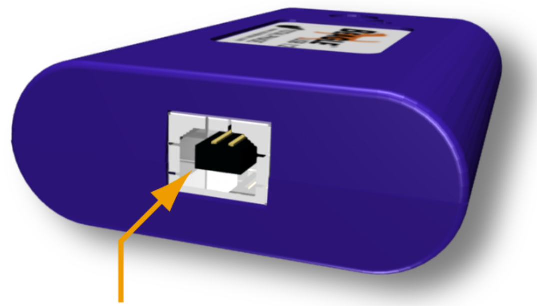



***Figure 6:*** *Beagle USB 12 Protocol Analyzer - Analysis Side*

On one side of the Beagle USB 12 analyzer is a single USB-B receptacle. This is the **Analysis** side (Figure 6). This port connects to the analysis computer that is running the Beagle Data Center application.

Please note the following performance issues:

- Use of USB ports that are mounted directly onto the motherboard is highly recommended. Ports that are not mounted directly can cause noise and sync errors due to poor quality of cables and connections.
- For best performance, it is recommended that the Beagle USB 12 analyzer be connected to its own USB host controller. All other USB devices should be connected to separate controllers.

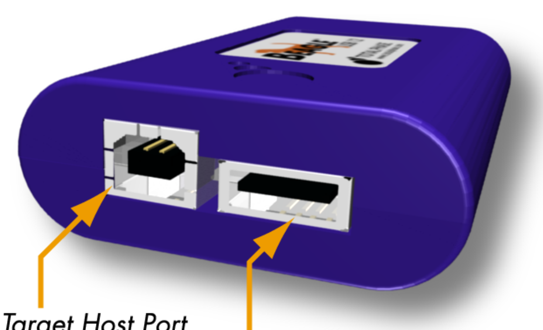



***Figure 7:*** *Beagle USB 12 Protocol Analyzer - Capture Side*

On the opposite side is the **Capture** side (Figure 7), are a USB-A and USB-B receptacle. These are used to connect the target host computer to the target device. The target host computer can be the same computer as the analysis computer, though for more performance critical applications, separate target host and analysis computers may be necessary.

The **Capture** side acts as a USB pass-through. The Beagle USB 12 analyzer is galvanically isolated from the USB bus to ensure the signal integrity. In order to remain within the USB 2.0 specifications, no more than 5 meters of USB cable should be used in total between the target host computer and the target device. For best performance, it is recommended that the absolute minimum amount of cable be used.

Please note, that on the **Capture** side, there is a small gap between the two receptacles. In this gap, two LED indicators are visible, a green one and an amber one, as shown in Figure 8. When the Beagle USB 12 analyzer has been correctly connected to the analysis computer, the green LED will illuminate. When the Beagle USB 12 analyzer is correctly connected to the target host computer, the amber LED will illuminate.

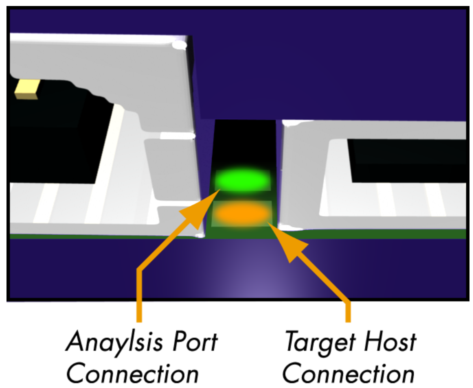



***Figure 8:*** *Beagle USB 12 Protocol Analyzer - LED Indicators*

Please check all the connections if the one or both LEDs fail to illuminate after the Beagle USB 12 analyzer has been connected to the analysis computer or the target host computer.

### Beagle I$^2$C/SPI Protocol Analyzer

The Beagle I$^2$C/SPI analyzer is physically similar to the Aardvark I$^2$C/SPI Host Adapter.

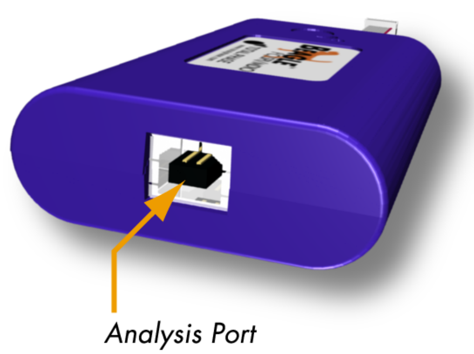



***Figure 9:*** *Beagle I$^2$C/SPI Protocol Analyzer - Analysis Side*

On one side of the Beagle I$^2$C/SPI analyzer is a single USB-B receptacle. This is the **Analysis** side (Figure 6). This port connects to the analysis computer that is running the Beagle Data Center application.

Please note the following performance issue:

- Use of USB ports that are mounted directly onto the motherboard is highly recommended. Ports that are not mounted directly can cause noise and sync errors due to poor quality of cables and connections.

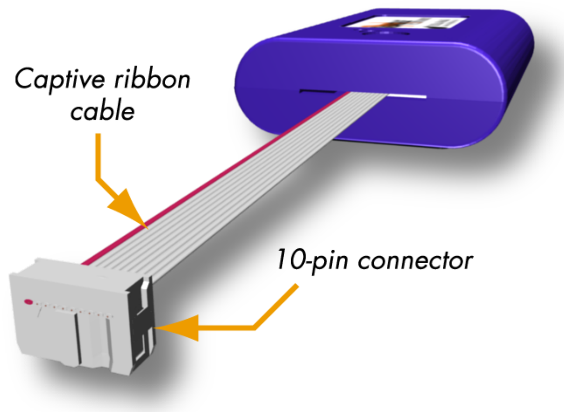



***Figure 10:*** *Beagle $I^2C$/SPI Protocol Analyzer - Capture Side*

On the opposite side is the **Capture** side (Figure 10, is a captive 10-pin ribbon cable. This cable is used to connect to the serial bus. The ribbon cable connector is a standard 0.100" (2.54mm) pitch IDC type connector. This connector will mate with a standard keyed boxed header.

Alternatively, Total Phase sells a 10-pin split cable with and without grabber clips which connects to the Beagle $I^2C$/SPI analyzer and provides individual flying leads for each pin which can be connected to the serial bus.

This 10-pin connector has the same pinout as the Aardvark $I^2C$/SPI Host Adapter. This pinout is documented in the Beagle Protocol Analyzer Datasheet.

## 3.6 Overview of the Komodo CAN Duo Interface

The Komodo CAN Duo Interface is a compact, multifunctional tool for passively monitoring and actively participating on up to two CAN buses simultaneously.

**USB Downlink**

On one end of the Komodo CAN Duo Interface is a single USB-B receptacle. This port connects to the analysis computer that is running the Data Centter application. This port must be plugged in to provide power to the Komodo CAN Duo Interface and to power the CAN bus over V+ (See Section 9.2 to learn how to enable target power).

While the Komodo CAN Duo Interface has a single USB port, it presents two virtual ports to the user. This allows two separate applications to connect simultaneously to a single Komodo CAN

Duo Interface. See Section 4.4 for more information on connecting to a device.

**CAN**

The Komodo CAN Duo Interface features two connectors for each CAN channel: a common DB-9 connector (Figure 11) and a block screw terminal (Figure 12) to which wires can be easily connected.

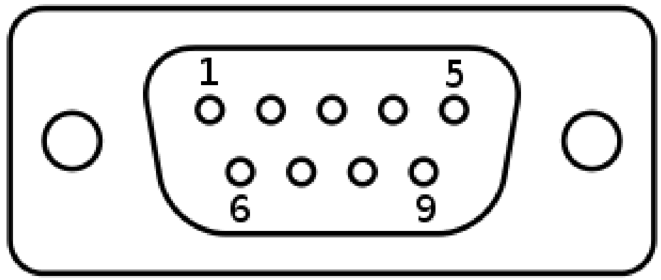



***Figure 11:*** *Komodo CAN Duo Interface - CAN DB-9 Connector*

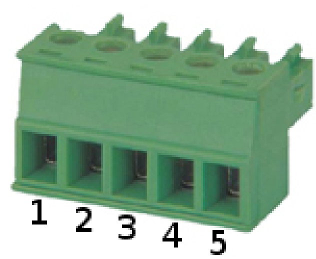



***Figure 12:*** *Komodo CAN Duo Interface - CAN Terminal Block*

**GPIO**

On the end opposite the USB port is a DIN-9 connector (Figure 13) for GPIO use.

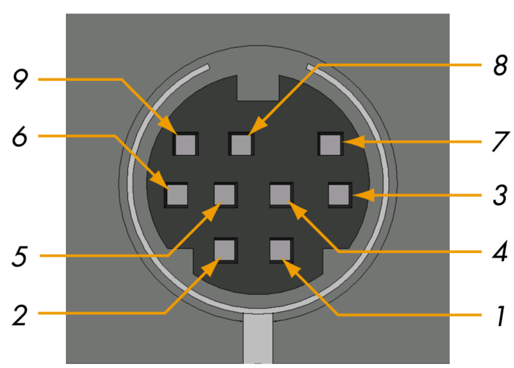



***Figure 13:*** *Komodo CAN Duo Interface - GPIO DIN-9 Connector*

Even though the GPIO DIN-9 cable included with the Komodo interface is labled with 4 inputs and 4 outputs, each GPIO pin can be configured as an input or an ouput. Table 1 shows the pinout for the DIN-9 connector on the Komodo interface along with corresponding color and label on the cable. The GPIO configuration window in Data Center is color-coded to make configuration easier (See Section 9.2 for details).

***Table 1:*** *GPIO Cable Pin Assignments*

| Number | Color | Label |
|---|---|---|
| Pin 1 | Brown | IN 1 |
| Pin 2 | Red | IN 2 |
| Pin 3 | Orange | IN 3 |
| Pin 4 | Yellow | IN 4 |
| Pin 5 | Green | OUT 1 |
| Pin 6 | Blue | OUT 2 |
| Pin 7 | Purple | OUT 3 |
| Pin 8 | Grey | OUT 4 |
| Pin 9 | Black | GND |

# 4 Using the Total Phase Data Center Application

## 4.1 Starting Data Center Software

### Windows

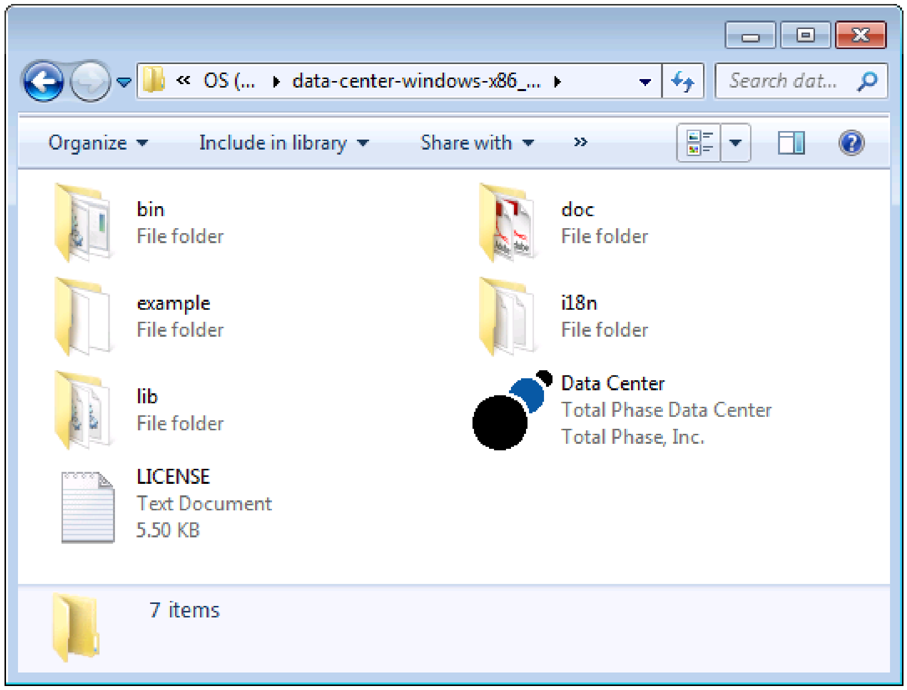



*Figure 14: Total Phase Data Center Software Directory Contents*

1. Go to the folder (Figure 14) where the software package was extracted.
2. Double-click on **Data Center.exe**.

### Linux

1. Go to the installation directory where the software package was unzipped.
2. Execute `Data Center`.

### Mac OS X

1. Go to the installation directory where the software package was unzipped.
2. Double click **Data Center**.

### Windows File Associations

When Total Phase Data Center software is executed on a Windows machine, it will check to see if the correct file associations have been set for Data Center files (*.tdc files). This file association will allow users to double click on TDC files and have them automatically open in Data Center. It will also provide an icon for the TDC files to make them easier to distinguish.

Click **OK** to set the association, or **Cancel** to leave the file type unassociated.

Please note that only users with write permissions to the Windows registry will be able to set the file association for TDC files.

### Command Line Options

To launch the Data Center application from the command line, use the script located in the `bin` directory in the software package. Note that the `bin` directory is located inside the app bundle on Mac OS X.

The following option are available when running the Data Center application from the command line:

- **-b FILE**, Run the given file in batch mode.
- **-c**, Create a command line interface.
- **-m**, **–minimized**, Start in a minimized state.
- **-r PORT**, Create a remote console on the given TCP port.

#### Batch Mode

The **-b FILE** option allows for the specified file to be run in batch mode when the Data Center application is launched. The file can contain commands in the same format as those entered in the command line window (Section 5.3).

#### Command Line Console

Using the **-c** option will create a command line console on the command line where the Data Center application was launched. Commands that can be entered in the command line window (Section 5.3) can also be entered in the console.

#### Minimized State

Using the **-m** or **–minimized** option will launch Data center in minimized mode. This allows users to launch the application, use the command line console, and never see the GUI on screen.

### Remote Console

The **-r PORT** option will create a remote console on the given port. Connecting to this port via Telnet will give the user a command line console similar to the one found in the command line window. This allows users to control the Data Center application when they can't physically be in front of the machine running the application.

Certain commands that require a graphical interface will not be permitted. Additional arguments may be required in order to execute these commands from the remote console. Refer to a command's help output for more details on the required arguments (Section 5.3).

## 4.2 Exiting Data Center Software

To exit the application, select **File | Quit** from the menu or use the keyboard shortcut **Ctrl+Q**.

Upon quitting, Data Center software will verify that the current capture session has been saved. If it has not been saved, the user will be prompted to save or discard the file before exiting (Figure 15).

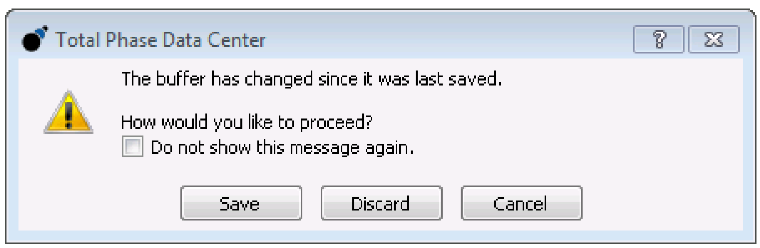



***Figure 15:*** *Unsaved Warning Dialog*

## 4.3 Getting Around the Total Phase Data Center Application

The Data Center application is a powerful, yet easy-to-use, graphical interface to the Beagle analyzers and Komodo interface. The general interface of this application is shown in Figure 16.

**Title Bar** (1)
The title bar provides the status of the current capture file. A name of "Untitled" will be used when a new capture buffer is created and the data has not been saved to a file.

An asterisk (*) preceding the filename indicates that the capture buffer contains new data that has not been saved. Figure 17 shows this situation with a filename of "usb_capture".

A plus (+) preceding the filename indicates that the data in the capture buffer has been saved, but reduced save settings have been used. For additional information on reduced save settings see the Save Settings section below. Figure 18 shows an example of this notation.

**Toolbar** (2)
The toolbar provides single click access to the majority of the Data Center application's functionality.

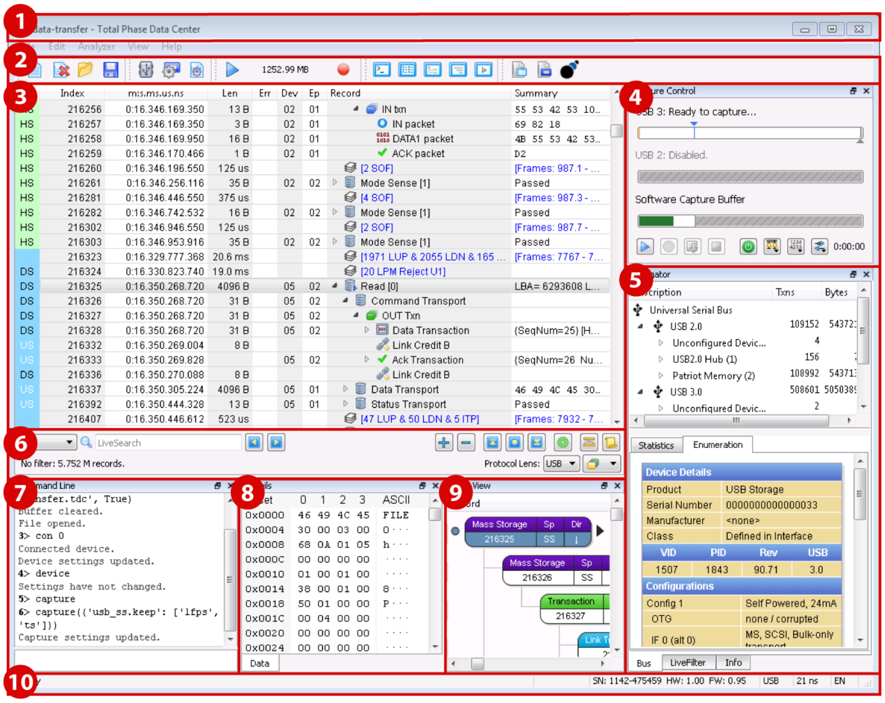



*Figure 16: Beagle Data Center Software Interface*

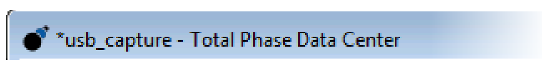



*Figure 17: Unsaved data exists in the capture buffer*

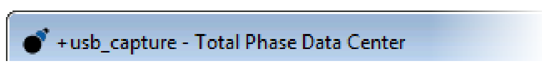



*Figure 18: Capture buffer has been saved with reduced save settings*

**Transaction Window** (3)
The Transaction window displays all the raw data from the bus capture. When capturing packets over the USB protocol, the application will group related packets under a single transaction entry in the Transaction window. The user may expand or collapse the entries in a transaction by double-clicking the record, or by single-clicking the icon preceding the record's description in the **Record** column.

**Capture Control Window** (4)
The Capture Control Window provides an interface to start and stop the capture as well as see the amount of capture buffer being used. When using the Beagle USB 5000 analyzer, there are additional features to manually trigger a capture and control other properties of the capture.

**Navigator Window** (5)
The Navigator window contains three tools that allow the user to quickly find relevant data.

- The **Bus Pane** shows all devices that the Beagle Analyzer has detected on the bus during a capture, as well as the address(es), endpoint(s), enumeration information, and statistics corresponding to that device.
- The **LiveFilter Pane** allows the user to filter the captured data based on selected param-

eters.

- The **Info Pane** gives a detailed description of the information and fields, if any, contained in a packet.

The specific operation of each panel depends on the protocol being analyzed. For more specific information, see the relevant protocol sections below.

**Transaction Window Controls** (6)
These controls allow the user to navigate the Transaction window and alter the way records are displayed.

**Command Line Console** (7)
The command line console provides a command line interface to the application and logs all actions performed. The user can then repeat or modify previous actions and create batch scripts that are loaded on startup. For a list of commands, type **help** into the console. Command-specific documentation can also be accessed using help with the **help COMMAND** syntax.

**Details Window** (8)
The raw bytes from the selected record are displayed in the Details window. Timing information will also be displayed if the protocol supports it. Each protocol type may have a different set of panels that are specific to that protocol.

**Block View** (9)
The Block View provides an alternate representation of the selected record that combines the hierarchical layout of the Transaction window with the detailed information found in the Info Pane.

**Status Bar** (10)
The status bar provides the user with information about the current status of the software and the Beagle analyzer. It displays information from the search, delta-time, data payload, and instantaneous bandwidth functions. The status bar also displays the hardware and firmware versions of the connected Beagle analyzer.

### Toolbar

The Toolbar (Figure 19) is the primary means of operating the Data Center application. It is comprised of the following functions:

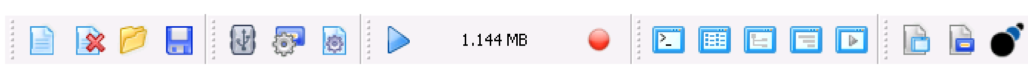



***Figure 19:*** *Beagle Data Center Toolbar*

**File New** 
Use the **File New** button to discard the current capture and create a new, unnamed file.

The File New command can also be issued through **File | New** or with the keyboard shortcut **<Ctrl>+N**.

**File Clear** 
Use the **File Clear** button to discard the current capture and keep the current file active.

The File Clear command can also be issued through **File | Clear** or with the keyboard shortcut **<Ctrl>+L**.

**File Open** 
Use the **File Open** button to open a previously saved capture file.

The File Open command can also be issued through **File | Open** or with the keyboard shortcut **<Ctrl>+O**.

**File Save** 
Use the **File Save** button to save the current capture to disk.

The File Save command can also be issued through **File | Save** or with the keyboard shortcut **<Ctrl>+S**.

**Connect to Analyzer...** 
The **Connect to Analyzer...** button launches the Connection dialog, which is the primary means of connecting and disconnecting Beagle protocol analyzers to the Data Center software.

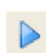
The Connection dialog can also be accessed through **Analyzer | Connect to Analyzer...**.

**Device Settings...** 
The **Device Settings...** button launches the Device Settings dialog, which allows the user to configure device-specific settings.

The Device Settings dialog can also be accessed through **Analyzer | Device Settings...**.

**Capture Settings...** 
The **Capture Settings...** button launches the Capture Settings dialog, which allows the user to configure capture-specific settings.

The Capture Settings dialog can also be accessed through **Analyzer | Capture Settings...**.

**Start Capture/Stop Capture**  
To start a capture, simply press the **Start Capture** button. When a capture is running, the **Start Capture** button becomes the **Stop Capture** button. To stop a capture, press the **Stop Capture** button.

The capture can also be started by selecting the menu item **Analyzer | Run Capture** or using the keyboard shortcut **<Ctrl>+R**. The capture can be stopped by selecting the menu item **Analyzer | Stop Capture** or by using the same keyboard shortcut **<Ctrl>+R**.

**Capture Size**
Indicates the amount of data that has been captured and displays this amount in the appropriate format (either kilobytes or megabytes).

**Capture Indicator** 
Indicates the current state of the capture. For the Beagle I2C/SPI analyzer, the Beagle USB 12 analyzer the Beagle 480 analyzer, and the Komodo interface the indicator has two states. A red indicator means that the capture is currently stopped. A green indicator means that the capture is currently active.

For the Beagle 5000 analyzer, the indicator has three states. A red indicator means that the capture is currently stopped. An orange indicator means that the capture is active and the analyzer is currently capturing pre-trigger data. A green indicator means that the capture is currently active and the analyzer is capturing post-trigger data.

**Command Line**
Toggle the visibility of the Command Line window.

**Details**
Toggle the visibility of the Details window.

**Navigator**
Toggle the visibility of the Navigator window.

**Block View**
Toggle the visibility of the Block View window.

**Capture Control**
Toggle the visibility of the Capture Control window.

**Manual**
Opens a PDF copy of the Data Center Software Manual.

**Datasheet**
Opens a PDF copy of the Beagle Analyzer Datasheet.

**Website**
Launches a web browser and opens the Total Phase website, http://www.totalphase.com.

### Transaction Window

The Transaction window displays the transactions captured on a serial bus in real-time. When a transaction is selected, detailed information about that transaction is displayed in the Details window, the Info pane, and the Bus pane.

The Transaction window has additional protocol specific parsing, providing high level information about the data as is appropriate to the protocol. Specific information about these protocol-dependent features can be found in the sections in this manual pertaining to the respective protocols.

### Transaction Window Controls

These tools allow the user to have more control over the display and navigation of the Transaction window.

**Search Tool**
Run an instantaneous search for text or data across all records. Clicking the magnifying glass

reveals a menu allowing the user to choose what type of data to search for. The **Find Next** and **Find Previous** buttons allow the user to quickly navigate the matching records. The input format required by the search tool is the same as required by the text and data filtering. Please refer to the filtering sections below for additional details.

**Expand/Collapse All** 

The **Expand All** button (plus sign) expands all the transactions that have been captured, while the **Collapse All** button (minus sign) collapses all transactions that have been captured. Please note that the **Incoming Expand/Collapse** button will also be toggled, if needed, to reflect a similar state to either button that was clicked. That is, using the **Expand All** button will also ensure that new transactions are expanded when added to the Transaction Window. Similarly, using the **Collapse All** button will ensure that new transactions are collapsed when added to the Transaction window.

**Buffer Navigation** 

The **Beginning of Buffer** and **End of Buffer** buttons move the scrollbar in the Transaction window to the top or bottom of the capture, respectively. The **Selected Record** button moves the scrolling position of the Transaction window such that the selected record is visible. The **Selected Record** button has no affect if there is no record selected. Please note that any time the scrollbar is moved using any of these buttons, scrolling will be disabled.

**Trigger Navigation** 

The **Find Next Trigger** button moves the scrolling position of the Transaction window such that the next trigger event is visible. The **Find Next Trigger** button will return an error message if there are no trigger events. If there is more than one trigger event, each button press will move the scrolling position to the subsequent trigger event.

**Incoming Expand/Collapse** 

When the **Incoming Expand/Collapse** button is in the expanded state, all new transactions added to the Transaction window are expanded. Similarly, if the button is in the collapsed state, all new transactions are collapsed.

**Scrolling** 

The **Scrolling** button has three states: enabled, locked, and disabled. When enabled, the Transaction window will automatically scroll to the most recently captured transaction. Moving the scrollbar or clicking in the Transaction window will cause the automatic scrolling to be disabled. When the button is in the locked state, the automatic scrolling is enabled, and won't be disabled when clicking in the transaction window or moving the scrollbar. When disabled, the Transaction window will remain at the position the user indicates.

**Filter Status**

When a filter is not applied, the filter status displays the total number of records that have been captured. When a filter is applied, the number of matched records is displayed along with the total number of records.

**Protocol Lens**

The **Protocol Lens** choice box displays the protocol lens that is being applied to the Transaction window. When captured data from multiple protocols are present in the Transaction window, only those transactions from the selected protocol will be shown. Transactions from other protocols will be collapsed to a single record per capture.

Please note that this setting does not affect the capture protocol setting of the current device. To view or change the capture protocol, open the Device Settings dialog.

**Capture View** 
Displays the capture view being applied to the transaction window. **Packet** View displays the capture in a time-ordered fashion. **Transaction** View shows data with protocol-level decoding. **Class** View shows data with class-level decoding. This option is only available for the USB protocol (see section 6.7).

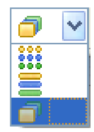


***Figure 20:*** *Capture View Drop Down Menu*

#### Filter Pane of Navigator Window

The Filter pane provides a useful and powerful set of tools to filter the transactions in the transaction window. Filters can be applied at any time, even in real-time during a capture. The use of filters can help developers quickly identify and locate data of interest in a large data set.

#### Details Window

The Details window provides actual byte content of a specific transaction. The Data Center software supports capturing bit-timing for the SPI and I2C protocols. When the capture protocol is configured as I2C and SPI, the Details window will have an extra tab that displays the bit-timing of the transaction selected in the transaction window.

Additional protocol specific viewing modes may be available and are documented in their respective sections.

### 4.4 Connecting to a Beagle Analyzer or Komodo Interface

The Data Center application must connect to a Beagle analyzer or Komodo interface before it can start a capture. To start the connection process, click on the **Connect to Analyzer...** button in the toolbar, or select **Analyzer | Connect to Analyzer ...** from the menu to open the Connection Dialog.

#### Connection Dialog

The Connection Dialog (Figure 21) displays all the devices that are connected to the computer. If the user connects or disconnects a device after the dialog has been opened, click the **Refresh** button in the top right corner to update the list of analyzers in the dialog.

The list of available devices provides the following information:

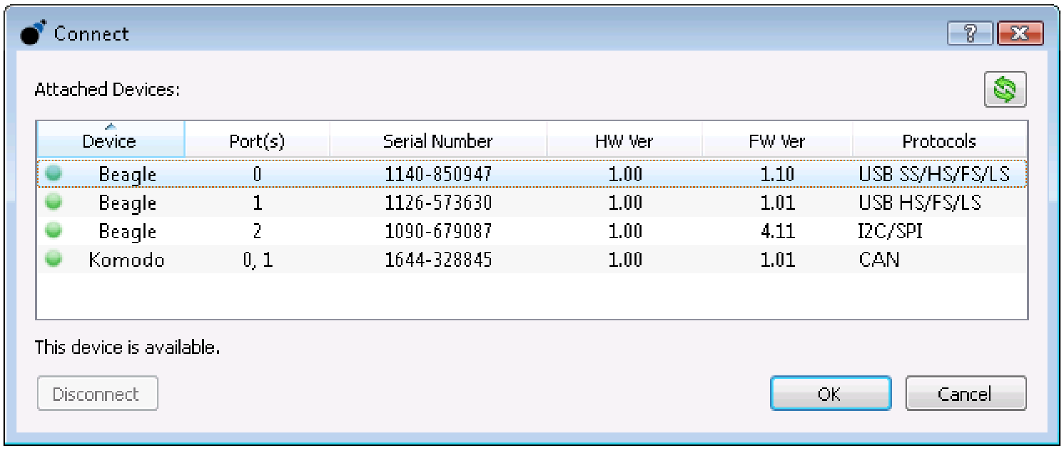



***Figure 21:*** *Connection Dialog*

**Port**
The port number of the Beagle analyzer or Komodo Interface.

Each Komodo device presents two port numbers to the user. This allows the user to connect to the same device from two separate applications. At least one port needs to be available to connect to a Komodo interface from Data Center.

**Availability**
The icon preceding the port number indicates the availability of the device. A green icon indicates an available device, and a red icon indicates a device that is being used by another application. A blue icon indicates the device that is currently connected to this instance of the application.

**Serial Number**
The serial number of the Beagle analyzer. This is a convenience to allow developers to easily identify the physical unit that is being connected to Data Center.

**HW Ver**
The hardware version of the Beagle analyzer.

**FW Ver**
The firmware version of the Beagle analyzer.

**Protocols**
The protocols that can be captured by the device.

### Connecting to a Beagle Analyzer or Komodo Interface

To connect to a Beagle analyzer or Komodo Interface:

1. Click on **Connect to Analyzer...** in the toolbar to open the Connection dialog.
2. Select a Beagle analyzer or Komodo interface from the list of available devices.
3. Click on the **OK** button at the bottom of the dialog.

If the Beagle analyzer is being used by another process, or if the Komodo interface is being used by two other processes, you will not be able to connect to it.

### Disconnecting a Beagle Analyzer or Komodo Interface

To disconnect a Beagle analyzer or Komodo interface:

1. Click on **Connect to Analyzer...** in the toolbar to open the Connection dialog.
2. Click on the **Disconnect** button at the bottom of the dialog.

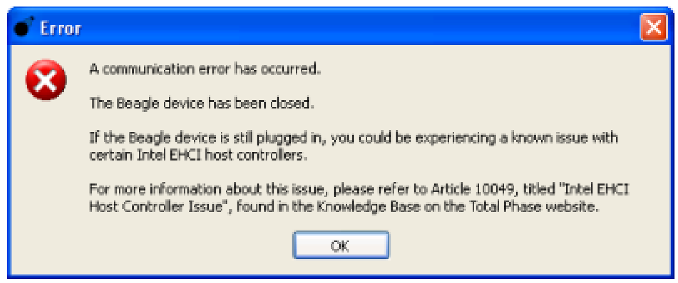



***Figure 22:*** *Beagle Analyzer Connection Error*

Errors can occur if the Beagle analyzer or Komodo interface is physically disconnected before it is disconnected via the software. In these cases, the Data Center application will automatically close the analyzer and display an error message (Figure 22).

## 4.5 Starting a Capture

The application must be connected to a Beagle analyzer or Komodo interface in order to start a capture. If an analyzer or interface is not connected, any attempts to configure a device or run a capture will not succeed.

To start a capture:

1. Connect to a Beagle analyzer or Komodo interface.
2. Click on the **Run Capture** button in the toolbar, or click on the **Run Capture** button in the Capture Control window, select **Analyzer | Run Capture** from the menu, or use the keyboard shortcut **Ctrl+R**.

Once the capture has been started, the capture indicator will flash green and a record indicating the capture start time will appear in the transaction window. When using the Beagle 5000 analyzer, the indicator will initially be orange after starting a capture and no records will be displayed until the trigger has occurred.

While the application is capturing data, it is not possible to reconfigure the device, change the capture settings, or connect to a different Beagle analyzer. To access these options, you must first stop the capture.

Th e current capture data will be appended to any data that has already been captured. Data Center software can only capture to a single file at a time. To capture to a new file, go to **File | New**, or use the keyboard shortcut **Ctrl+N**. If the current data is unsaved, the application will issue a warning (Figure 15). The user has the option to save the data before continuing.

To clear the capture data while keeping the current file open, go to **File | Clear**, or use the keyboard shortcut **Ctrl+L**. If you are discarding data that has not yet been saved, Data Center provides a warning (Figure 23). Any subsequent data captured will be saved to the same file.

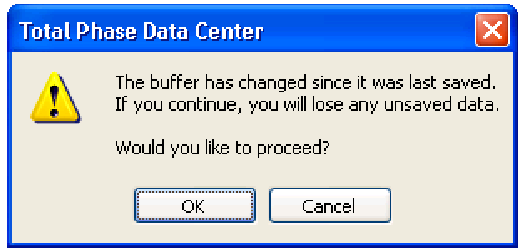



***Figure 23:*** *Clear Capture Data Warning Dialog*

#### Maximum Capture Size

The data captured by Data Center software is stored in memory. The total amount of memory used by the capture is displayed in the toolbar. The **Software Capture Buffer** progress bar in the Capture Control window displays graphically how much of the available memory has been used.

The Data Center application will automatically stop the capture after it has captured a finite amount of data. The stopping point is defined by a user configurable capture limit setting. When the limit is reached, a Capture Limit Dialog (Figure 24) will appear.

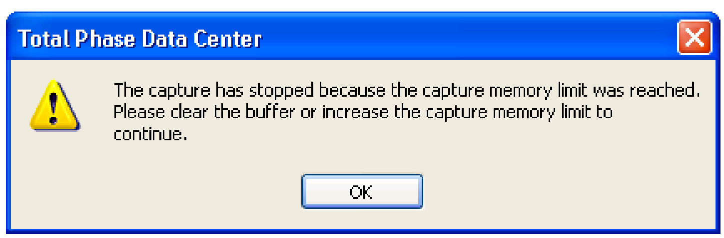



***Figure 24:*** *Capture Limit Dialog*

For information regarding changing the capture limit, please refer to the Changing Settings section (4.14).

### 4.6 Triggering a Capture

After the capture has been started, the Beagle USB 5000 analyzer has the ability to trigger the capture when certain events occur on the bus. Data is only stored on the analyzer until the capture is triggered and then it is downloaded to the analysis machine. Refer to the Device Settings section (6.3) and the Capture Control section (6.5) for more details.

The other Beagle analyzers trigger immediately when the capture is started.

## 4.7 Stopping a Capture

To stop a data capture: click the **Stop Capture** button in the toolbar, click the **Stop Capture** button in the Capture Control window, go to **Analyzer | Stop Capture**, or use the keyboard shortcut **Ctrl+R**.

The capture indicator will turn red and a record indicating the capture stop time will be inserted into the transaction window.

## 4.8 Filtering a Capture

A capture can be filtered at any point during or after a capture.

### Applying Filters

Filters are constructed and applied to the capture through the LiveFilter tab of the Navigator pane. Click on the **LiveFilter** tab at the bottom of the Navigator pane to reveal the LiveFilter options. To apply a filter, click on the **Apply** button in the LiveFilter tab. The results will be immediately displayed in the transaction window.

All filter parameters are applied at the same time. A transaction must meet all the filter requirements in order to appear in the transaction window.

Specific protocols may have additional filtering options available. Information about these options can be found in their respective sections.

#### Instant Filters

Once filters are enabled, it is possible to instantly apply filters to provide immediate feedback to ensure that the correct filter parameters have been set. For data captures larger than 1 GB, there may be a small delay when applying a filter.

Instant filters are controlled by the instant filter toggle button next to the **Apply** button. To activate or deactivate this feature, click on the .

#### Enabling/Disabling Filters

If a filter is enabled, the enabled button at the bottom of the LiveFilter tab will be active. To enable/disable the filter, click on the button. Any hidden transactions will immediately become visible.

#### Editing Filters

You may edit filter settings without applying them by editing the fields of the LiveFilter tab. Clicking the **Revert** button will update the LiveFilter fields with the options from the last filter that was applied, regardless of whether it is currently enabled or disabled.

#### Restoring LiveFilter Defaults

The default filter state matches all packets and does not filter any data from the capture. To restore the default filter state to the LiveFilter tab, click the **Default** button at the bottom of the LiveFilter tab window. Note that the **Default** button only affects the state of the LiveFilter pane and will not apply any settings to the filter.

## 4.9 Searching a Capture

A capture can be searched for arbitrary patterns in the text fields of the Transaction window and in the data payload of each record. The user can choose which fields to search with the drop down menu to the left of the LiveSearch box (Figure 25).

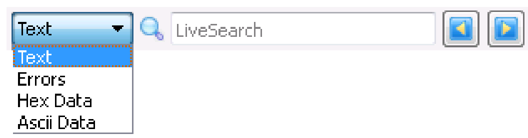



***Figure 25:*** *Search options*

- **Text**
  Search the textual information in the Record and Data columns. Any raw data that is shown in the Data column will not be searched when **Text** is selected. Use **Hex Data** or **ASCII Data** to search the data values.

- **Errors**
  Search the error codes in the Error column.

- **Hex Data**
  Search the payload data of each record and transaction for the hexadecimal data specified in the search pattern. The requirements for the search pattern format are the same as the requirements for the data fields in the filter options. Refer to the filtering sections below for more information.

- **ASCII Data**
  Search the payload data of each record and transaction for the ASCII data specified in the search pattern. As with the **Hex Data**, refer to the filtering sections below for more information regarding the search pattern format.

## 4.10 Saving a Capture

Captures can be saved to a binary file for later analysis. By default, all the data that was captured will be saved to file, regardless of how the data is being filtered. This is to ensure that no information is lost. However, the save settings can be modified to save only the filtered view. See the **Save Settings** section for more details.

To save a capture, go to **File | Save**, or use the keyboard shortcut **Ctrl+S**. If the capture(s) have not yet been saved, the application will open a file save dialog to determine the name and location of the save file.

After the user supplies a valid file location, the application will open another dialog to allow the user to set the file's Save Settings (Figure 26). The Save Settings dialog offers several options for discarding data from the capture in order to reduce the size of the binary file. To preserve all of the captured data, leave all settings unchecked.

The data will be saved into a Total Phase Data Center file (*.tdc).

**Save Settings**

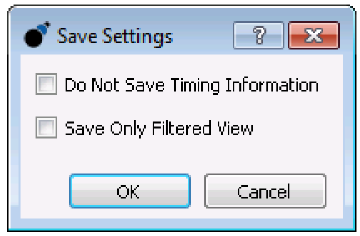



***Figure 26:*** *The Save Settings Dialog*

The save settings must be configured the first time a capture is saved to a file. Excluding data will reduce the size of the saved binary file.

**Do Not Save Timing Information**

In SPI and $I^2C$ capture modes, the Data Center application extracts and displays the timing between bits in a single transaction. While this information can be quite useful, it significantly increases the size of the binary capture file. If this option is selected, none of the bit level timing information will be saved.

**Save Only Filtered View** If selected, only the filtered transactions of the current protocol will be saved. Please note that significant amounts of information may be lost. Because transactions may be missing, the saved file is marked as incomplete. If the user reopens this file with Data Center software, it will be marked as a "**(Filtered File)**" in the title bar and appending additional captures to this file will be disallowed. To resume capturing, the user will need to clear the current capture, create a new file, or open another file.

Also note that any transaction or record that is a soft match (denoted by the faded color) will be saved when saving only the filtered view.

Furthermore, please note that when saving a filtered view of a USB capture, you may lose the ability to do class-level parsing of your capture. This can happen if you filter out certain transactions that are necessary for class-level parsing when in the protocol-level view. If you would like to have the option of class-level parsing for subsequent loading of the saved file, save your filtered view while classification (i.e., class-level parsing) is enabled.

## 4.11 Opening a Saved Capture

To open a previous capture, go to **File | Open...**. This will open a dialog the user can use to navigate the file system and select a Data Center file. Data Center files have the extension *.tdc.

When opening a file, the current capture data will be overwritten. If the current capture data has not been saved, the user will be prompted (Figure 15) to save their data. Click **Save** to save the capture data, or **Discard** to ignore it.

It is possible to append additional capture data to an existing file. Information rows in the data set will indicate when the separate captures were performed.

There is one exception, however. Additional data cannot be captured to a file that was previously saved with a Filtered View.

## 4.12 Exporting a Capture

It is possible to export an entire capture or a subset of a capture for future analysis. A capture can only be exported as a comma delimited file (*.csv, *.xml).

To export a capture, go to **File | Export**, or use the keyboard shortcut **Ctrl+E**. The application will open a file dialog to determine the name and location of the export file. The Aardvark Control Center (*.xml) format is an available filetype when exporting I²C and SPI captures. These files can be replayed in Control Center to simulate the transmissions in the original capture.

After the user supplies a valid file location and format, the application will allow the user to set the file's Export Settings (Figure 27). There are currently no available export setting for XML format.

### Export Settings

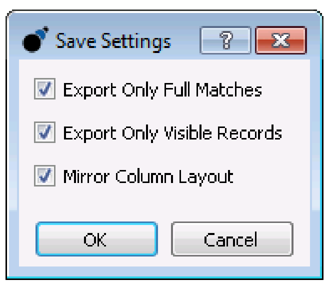



***Figure 27:*** *Export Settings Dialog*

The export settings must be configured each time a capture is exported, unless the **Re-Export** option is being used. See the **Re-Exporting a Capture** section below for more details.

Please note that the current state of the filter is used when exporting. This means that the filter must be enabled in order to export a filtered view.

**Export Only Full Matches**

If selected, only the records that are full matches will be exported. Any record that is a soft match (denoted by the faded color) will not be exported when this option is enabled.

**Export Only Visible Records**

If selected, only the records that are visible will be exported. Any record that is hidden, such as a record whose parent is collapsed, will not be exported when this option is enabled.

**Mirror Column Layout**

If selected, only columns visible in the transaction window at the time of export will be written to the CSV. In addition, the order of these columns in the CSV will match their order in the transaction window.

### Re-Exporting a Capture

After an export has been performed, the **File | Re-Export** option can be used to re-export the capture to the same file, with the same export settings. The user is not prompted for any additional information.

If an export has not previously been performed, the user will be prompted for a file name and export settings as if the **File | Export** option was selected.

Please note that it is the current state of the filter that is used when exporting, and not the state when the initial export was performed.

## 4.13 Preferences

The Preferences dialog (Fig. 28) allows the user to configure the Data Center software.

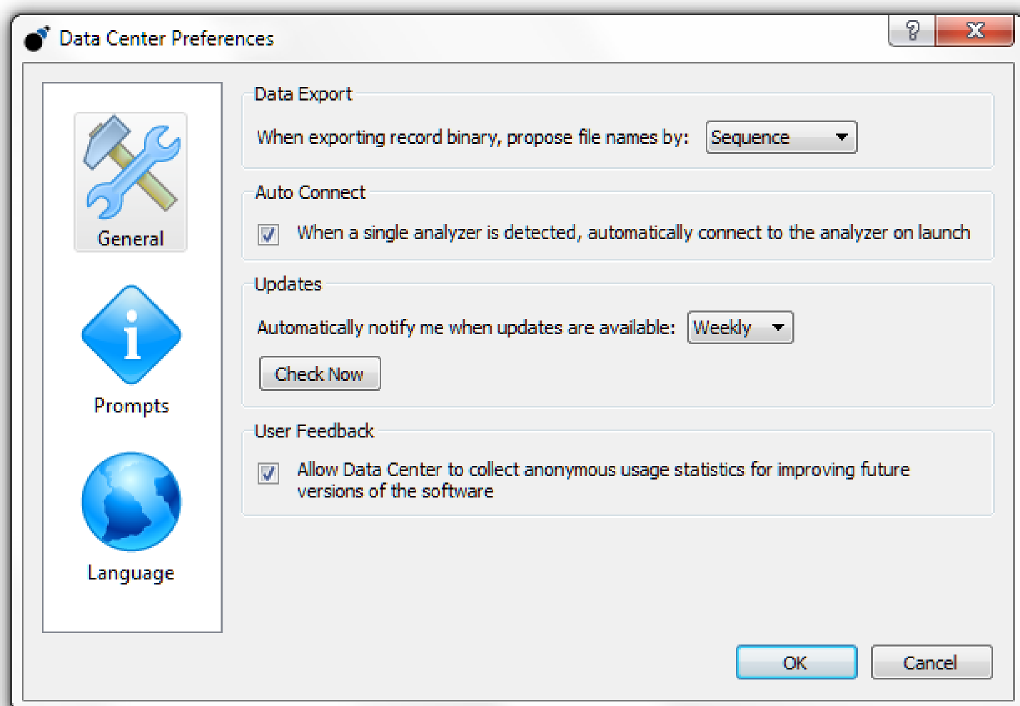



***Figure 28:*** *Preferences Dialog*

**General**
The **Data Export** option allows the user to select how file names are proposed when exporting record data. After a data export has been performed once, the Data Center software will use the previous filename to propose a new filename. When **Sequence** is selected, the numeric part of the filename will be incremented. When **Record Index** is selected, the numeric part of the filename will be replaced with the selected record's index. If the previous filename does not end with a numeric value, the proposed filename will be the filename used for the previous export.

For example, assume the first export has been performed with filename `cap_000.bin`. With **Sequence** selected, the proposed filename for the next data export will be `cap_001.bin`. With **Record Index** selected, and the selected record's index being 1234, the proposed filename will be `cap_1234.bin`.

The **Auto Connect** option allows single-analyzer users to conveniently bypass the device connection stage (Section 4.4) during Data Center launch. When this option is enabled and **one analyzer** is attached to the system, Data Center will automatically connect to the analyzer on launch.

**Note:** Auto connect may fail if the device is in use by another application or is otherwise not connectable. In this case, a warning will appear alerting the user of the connection failure.

This General section also allows the user to configure when to be notified that a newer version of Data Center software is available.

**Prompts**
The Prompts section contains options for configuring the behavior of Data Center during several scenarios.

**Language**
The Language section provides a list of available languages to choose from. Selecting the preferred language will translate all the strings used throughout the application.

## 4.14 Changing Settings

The Capture Settings and Device Settings dialogs allow the user to change the parameters of a capture. Capture options that are common to all protocols are available in the Capture Settings dialog. In addition, each protocol has device settings which are discussed in the protocol-specific sections of this manual.

To change the capture settings, Click on **Capture Settings...** in the toolbar, or go to **Analyzer | Capture Settings...**.

### Capture Data Limit

The capture data limit setting limits the amount of memory that captured data can occupy on the analysis computer. Once this limit is reached, the capture will either automatically stop or records will begin to be deleted, depending on the circular buffer setting.

A slider in the capture settings window allows the capture data limit to be set as percentage of available memory. The slider can only be set to percentages that correspond to a valid capture

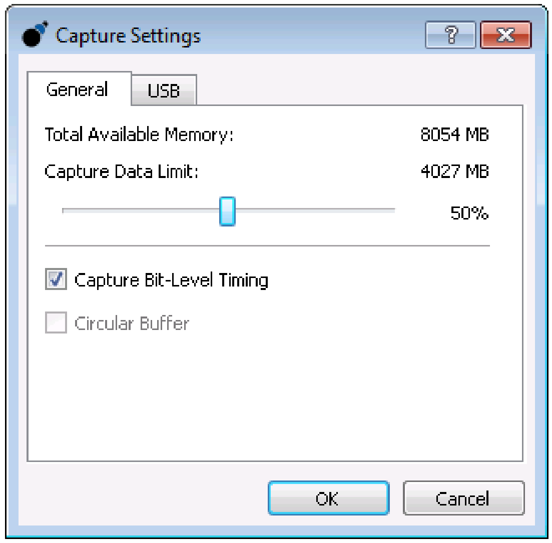



***Figure 29:*** *Default Capture Settings*

data limit. The capture data limit must be at least 16 MB and no greater than 80% of available system memory.

By default, The capture limit is set to 50% of available memory. The capture limit has an upper limit of 80% of the available memory. Total Phase recommends using caution when setting the capture limit above this amount. On an extremely busy computer, the capture limit should be set even lower. If the application starts swapping memory, incoming capture data may be lost.

Note that the capture buffer limit is intended to be an approximate, as it is difficult to keep exact under real-time capture constraints. As such, the actual capture size may fluctuate around this setting when the circular capture buffer is enabled, or go slightly over when it is not.

#### Bit-Level Timing

The capture of bit-level timing is optional. By not capturing bit-level timing data, the Data Center software will have improved performance, reduced memory usage and will be able to capture data for a longer period of time before running into the capture data limit.

The bit-level timing option is only available when capturing SPI and $I^2C$ data.

#### Software Circular Buffer

The software circular buffer option allows the Data Center application to discard past records during a capture in order to keep the capture size below the capture data limit. The removing of records will begin at the start of the first index of the transaction table, regardless of whether or not that record is visible.

This option is only available when the **Capture Data Limit** is set to 1024 MB or less.

The software circular buffer is not available with the Beagle USB 5000 analyzer. Instead, the Beagle USB 5000 analyzer features a hardware circular buffer which is discussed in Section 6.

Please note that when using the circular buffer for USB captures, you may lose the ability to see your capture with class-level parsing. This is because certain records necessary for class-level parsing may be discarded. These required records are usually sent when the device is initially plugged in, and are therefore the first to be dumped when the circular buffer rolls over. If you would like to preserve the ability to use class-level parsing, enable the classification option before starting the capture. This activates a special feature that preserves the records necessary for class-level parsing.

## 4.15 Getting Help

Help files are available to assist the user. To open the manual, go to **Help | Manual**, or use the keyboard shortcut **F1**. To open the datasheet, go to **Help | Datasheet**, or use the keyboard shortcut **F2**. To visit the Total Phase website, select the menu item **Help | Website**, or use the keyboard shortcut **F3**. Each of these commands are also available from the toolbar. See Section 4.3 for details.

## 4.16 Example Captures

Examples of common types of captures are available in all supported protocols for the user to peruse. To access the Examples dialog (Figure 30), either select the **File | Examples...** menu item, select the **Help | Examples...** menu item, or use the keyboard shortcut **F4**.

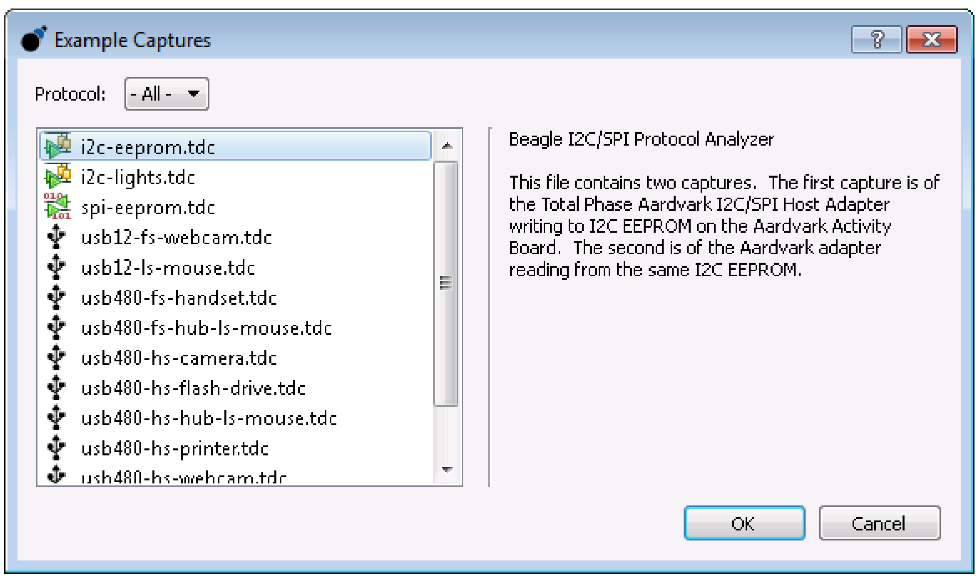
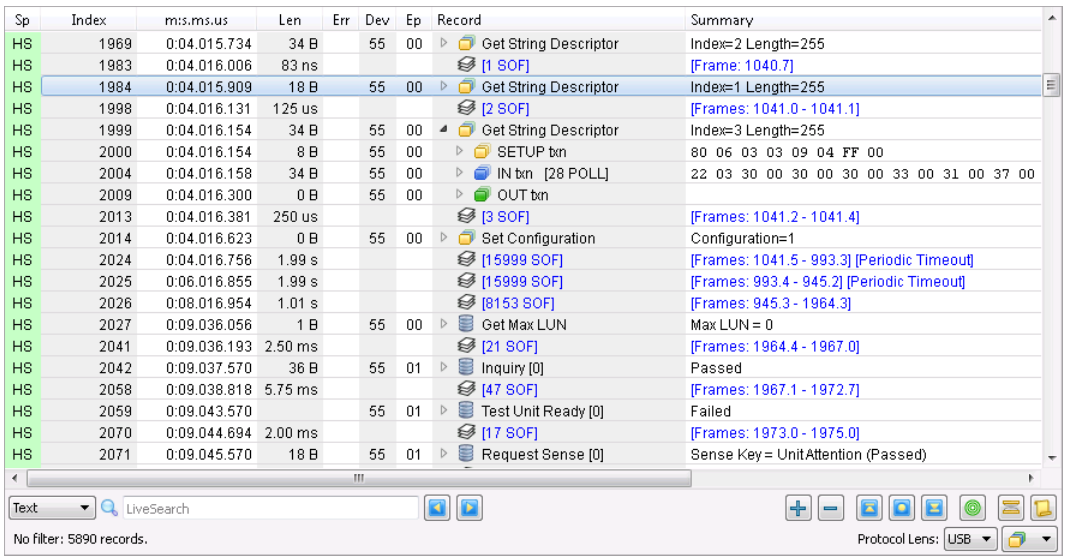



***Figure 30:*** *Example Captures Dialog*

# 5 General Monitoring

The Data Center application has a common interface regardless of what protocol is being captured. This section describes those common features. Details that are specific to each protocol can be found in the protocol-specific sections below.

## 5.1 Transaction Window



| Sp | Index | m:s.ms.us | Len | Err | Dev | Ep | Record | Summary |
|---|---|---|---|---|---|---|---|---|
| HS | 1969 | 0:04.015.734 | 34 B | | 55 | 00 | Get String Descriptor | Index=2 Length=255 |
| HS | 1983 | 0:04.016.006 | 83 ns | | | | [1 SOF] | [Frame: 1040.7] |
| HS | 1984 | 0:04.015.909 | 18 B | | 55 | 00 | Get String Descriptor | Index=1 Length=255 |
| HS | 1998 | 0:04.016.131 | 125 us | | | | [2 SOF] | [Frames: 1041.0 - 1041.1] |
| HS | 1999 | 0:04.016.154 | 34 B | | 55 | 00 | Get String Descriptor | Index=3 Length=255 |
| HS | 2000 | 0:04.016.154 | 8 B | | 55 | 00 | SETUP txn | 80 06 03 03 09 04 FF 00 |
| HS | 2004 | 0:04.016.158 | 34 B | | 55 | 00 | IN txn [28 POLL] | 22 03 30 00 30 00 30 00 33 00 31 00 37 00 |
| HS | 2009 | 0:04.016.300 | 0 B | | 55 | 00 | OUT txn | |
| HS | 2013 | 0:04.016.381 | 250 us | | | | [3 SOF] | [Frames: 1041.2 - 1041.4] |
| HS | 2014 | 0:04.016.623 | 0 B | | 55 | 00 | Set Configuration | Configuration=1 |
| HS | 2024 | 0:04.016.756 | 1.99 s | | | | [15999 SOF] | [Frames: 1041.5 - 993.3] [Periodic Timeout] |
| HS | 2025 | 0:06.016.855 | 1.99 s | | | | [15999 SOF] | [Frames: 993.4 - 945.2] [Periodic Timeout] |
| HS | 2026 | 0:08.016.954 | 1.01 s | | | | [8153 SOF] | [Frames: 945.3 - 1964.3] |
| HS | 2027 | 0:09.036.056 | 1 B | | 55 | 00 | Get Max LUN | Max LUN = 0 |
| HS | 2041 | 0:09.036.193 | 2.50 ms | | | | [21 SOF] | [Frames: 1964.4 - 1967.0] |
| HS | 2042 | 0:09.037.570 | 36 B | | 55 | 01 | Inquiry [0] | Passed |
| HS | 2058 | 0:09.038.818 | 5.75 ms | | | | [47 SOF] | [Frames: 1967.1 - 1972.7] |
| HS | 2059 | 0:09.043.570 | | | 55 | 01 | Test Unit Ready [0] | Failed |
| HS | 2070 | 0:09.044.694 | 2.00 ms | | | | [17 SOF] | [Frames: 1973.0 - 1975.0] |
| HS | 2071 | 0:09.045.570 | 18 B | | 55 | 01 | Request Sense [0] | Sense Key = UnitAttention (Passed) |



***Figure 31:** The Transaction window with Protocol Lens set to USB*

The Transaction window (Figure 31) displays all the transactions that were captured on a serial bus in real time, as well as bus events or capture meta-information such as when the capture began or ended. Each discrete message on the bus will appear as a single record, or transaction, in the Transaction window. When a transaction is selected in the Transaction window, the byte content and/or timing data of that transaction is displayed in the Details window. Packet meta-information such as originating device and time stamp will be displayed in the Navigator window.

The transaction table provides the following information that is common among all protocols:

**Index**
The transaction index number. The first record of the first capture is considered index 0.

**Timestamp**
The timestamp column has two different modes of operations.

By default, the column displays the time that the transaction was captured. The time counter starts at 0 when a capture is started. Every time a new capture is started, the time is reset to zero. The reference point in this mode can be changed to see the time relative to a particular event.

The second mode of operation is **Interval Time View** which can be set in the context menu. In this mode, the column displays the delta time between the start of the previous record and the start of the current record. To get the delta time between the end of the previous record and the start of the current record, **EoR Reference** must be enabled in the context menu.

The timestamp column can be configured to display timestamps at millisecond (min:sec.ms), microsecond (min:sec.ms.µs), or nanosecond (min:sec.ms.µs:ns) resolution. To change the timestamp precision of the Transaction window, open the context menu over the table and select the desired precision from the **Timestamp Resolution** menu. The timestamp of the transaction is displayed in nanosecond precision in the Info pane.

**Duration (Dur)**
The elapsed time that the transaction was in the bus. The duration value displayed is shown in an abridged format. The full duration to nanosecond precision is displayed in the Info pane.

**Length (Len)**
The number of bytes in the transaction.

**Error codes (Err)**
Error codes listing abnormal conditions that occurred while capturing the transaction. See Table 2 for the possible error codes.

***Table 2:*** *Error code values*

| Code | Meaning | Description |
|---|---|---|
| U | Unexpected | A packet or event occurred outside of the expected context. |
| T | Time out | Capture for transaction timed out while waiting for additional data. |
| M | Middle of packet | Data collection was started in the middle of a packet. |
| S | Short buffer | Transaction was too long to fit in capture buffer. |
| P | Partial last byte | The last byte in the buffer is incomplete. The number following the error code indicates how many bits were received for the last byte. |

In addition, there are error code values specific to USB transactions listed in Table 7 and error values specific to CAN transactions in Table 8.

**Record**
A description of the transaction.

**Data**
The bytes in the transaction.

There are also additional protocol specific features for the Transaction window which provide high level information about the data captured. Specific information about these features can be found in their respective sections in this manual.

Each column in the Transaction window can be hidden by right clicking the column's header and selecting the appropriate option in the context menu. Additionally, columns can be resized and reordered to create a custom layout for each lens.

Right clicking in the Transaction window will bring up a context menu with the following options.

**Export Binary Data**

Export the selected record's data to a file. The data that is exported is the same as that shown in the Details pane. If a previous data export has been performed, a new filename will be proposed according the Data Export setting in the Preferences dialog (Section 4.13).

**Quick Filters**
Several commonly used filters are accessible though the quick filters menu. Only filters relevant to the currently selected record are available for use.

**Timestamp Reference**
By default, each capture session starts at time 0 and the timestamp displayed for each transaction is relative to the Capture started event. The user can select any transaction in a session to be the time reference and the timestamps of the other transactions will be adjusted accordingly. To denote that a transaction other than the Capture started event is set as the time reference, the timestamps for the entire session are colored. Also, the time reference is set on a capture session basis since the timestamps for each session are independent.

**Interval Time View**
This is a secondary mode of operation for the Timestamp column. When enabled, the timestamp column will display the delta time from the start of the previous transaction to the start of the current transaction.

**EoR Reference**
By default, delta time calculations are based on the start time of the previous record and the start time of the current record. By enabling **EoR Reference** (End of Record), the delta time calculations will be based on the end time of the previous record and the start time of the current record. When **EoR Reference** is enabled, the delta time displayed in the **Timestamp** column and in the **Status Bar** will be affected. Additionally, the "Delta time" heading in the **Status Bar** will contain an asterisk (*) to indicate that **EoR Reference** is enabled.

**Timestamp Resolution**
The resolution of the timestamp can be set to milliseconds (ms), microseconds (us), or nanoseconds (ns).

**Expand/Collapse All**
Expand or collapse all the transactions in the Transaction window.

**Fully Expand/Collapse Branch**
Expand or collapse the entire branch below the selected record. This does not affect any record in other branches or records that are parents of the selected record. Holding the Ctrl key while double clicking a record has the same effect.

**Expand All to Level**
Expand all transactions to the level of the selected record.

**Comments**
Comments allow the user to insert a record into the capture stream that contains arbitrary text. When a capture is running, comments can only be appended to the end of the capture buffer. When a capture is not running, comments can be inserted into the capture buffer as a top level record. This means that comments cannot be inserted in the middle of a transaction tree. Once a comment has been inserted, it can be modified or removed only when the capture is not running.

### Delta Time and Data Payload Display

When moving the mouse over the Transaction window, the transaction that the mouse pointer is over will be highlighted. The time difference between this transaction and the currently selected transaction will be displayed as "Delta time" in the status bar at the bottom of the application window. The time displayed will be the time difference between the start times of the two transactions. When **EoR Reference** is enabled, the time displayed will be the time difference between the end of the first transaction and the start of the second transaction.

Next to the "Delta time", the "Transferred length" will be shown along with the bandwidth. The "Transferred length" is the number of data bytes (i.e. the summation of the length field inclusively) between the selected transaction and the hovered transaction. The corresponding data bandwidth for this range is displayed in parenthesis. Be aware that due to the way transactions may be ordered, the reported bandwidth is an approximate value. Also note that the data payload information is only displayed when the selected and hovered transactions are at the same level in the same branch.

The delta time and data payload will only be displayed when the two transactions are from the same capture session. So if you start then stop a capture, then start another capture, these statistics will only be displayed when both the selected transaction and the one the mouse is over are from the same capture session.

### Goto

It is possible to jump to a specific record index in the Transaction window with Data Center's Goto feature. Goto can be activated by selecting **Goto** in the application **Edit** menu, or by pressing **Ctrl+G**.

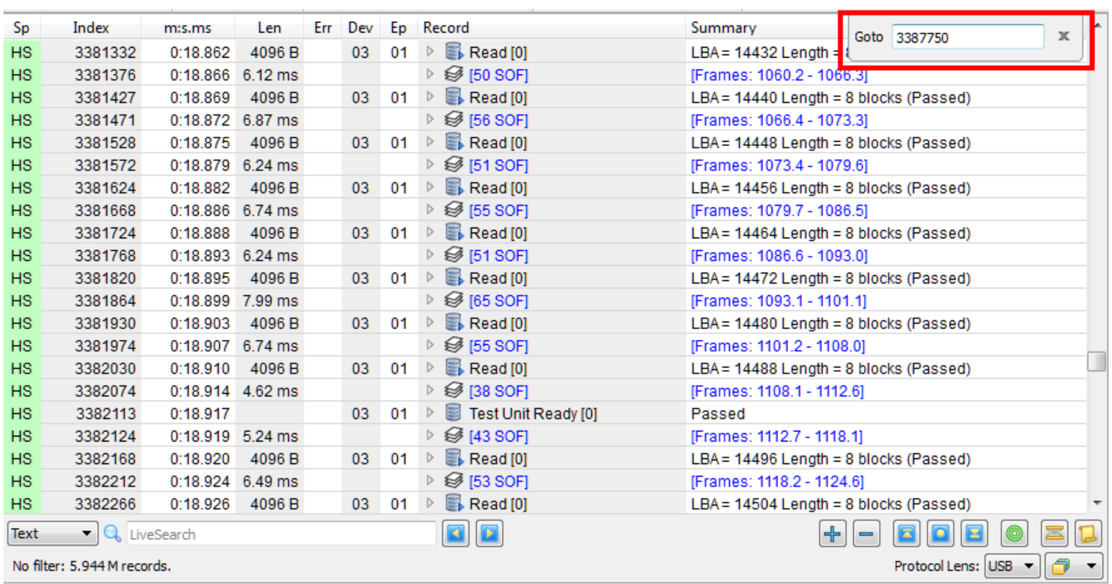


***Figure 32:*** *The Transaction window with the Goto box highlighted*

When Goto is activated, a Goto box appears in the upper right corner of the Transaction window

(Figure 32). Focus is automatically given to the Goto index entry box, so the user can begin typing as soon as the Goto box appears.

To jump to a particular record, simply enter the record's index in the Goto index entry area, and press **Enter** or **Return**. If the index is valid, the Transaction window will automatically jump to the desired record index, and the Goto box will disappear.

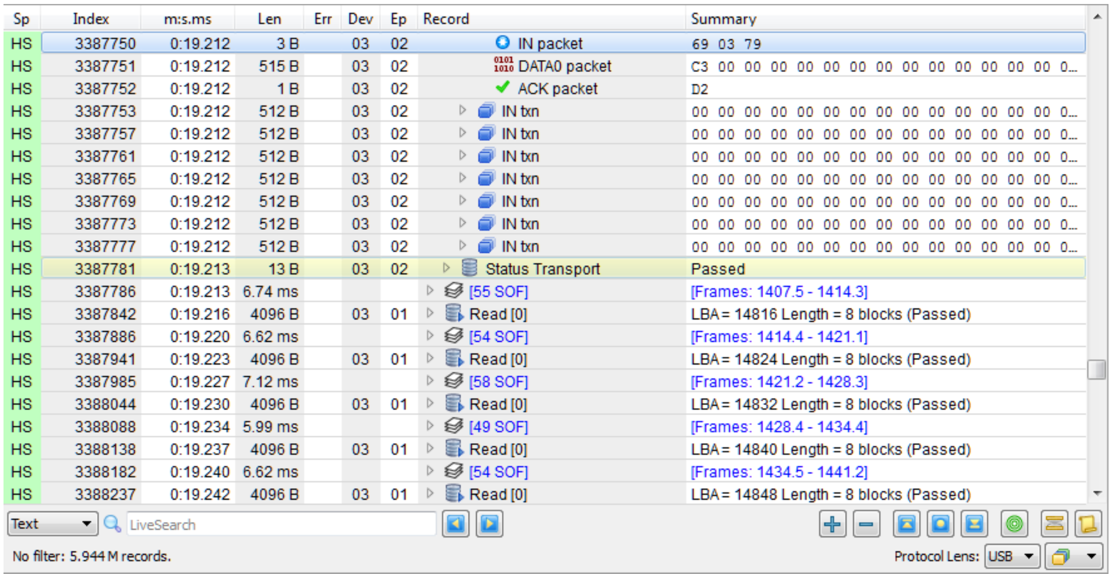



***Figure 33:*** *The Transaction window following a successful use of Goto*

Parent records are expanded as needed to expose the destination record (Figure 33). If the record of interest is hidden by a filter, the Transaction window will jump to the nearest top-level record with an index less than or equal to the Goto index.

To dismiss Goto, either click on the **x** to the right of the Goto index entry area, or move cursor focus out of the Goto index entry area by pressing Tab / ESC or by clicking outside of the Goto box.

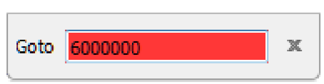



***Figure 34:*** *The Goto index entry area pulsing red to indicate an invalid index*

The destination index is always validated before a jump occurs. The index must fall between the first and last record indices (inclusive). If the entered index is invalid, the Goto index entry box will pulse red (Figure 34) and Goto will remain visible.

## 5.2 Details Window

The Details window provides lower level detailed information about a specific transaction. There are two types of panes available. A Data pane is available for all protocols, and a Timing pane is available for SPI and I$^2$C captures. The way the data is displayed in these modes will depend on the protocol type.

Additional protocol specific viewing modes may be available and are documented in their respective sections.

### Data Pane

The Data pane (Figure 35) provides a way for the user to examine the raw bytes of a transaction, regardless of the protocol specific structure of the data. By default, the Data pane is configured to show the data in hexadecimal and ASCII format. Right clicking in the Data pane will bring up a context menu that allows the user to configure the view by adding additional panes and adjusting the size, grouping, and radix of each pane.

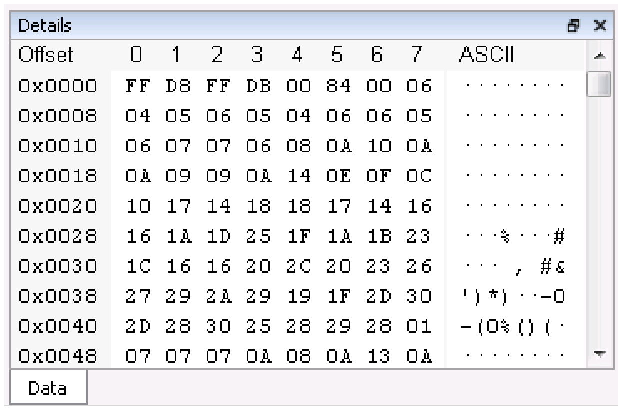



***Figure 35:*** *Details Window - Data Pane*

When a pane is configured to display the data in ASCII format, a "." is used if the equivalent ASCII character is a non-printing character. Also, when a sequence of bytes is highlighted in one pane, the corresponding byte representation of the same data in the other panes is also highlighted. This allows the user to easily translate between the different representations of the data.

Please see the protocol specific sections later in this document for more protocol-related features.

### Timing Pane

The Timing pane of the Details window (Figure 36) provides bit-level timing for the data of $I^2C$ and SPI transactions. Each byte of the transaction appears as a row in this pane. All the bytes from the transaction will be displayed in this pane, including start and stop conditions.

The first line of the table displays the transaction timestamp as well as the transaction duration, both to nanosecond precision.

Each row contains the following information:

**Offset**
The offset position of the byte.

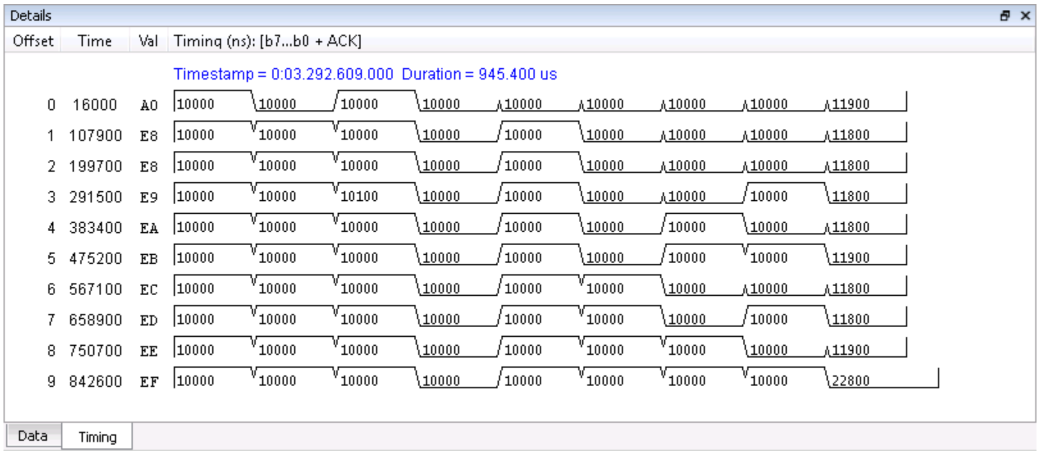



***Figure 36:*** *Details Window - Timing Pane*

**Time**
The time in nanoseconds from the start of the transaction to the start of the byte.

**Value**
The hexadecimal value of the byte.

**Timing**
A graphic display of each individual bit of a byte. Each bit is displayed as being either high or low with the time in nanoseconds from the start of the current bit to the start of the subsequent bit.

The lengths of the timing blocks in the graph are not drawn to scale and are intended merely to provide a hint to the relative time scale of one bit time to the next.

Please note that depending on the protocol, the bit order may be MSB or LSB. You can determine the bit order by looking at the column label. The text in the label will indicate if the data is MSB (b7. . . b0) or LSB (b0. . . b7).

In the case of the I$^2$C protocol, the timing mode displays 9 bits per line. The ninth bit is the ACK/NACK bit.

In the case of the SPI protocol, the timing mode displays both MOSI and MISO. The MOSI line is displayed in red and the MISO line is blue.

Please see the protocol specific sections later in this document for more details.

## 5.3 Command Line Window

The Command Line window gives the user another method to interact with the Data Center application. All operations that can be done by pointing and clicking throughout the application can also be done via the command line. When an operation is performed in the application, its corresponding command line command is echoed in the command line output and added to the command line history. The user can use the arrow keys in the command line input box to scroll back through the command line history and edit or repeat previous commands.

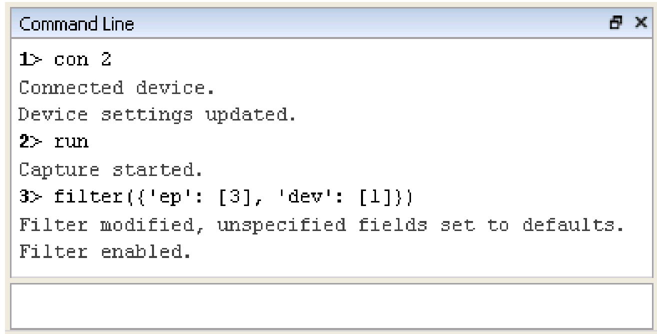



***Figure 37:*** *Command Line Window*

To view all available commands, type **help** into the command line. Type **help COMMAND** to see help specific to a particular command.

When the command line input is in focus, pressing the escape key once will clear the command line input box, and double pressing the escape key will clear the command line output and history.

The command line uses the Python syntax and behaves similarly to the command line found in the Python interpreter. This means that local variables can be defined and control structures can be used as well. The arguments passed to the Data Center application commands are generally singleton values (such as integers, strings, or True/False), lists, or dictionaries. For more information on Python syntax and data structures, see http://www.python.org.

## 5.4 Bus Pane

The Bus pane of the Navigator window shows the devices that have been detected on the serial bus being monitored. For CAN, $I^2C$ and USB, the devices are distinguished by the IDs or addresses. When an SPI bus is monitored, all the traffic is lumped into one device per capture since the Beagle can only monitor one slave select at a time. Clicking on a device in the bus tree will reveal more detailed information regarding that device. For additional information on what is displayed for each protocol, see the protocol specific sections later in this document.

Right click on any part of the tree to bring up a context menu with the following options:

- **Filter: Show Only** – Only show records of the selected category.
- **Filter: All Except** – Only show records that are not part of the selected category. (Bus Pane Only)
- **Filter: Disable** – Disable the active filter.
- **Find First** – Find the first instance for the selected category. (Statistics Pane Only)
- **Find Last** – Find the last instance for the selected category. (Statistics Pane Only)
- **Fully Collapse Branch** – Collapse the current branch from the selected element down.
- **Fully Expand Branch** – Expand the current branch from the selected element down.

- **Expand All** – Expand the entire statistics tree.
- **Collapse All** – Collapse the entire statistics tree.

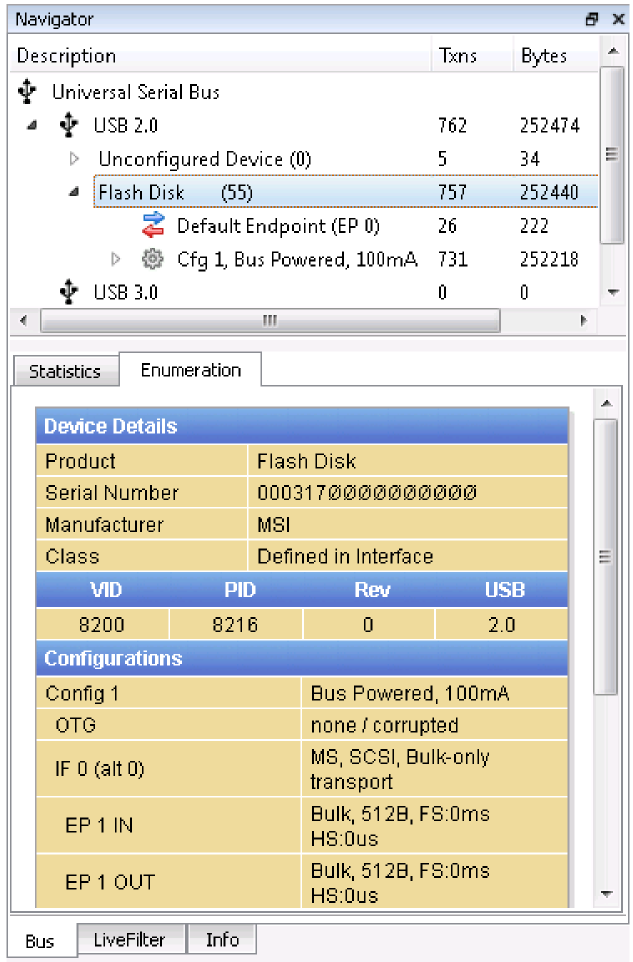



***Figure 38:*** *Bus Pane*

### Real-Time Statistics Pane

The Statistics Pane (Figure 39) provides a real-time count of Errors and protocol specific constructs as data is being captured. The Statistics Pane is tied to the Bus Pane. When a bus is selected in the Bus Pane, the aggregate of the bus level data and the bus' connected devices data will be displayed in the statistics table. When a device is selected, only the device data is displayed.

The **Statistic** column provides a hierarchical view of all the statistics available. The **Available** column provides a count off the total number of packets, transfers, and events counted during the capture. While the capture is active, the **Visible** column is not updated. After the capture is stopped, the **Refresh** button must be clicked to update the **Visible** statistics. The **Refresh** button will be disabled unless an action occurred that may have modified the count of available

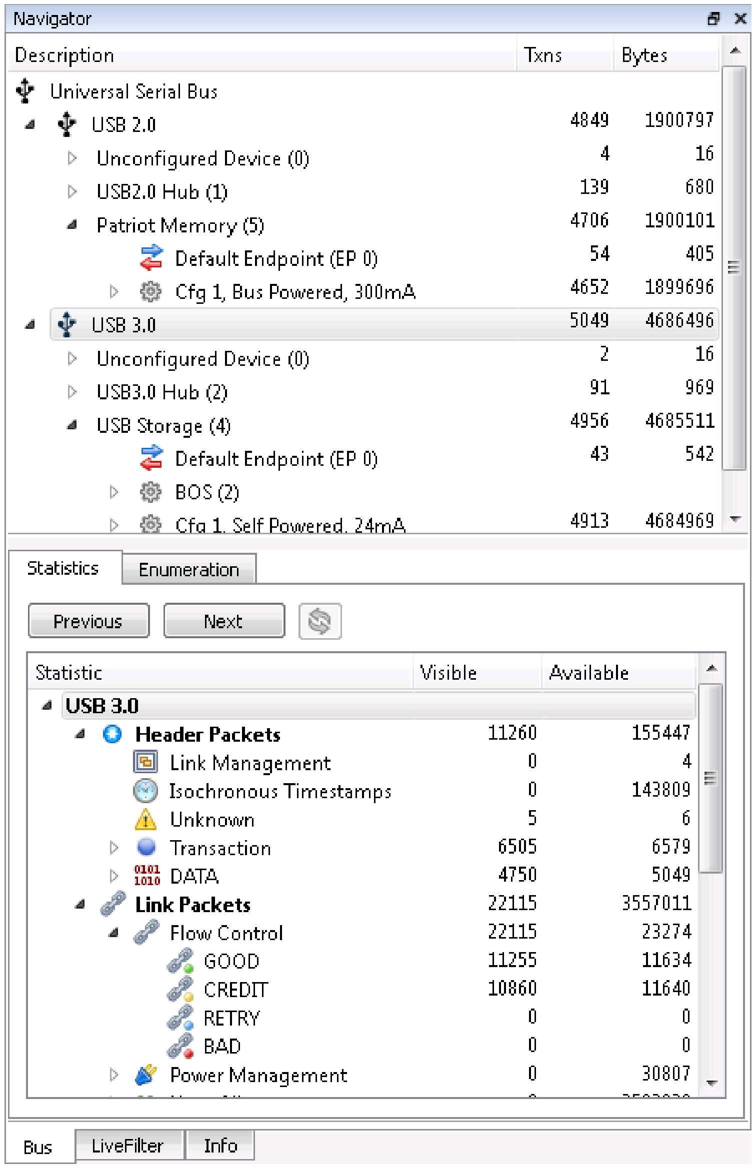



***Figure 39:*** *Real-time Statistics Pane provides a quick and easy way to access types of packets*

data. If a filter is applied, the **Visible** column will show the number of packets, transfers, and errors that match the current filter. The **Available** column will display the total count.

When the individual statistics are expanded in the Statistics Pane, the sub data will vary, presenting data counts of subcategories of the expanded statistic. If a paticular statistic is available for a device and a bus is selected in the Bus Pane, expanding the bottom node of the tree will display data counts for its internal devices.

It is possible to quickly jump to a statistic of interest by selecting the category in the Statistics Pane and clicking the **Previous** Previous and **Next** Next buttons. When the **Next** button is clicked, the capture will be searched chronologically from the current cursor location in order to find the next occurrence of the selected packet, transfer, or error. If the end of the capture is reached, the search will restart at the beginning of the capture. When the **Previous** button is clicked, the capture will be searched in reverse chronological order from the current cursor location in order to find the next occurrence of the selected packet, transfer, or error. If the beginning of the capture is reached, the search will restart at the end of the capture. In the transaction window, the transaction hierarchy will be expanded to expose the matching record of interest.

## 5.5 Filtering

The Filter pane (Figure 40) in the Navigator window allows the user to non-destructively filter the data shown in the Transaction window.

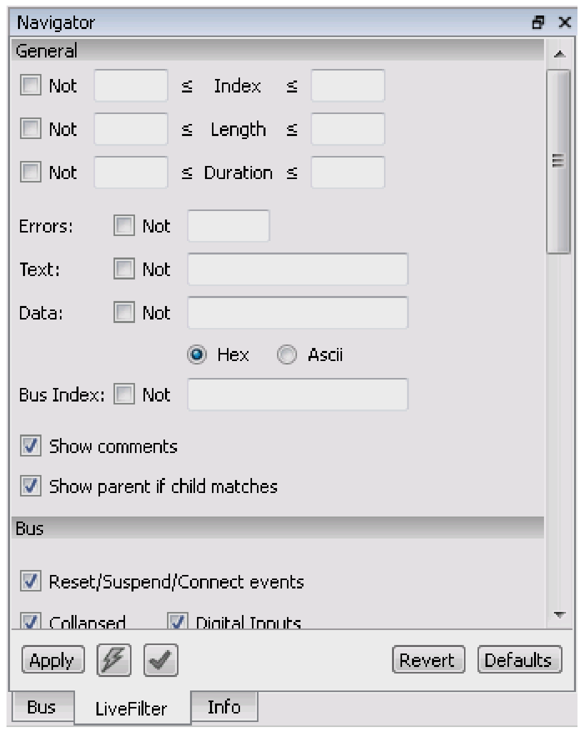



***Figure 40:*** *Filter Pane*

The general filter fields are described below. Information on applying filters can be found in Section 4.8.

### Filter Fields

The general set of filters available (Figure 40) for the all the protocols include:

#### Index

An integer range that filters the transactions based on a minimum and maximum index number.

#### Length

An integer range that filters the transactions based on a minimum and maximum length.

#### Duration

An integer range that filters the transactions based on a minimum and maximum duration in nanoseconds.

#### Earliest Time/Latest Time

A time range that filters the transactions based on the earliest and latest timestamp.

#### Errors

A list of codes (as defined in Tables 2, 7, and 8)) that filters the transactions based on whether the transaction contains any of the codes.

The codes must be in a list with no characters between each codes. For example, filtering for transactions that are unexpected or that have timed out would require the Errors field to contain "UT".

Using an asterisk (*) by itself will cause the filter to match any transaction that has an error.

#### Text

A case-insensitive string pattern that filters the transactions based on the text in the Record or Data column. Any raw data that is shown in the Data column will not be examined when running this filter. Use the data filter to filter the raw data.

**Syntax**

There are 4 special characters defined in Table 3 that may be used in the pattern.

***Table 3:*** *Special values for text and data pattern filters*

| Value | Name | Meaning |
|---|---|---|
| ? | Placeholder | One character of any value. |
| * | Wildcard | Zero or more characters of any value. |
| ^ | Start Anchor | Pattern must match at the beginning of the string. Only valid when put at the beginning of a pattern. |
| $ | End Anchor | Pattern must match at the end of the string. Only valid when put at the end of a pattern. |

See Table 4 for examples of text patterns.

*Table 4: Example text pattern filter entries*

| Example | Result |
|---|---|
| `in` | Matches "`IN`"<br>Matches "`PING`" |
| `^in` | Matches "`IN`"<br>Does not match "`PING`" |
| `config*n` | Matches "`Set Configuration`"<br>Matches "`Get Configuration Descriptor`" |
| `config*n$` | Matches "`Set Configuration`"<br>Does not match "`Get Configuration Descriptor`" |
| `get*descriptor` | Matches "`Get String Descriptor`"<br>Matches "`Get Device Descriptor`" |
| `s???t` | Matches "`Start`"<br>Matches "`SPLIT`" |

### Data

A data pattern that filters the transactions based on the data contained in the transaction. The pattern may be specified as either a hexadecimal pattern or an ASCII pattern. Data patterns may contain the same special characters used in text patterns described in Table 3.

#### Syntax

When an ASCII pattern is used, the pattern is case-insensitive and applied to the raw data as it is shown in the ASCII portion of the hex editor found in the Details window. The syntax of ASCII patterns is the same as text patterns (Table 4).

When a hexadecimal pattern is used, each byte, or special character, must be separated with spaces. See Table 5 for examples of hexadecimal data patterns.

*Table 5: Example hexadecimal data pattern filter entries*

| Example | Meaning |
|---|---|
| `1 2 3` | Must contain the sequence of `01 02 03` somewhere in data. |
| `^ 1 ? 2 FF` | Must have `01`, `02`, and `FF` in the first, third and fourth byte positions, respectively. |
| `0 ff $` | Must end with `00 FF`. |
| `^ ? c0 ? $` | Must be exactly three bytes long and have the value `C0` as the second byte. |
| `^ ? ? a5 * 20 $` | Must have `A5` in the third byte and end with `20`. |

### Bus Index

An integer corresponding to a 'device' as presented in the Bus Pane (Section 5.4). In the Bus Pane, the index is presented as "BusIdx" in the name of the device. In older saves where this functionality is not supported, the device name does not contain the "BusIdx" suffix.

### Show parent if child matches

Checking this box will force any non-matching parents of a matching record to show as a soft match (Section 5.5).

Figure 41 shows what the Transaction window would look like after filtering for just DATA packets in a USB capture when the **Show parent if child matches** option is enabled. The parent of each of the DATA packets is shown as a soft match.

| | | | |
|---|---|---|---|
| 02 | 00 | ⊟ IN txn [2 POLL] | 2E 03 42 00 65 |
| 02 | 00 | DATA1 packet | 4B 2E 03 42 00 |
| 02 | 00 | ⊟ OUT txn | |
| 02 | 00 | DATA1 packet | 4B 00 00 |
| 02 | 00 | ⊟ Get String Descriptor | Index=0 Length=255 |
| 02 | 00 | DATA0 packet | C3 80 06 00 03 |
| 02 | 00 | ⊟ IN txn [2 POLL] | 04 03 09 04 |
| 02 | 00 | DATA1 packet | 4B 04 03 09 04 |

***Figure 41:*** *"Show parent if child matches" is checked*

Unchecking the **Show parent if child matches** option hides the non-matching parents. Any record that no longer has a visible parent will have a dot placed to the left of the icon in the Record column. Figure 42 shows an example of this situation.

| | | | |
|---|---|---|---|
| 02 | 00 | ▪ DATA1 packet | 4B 1C 03 54 00 |
| 02 | 00 | ▪ DATA1 packet | 4B 00 00 |
| 02 | 00 | ▪ DATA0 packet | C3 80 06 00 02 |
| 02 | 00 | ▪ DATA1 packet | 4B 09 02 27 00 |
| 02 | 00 | ▪ DATA1 packet | 4B 00 00 |
| 02 | 00 | ▪ DATA0 packet | C3 80 06 00 03 |
| 02 | 00 | ▪ DATA1 packet | 4B 04 03 09 04 |
| 02 | 00 | ▪ DATA1 packet | 4B 00 00 |

***Figure 42:*** *"Show parent if child matches" is not checked*

### Soft Matches

A soft match is a record that doesn't match the applied filter, but one of its children, or its parent, matches. A soft match is displayed with its icon and text grayed out to distinguish it from records that are full matches.

Be definition, all children of a matched record will be shown. If a child also matches the filter, it is a full match, otherwise it is a soft match. Similarly, the parents (up to the top level) of a matched record are soft matches if they don't match the filter, or they are full matches if they do. The user can disable the parents from showing using the **Show parent if child matches** filter (Section 5.5).

## 5.6 Info Pane

The Info pane (Figure 43), located in the Navigator window, shows detailed information about the record that is selected. The information may include:

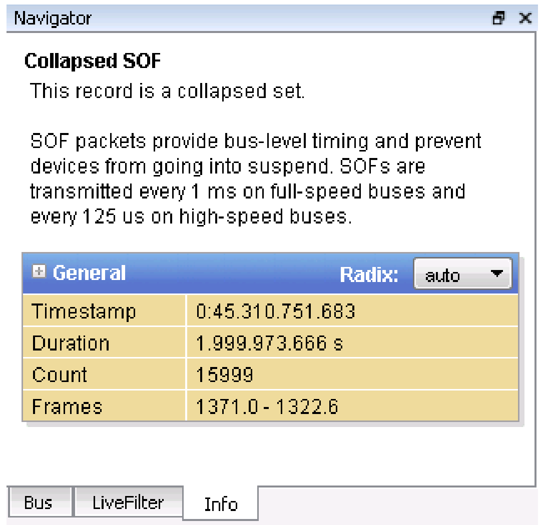



***Figure 43:*** *Info Pane*

- A description of the type of record and how it used in the protocol.
- The data represented in tabular form with fields parsed for convenience. There may be rows in a table that are hidden. To reveal hidden rows, use the +/- button to the left of the table title. The user may also change the format of the data shown with the radix button in the upper right corner of each table. The radix options are: decimal, hexadecimal, binary, and auto. "Auto" indicates that the format was chosen to be the most common or natural radix of the expressed field. It may be any one of the other modes based on what is appropriate for the data in that table row.
- A description of the error codes.

## 5.7 Capture Control Window

The Capture Control Window (Figure 44) provides the ability to start and stop a capture while providing real-time statistics about the capture in progress.

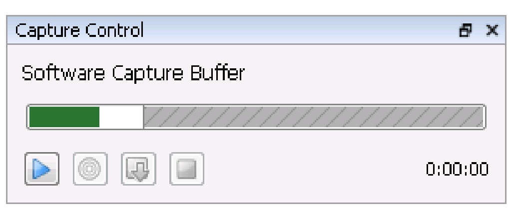



***Figure 44:*** *Capture Control Window*

The “Software Capture Buffer” progress bar in the Capture Control Windows represents the total amount of memory available in the Analysis PC. The white portion of the progress bar represents the amount of memory that can be used by the Data Center software for capturing data. Using the Capture Settings dialog, the Capture Data Limit can be adjusted.

When a capture is in progress, the progress bar will begin to fill up with a solid green bar. The green bar represents the amount of memory used by the Data Center software for the capture. When the status bar is completely green, the capture data limit has been reached and the capture will automatically stop. A timer in the bottom right corner displays the elapsed time since the capture was started.

The Capture Control Dialog provides a number of buttons to control the capture.

**Start Capture**
Use the **Start Capture** button to start the capture. When the capture is active, this button will be disabled.

The capture can also be started with the button in the toolbar, selecting the menu item **Analyzer | Run Capture**, or using the keyboard shortcut **<Ctrl>+R**.

**Manual Trigger**
Use the **Manual Trigger** button to manually trigger the capture. This button is only available when using the Beagle USB 5000 analyzer. Once the capture has been triggered, this button will be inactive.

The capture can also be triggered using the keyboard shortcut **<Ctrl>+T**.

**Stop Capture, Download Data**
Use the **Stop Capture, Download Data** button to stop the capture and to continue to download all captured data. This button is only available when using the Beagle USB 5000 analyzer. When the capture is not active, this button will be disabled.

**Stop Capture, Stop Download**
Use the **Stop Capture, Stop Download** button to stop the capture and to immediately stop downloading data. When the capture is not active, this button will be disabled.

The capture can also be stopped with the button in the toolbar, selecting the menu item **Analyzer | Stop Capture**, or using the keyboard shortcut **<Ctrl>+R**.

## 5.8 Block View

The Block View (Figure 45) provides an alternate representation of the selected record that combines the hierarchical layout of the Transaction window with the detailed information found in the Info Pane.

The record tree that the selected record belongs to is displayed in the Block View, with the selected record highlighted in blue. Rows that have an arrow next to the first block can be expanded to reveal hidden fields. To change the text size in the Block View, right click and select the appropriate zoom option.

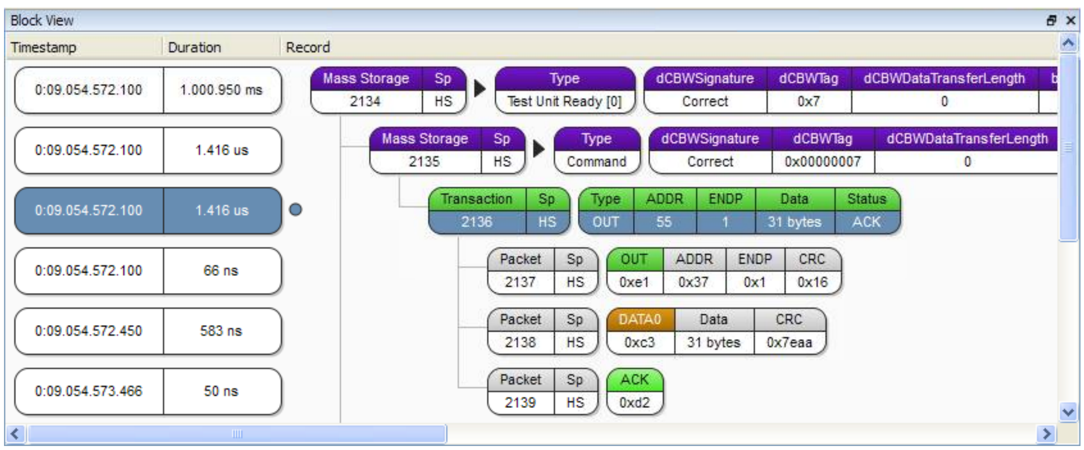



***Figure 45:*** *Block View*

# 6 USB Monitoring

The Beagle USB 5000 SuperSpeed Protocol Analyzer is capable of monitoring super-speed, high-speed, full-speed, and low-speed USB devices in real time.

The Beagle USB 480 Protocol Analyzer is capable of monitoring high-speed, full-speed, and low-speed USB devices in real time.

The Beagle USB 12 Protocol Analyzer is capable of monitoring full-Speed and low-Speed USB devices in real time.

The Data Center software is able to parse the USB packets and provides the user with several powerful tools for filtering the captured data.

## 6.1 Performing a USB Capture

Here are the steps for starting a capture with the Beagle USB 5000 analyzer, the Beagle USB 480 analyzer, or the Beagle USB 12 analyzer.

1. Start the Data Center application.
2. Plug in the power cord and turn on the analyzer when using the Beagle 5000. Make sure the white Total Phase logo LED has illuminated. This step is not necessary when using other Beagle analyzer since they draw from the USB power from the analysis PC.
3. Connect the Beagle USB analyzer to the analysis computer. Make sure that the green indicator LED has illuminated when using the Beagle USB 480 analyzer. **Be sure the analyzer is powered before plugging in any devices on the capture side to ensure the target device can function properly.**
4. Click the **Connect to Analyzer...** button in the toolbar and connect to the analyzer.
5. Ensure the Protocol Lens is set to **USB**.
6. Connect the Beagle USB analyzer to the target host computer. This can be the same computer. Make sure that the amber indicator LED has illuminated on the Beagle 480 analyzer. Make sure the Target Power LED has illuminated on the Beagle 5000.
7. Click the **Run Capture** button to start the data capture. Manually trigger the capture using the Capture Control window when using the Beagle 5000. Once the capture has started, the capture indicator will turn green and an informational transaction will appear in the Transaction window which notes when the capture was started.
8. Connect the target device.

   With the Beagle USB 5000 analyzer, any super-speed, high-speed, full-speed, or low-speed USB device can be connected directly. While USB 3.0 signals occur in parallel to USB 2.0/1.x signals, the Beagle USB 5000 analyzer is capable of recording both simultaneously. When monitoring super-speed USB, please make sure to use USB 3.0 cables to connect the target host and the target device.

With the Beagle USB 480 analyzer, any high-speed, full-speed, or low-speed USB device can be connected directly.

With the Beagle USB 12 analyzer, full-speed and low-speed devices can be connected directly to the Beagle USB 12 analyzer. High-speed devices can also be monitored, but they must be connected through an in-line full-speed hub.

9. To stop the capture, click on the **Stop Capture** button.

The Bus pane will only display a device's descriptors if the device's entire enumeration sequence was captured. If the target device connects to the host before the capture is started, the device enumeration will not be captured and the descriptors cannot be displayed.

## 6.2 USB Capture Settings

Various capture settings can be configured by the user when using a Beagle USB 5000 analyzer or Beagle USB 480 analyzer. Please note that there are no USB-specific user controlled capture settings when using the Data Center application with the Beagle USB 12 Protocol Analyzer.

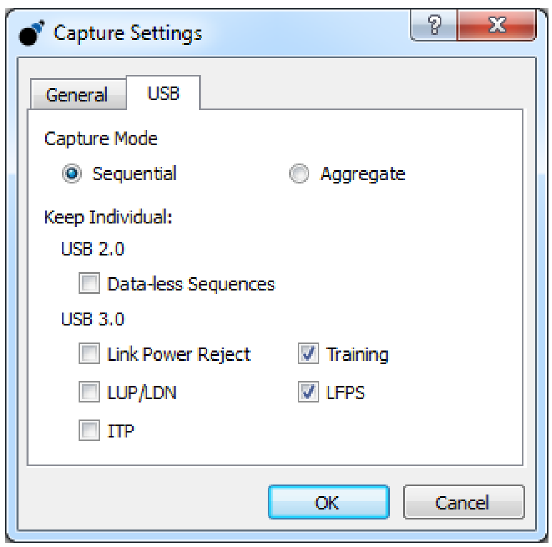



***Figure 46:*** *USB Capture Settings Dialog*

### Capture Mode

The Data Center software has two modes of operation during capture with a Beagle USB 480/5000 Protocol Analyzer.

- **Sequential** – Saves information necessary to display the capture in time-ordered Packet View (Section 6.7).
- **Aggregate** – Discards the information saved in Sequential Mode in order to reduce memory usage. Using this mode disables the ability to use Packet View (Section 6.7).

#### Keep/Drop Options

The Data Center software compresses individual data-less sequences, by default, to reduce memory usage during capture with a Beagle USB 480 analyzer or Beagle USB 5000 analyzer.

For USB 2.0 captures, the user can turn off compression and keep individual data-less sequences by checking **Data-less Sequences**.

For USB 3.0 captures, the user can turn off compression and keep individual sequences for the following types of traffic by checking their respective boxes:

- Link Power Reject
- Training
- LUP/LDN
- LFPS
- ITP

### 6.3 USB Device Settings

There are several USB device settings that can be set by the user when using a Beagle USB 480 Protocol Analyzer or a Beagle USB 5000 Protocol Analyzer. Some of these settings are shared between the two analyzers and others are specific to the Beagle USB 5000 analyzer. Please note that there are no user controlled device settings when using the Data Center application with the Beagle USB 12 Protocol Analyzer.

The USB device settings can be changed in the Device Settings dialog (Figure 47). To open this dialog, click on the **Device Settings...** button or go to **Analyzer | Device Settings...**.

#### USB 2.0 Bus Speed

The Beagle USB 480/5000 analyzer has the ability to automatically detect the speed at which the USB-under-test is operating. However, there may be a situation where the user wishes to explicitly lock it to a specific USB bus speed. This control allows the user to switch between the Beagle analyzer automatically detecting the bus speed to locking the hardware to a specific speed.

If the user knows that the traffic on a bus will always be a certain speed, there can be at least one minor advantage to locking the target speed in the capture settings. Specifically for low-speed and full-speed devices, by locking the bus speed, the Beagle analyzer will not attempt to detect high-speed signaling levels. This will help mitigate the appearance of Chirp J/K and Tiny J/K events which are not of great importance for low-speed- and full-speed-only devices, and these events would otherwise only serve to clutter the display.

#### Omit Packets Matching Beagle Analyzer's Device Address

When the analysis port of the Beagle USB 480 analyzer is connected to the same USB host controller as the traffic being monitored, the Beagle analyzer may observe its own USB traffic. This is because all downstream packets from the host are broadcast to all USB links. Therefore, packets from the host to the Beagle analyzer may appear on the capture side of the Beagle USB 480 analyzer.

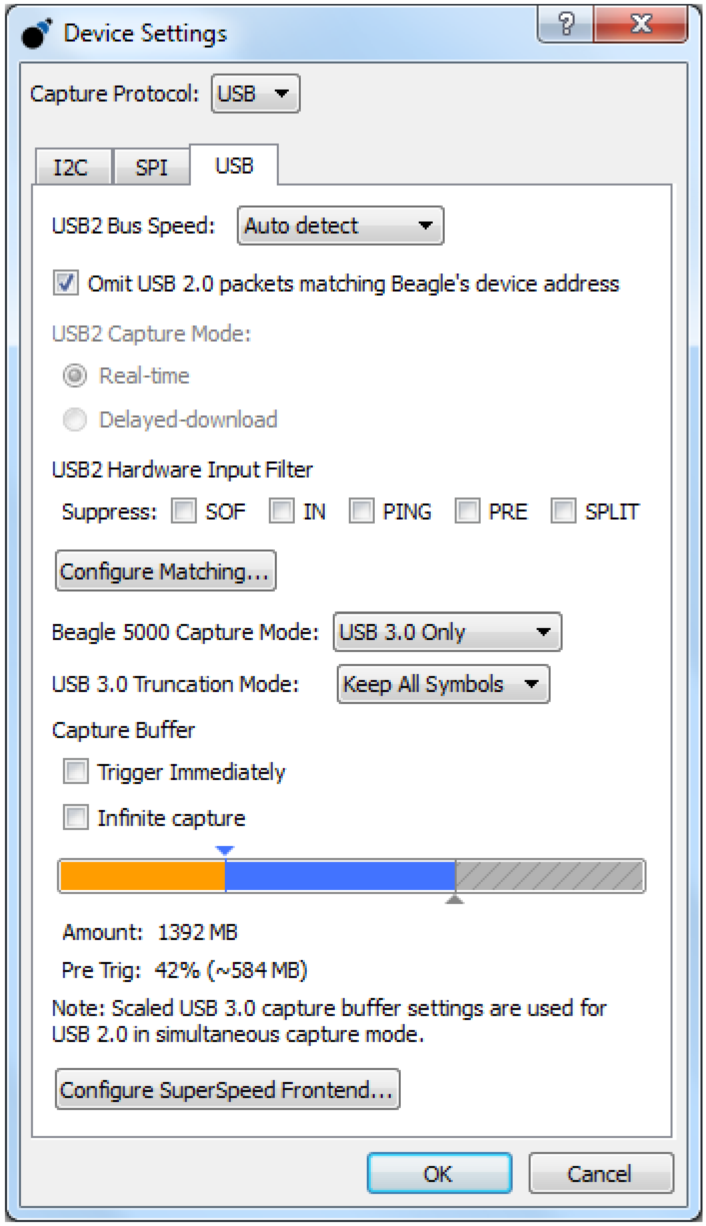



***Figure 47:*** *USB Device Settings Dialog*

One method to avoid flooding the capture with traffic for the Beagle analyzer is to enable the **Omit packets matching Beagle's device address** option. This option instructs the Beagle USB 480 analyzer to discard any packets directed to its own device address. Further information about this option may be found in Device Operation section of the Beagle datasheet.

Do not enable this option if the Beagle USB 480 analyzer is not on the same host controller or if it is not on the same computer as the traffic being monitored. This may cause the Beagle analyzer to discard USB packets intended for another device since device addresses across different USB buses can overlap. Furthermore, do this only if you are monitoring a High-speed device with the Beagle 480 Analyzer, as High-speed devices and Full-speed or Low-speed devices on the same host may also have overlapping addresses.

Do not enable this option when using a Beagle USB 5000 analyzer. USB 3.0 hosts are required to have an integrated USB 2.0 host which means that by default the analyzer will be monitoring a separate host controller.

#### USB2 Capture Mode

The Beagle USB 480 analyzer has two modes of operation during capture.

- **Real-time** – The capture is streamed from the Beagle USB 480 analyzer to the Data Center software as it is received. If the hardware is monitoring traffic faster than it can stream data to the analysis PC, it will cache the data in the analyzer's hardware buffer. If the hardware buffer should fill completely, the capture will be stopped.
- **Delayed-download** – The Beagle analyzer hardware will not stream the capture data to the analysis PC until the capture is concluded. The capture will be stored in the analyzer's hardware buffer until it is filled or the user stops the capture to download the results. This feature is useful if the Beagle analyzer is connected to the same host controller as the traffic being monitored, as this greatly reduces the amount of USB traffic generated by the Beagle analyzer while the capture is taking place. More information about the utility of a delayed-download capture can be found in the Beagle datasheet (Device Operation section). More details on running a delayed-download capture can be found in Section 6.4.

#### USB2 Hardware Input Filter

These options will enable the Beagle USB 480/5000 analyzer hardware to discard some common packet groups to reduce the amount of capture data received. Some of these packet groups correspond to polling operations and so these sequences do not contain any actual data transfer. When using the Beagle USB 5000 analyzer, these settings are only applicable when capturing USB 2.0 data.

Note: If there is a change in the digital input lines in the middle of one of the packet groups that is being filtered, that group will not be discarded. In this way, the context for the digital input line change is preserved.

The hardware filter options are:

- **SOF** – Discard Start-of-Frame packets.
- **IN** – Discard IN+ACK and IN+NAK packet groups.
- **PING** – Discard PING+NAK packet groups.
- **PRE** – Discard all PRE tokens.
- **SPLIT** – Enabling this option will cause the hardware to discard many polling split packet groups. The split groups that will be discarded are:
  - SSPLIT+IN

- SSPLIT+IN+ACK
- CSPLIT+IN+NAK
- CSPLIT+IN+NYET
- CSPLIT+OUT+NYET
- CSPLIT+SETUP+NYET

### Configure Matching

The digital inputs and outputs are configured in the USB 2.0/USB 3.0 matching system.

### USB 2.0

All USB 2.0 digital inputs and outputs for the Beagle USB 480 analyzer and Beagle USB 5000 analyzer can be configured in a single window (Figure 48).

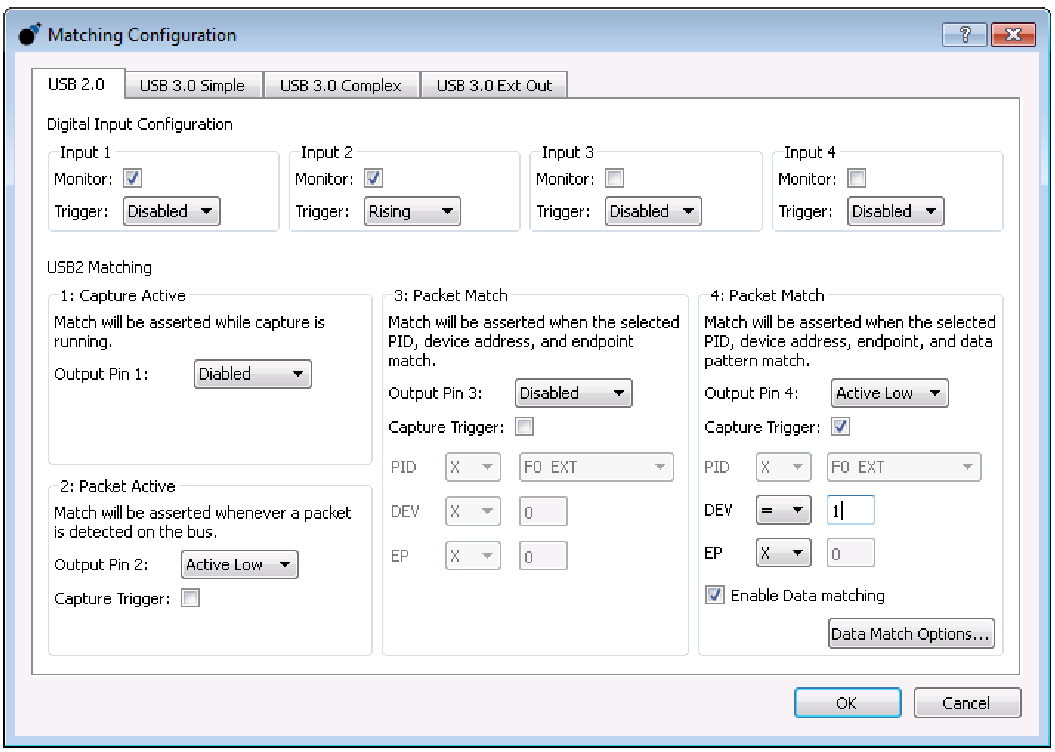



***Figure 48:*** *USB 2.0 Matching Configuration Dialog*

### Digital Input Configuration

This option enables the user to individually enable the four digital input lines on the Beagle USB 480/5000 analyzer hardware. During the capture, if there is a change on one of the enabled input lines, an event transaction will be displayed in the Transaction window with the new input line state.

When using the Beagle USB 5000 analyzer, there is an additional option to set individual digital inputs to trigger the capture. When configuring the trigger, the user can select whether a rising edge, falling edge, or either should trigger the capture.

### USB2 Matching

The lower half of the USB 2.0 matching window provides the capabilities to configure the digital output lines on the Beagle USB 480/5000 analyzer hardware. See the **Device Operation** section of the Beagle analyzer datasheet for information on the timing of the digital output pins.

#### Pin 1

When enabled, digital output pin 1 will switch to its active state at the beginning of capture and will stay at the active level until the capture is stopped. The options available on this tab are to set pin 1 to be active high, active low, or to disable the pin.

#### Pin 2

When enabled, digital output pin 2 will switch to its active state whenever there is a USB packet being transmitted on the bus. The options available on this tab are to set pin 2 to be active high, active low, or to disable the pin.

#### Pin 3

Digital output pin 3 can be set to match a PID, a device address, and an endpoint address. For example, pin 3 can go active when it observes a DATA0 packet to any device but 0x01 on any endpoint address. Each of the match settings can be set to match if the packet equals the parameter (**=**), does not equal the parameter (**!=**), or it can disregard the parameter (**X**). Pin 3 can also be set to be active high, active low, or to be disabled.

#### Pin 4

Digital output pin 4 can be set to match a PID, a device address, an endpoint address, and a data payload pattern. As with Pin 3, the PID, device, and endpoint settings can be set to match if the packet equals the parameter (**=**), does not equal the parameter (**!=**), or it can disregard the parameter (**X**). The data pattern to match can be set in the **Data Matching Options** once data matching is enabled. Pin 4 can also be set to be active high, active low, or to be disabled.

#### Data Matching Options

Clicking the **Data Match Options** button in the Pin 4 tab of the Digital Output Settings dialog will reveal the data match options (Figure 49). Checking the **Enable Data Matching** option will enable the data match functionality.

Note: PID matching will be disabled when data pattern matching is enabled. The data PIDs selected in the **PIDs to match** section will be used instead.

The data match pattern for output pin 4 can be specified in the hex editor. **XX** can be used anywhere in the pattern as a wildcard to match any 1-byte datum at that location. The data match pattern can be up to 1024 bytes in length.

When data pattern matching is enabled, every data packet with the specified PID will have its data payload (not including the PID or the CRC field) matched against as much of the pattern as

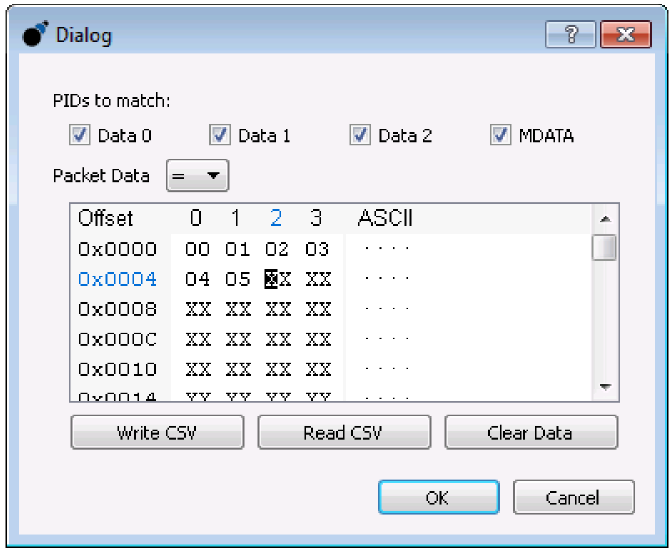



***Figure 49:*** *Digital Output Pin 4 Data Tab*

the data payload size. So a data packet with an 8-byte data payload will be compared against the first 8 bytes of the data match pattern. And a data packet with a 64-byte payload will be compared against the first 64 bytes of the data pattern.

Using the pulldown box above the hex editor, the packet can be set to match if the payload equals the pattern (**=**) or does not equal (**!=**) the pattern.

The **Write CSV** button allows the contents of the data match hex editor to be written to a comma-separated values (*.csv) file. The CSV file can be loaded into the data match table by clicking the **Read CSV** button. The **Clear Data** button will clear the contents of the data match table.

### USB 3.0 Simple

In the **USB 3.0 Simple** tab (Figure 50), the simple matching systems is capable of triggering the USB 3.0 capture or asserting the external output on the following types of data or events in either the upstream or downstream direction:

- Link Command
- Header Packet
- Data Payload
- CRC Error
- Training Sequences
- VBUS Detection
- External Input
- Reverse Polarity

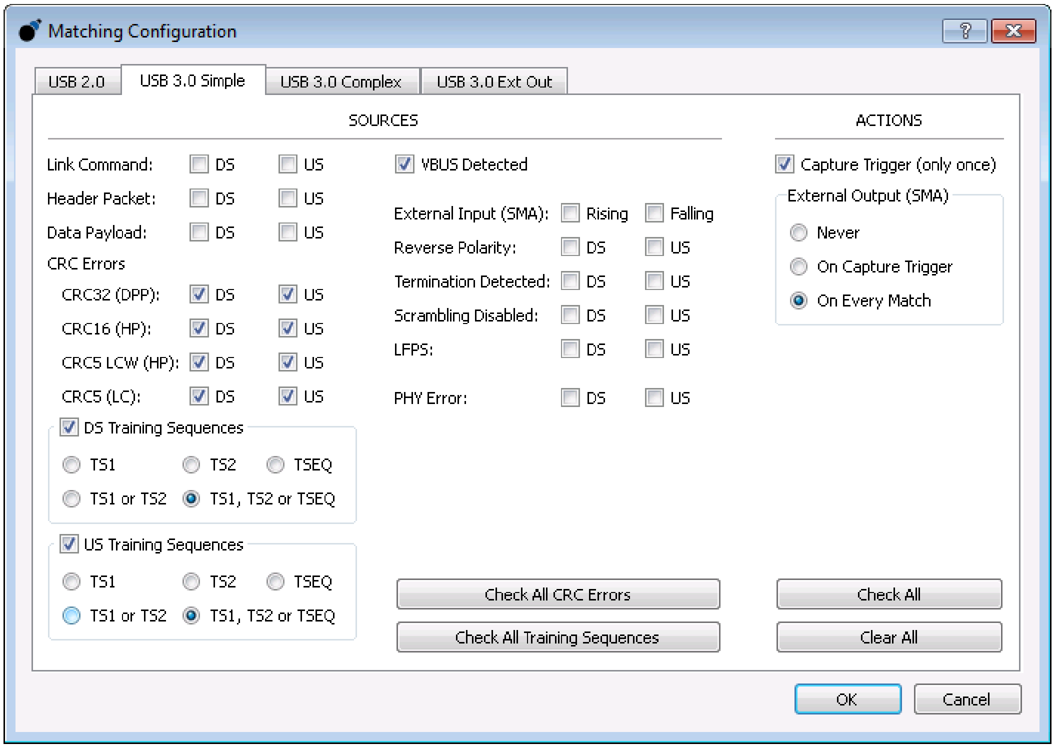



***Figure 50:*** *USB 3.0 Simple Matching Configuration Dialog*

- Termination Detection
- Scrambling Disable
- LFPS
- PHY Error

The simple match is designed to be very broad to provide a quick method to set triggers to capture data. For example, selecting Header Packet will match whenever a Link Management Packet, Transaction Packet, Data Payload Packet, or Isochronous Timestamp Packet occurs on the bus.

When a match occurs, the user can configure the analyzer to trigger and/or assert the external output. The external output can be asserted on the capture trigger or on every time the match occurs.

**WARNING: The USB 3.0 Digital Input and Output are only rated for 1.8V.** The USB 3.0 input and output of the Beagle USB 5000 analyzer have been optimized for maximum edge performance at 125 MHz. **Applying signals with higher voltage will damage your analyzer and void the warranty.**

Details about the different types of packets and events are covered in the Beagle analyzer datasheet.

As a convenience, common configurations can be set with a click of a button. These configurations are:

- Check All CRC Errors
- Check All Training Sequences
- Check All
- Clear All

### USB 3.0 Complex

The **USB 3.0 Complex** tab (Figure 51) provides additional capabilities beyond the USB 3.0 Simple options to match a specific packet type or data pattern in addition to bus events and timers.

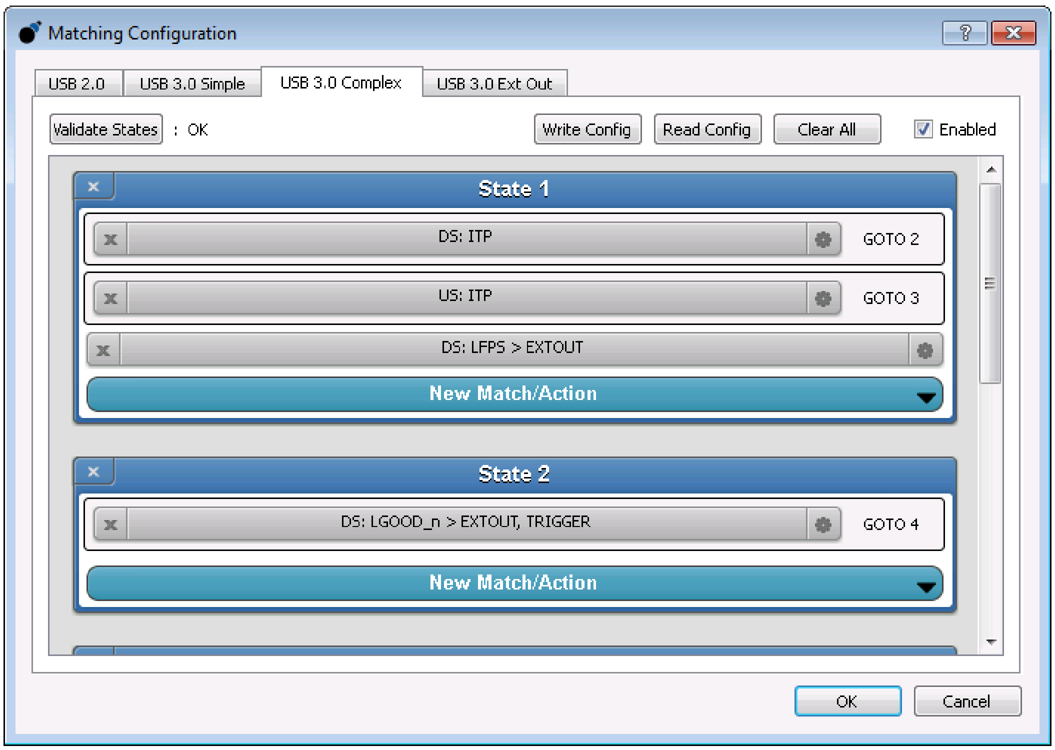



***Figure 51:*** *USB 3.0 Complex Matching Configuration Dialog*

Instead of an array of check boxes, complex matches are built through states. The Standard Beagle USB 5000 analyzer allows for a single state to be defined. The optional advanced matching upgrade allows for up to eight states to be defined.

Within any state, match action units are specified to define the match conditions which can trigger an action. Matches can be based on packet type, packet data, events, and/or timers. The Standard Beagle USB 5000 analyzer provides an upstream data match, a downstream data match, and an event match. The optional advanced matching upgrade allows for up to three upstream data matches, up to three downstream data matches, an event match, and a timer per state. The actual number of matches available in any one state will depend on the resources available.

Each match action unit defines one or more actions when the match conditions are met. The available actions are: assert external output, trigger capture, filter out the matching data, or go to another state (including the current state). These actions provide flexibility to define very complex scenarios to capture specific USB 3.0 traffic.

The top of the USB 3.0 Complex Match tab has the following functions:

**Validate States**

The Validate States button will do a simple verification of the state configuration to make sure that there are no obvious errors. If the configuration is valid, "OK" will appear next to the button. If the configuration is invalid, an error message will appear next to the button (Figure 52).


***Figure 52:*** *USB 3.0 Complex Matching - Invalid state*

**Write Config and Read Config**

Users are able to save and load complicated matching for later use. To save a complex state match, click **Write Config**. A file dialog will open so that the user can save a CSM file with the configuration. To load a match, click **Read Config**, and find the desired CSM file with the file dialog. The CSM file is a text based file that can be edited with a text editor.

**Clear All**

**Clear All** will delete all complex state match data. When the button is clicked, a modal dialog will appear to confirm the deletion of all match data.

**Enabled**

The **Enabled** checkbox indicates whether the complex matching system is enabled. Simply click the checkbox to activate the interface to define a complex state match.

**State Configuration**

Each state in the complex state match is delimited by a graphical box. The box is identified by the state number. By default the first state is created when creating a new complex match.

To create a new state, simply click the **New State** button. Only a single state can be defined when using a standard Beagle USB 5000 analyzer. Up to eight states can be defined with an optional upgrade to the advanced matching.

A state can be deleted by clicking on the button at the top left of a state window. A modal dialog will open when the button is clicked to verify that the deletion of the state.

To change the order of the states, click and drag a state box. A dotted box will appear where the state will be moved to and the other states will reorder themselves (Figure 53). Once the state is released, all other states will renumber themselves accordingly. Any existing "GOTO" command will be updated with the new state number.

**Match Action Units**

Within each state, one or more match action units can be defined by clicking on the **New Match Action** button. When clicked, the available match action units will appear as a drop down menu

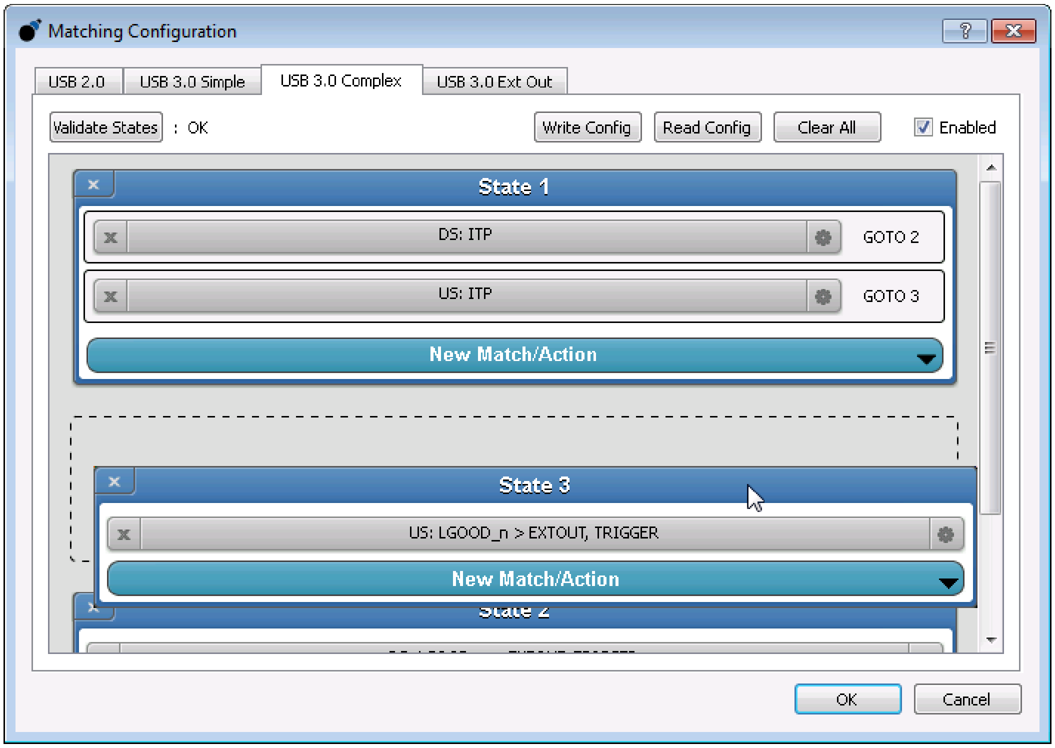



***Figure 53:*** *USB 3.0 Complex Matching - Reordering States*

(Figure 54). A Match Action Configuration window will open specific to the type of match action unit selected. If a match action unit is not available, it will be grayed out in the menu.


***Figure 54:*** *USB 3.0 Complex Matching - Creating a new match action unit*

After a match action unit is defined, a match action unit block will appear in the state with a textual description of the match criteria and action settings. If a "Go To" branching action has been defined, the "GOTO" parameters will appear next to the match action unit.

To change the settings of a match action unit, click the Settings button, this will open the match action configuration window. To delete a defined match action unit, click on the delete button. A modal dialog will open to confirm the deletion.

Each state can have a number of match action units. If multiple matches occur simultaneously, all non-GOTO actions will execute before the GOTO executes and the state changes. If multiple matches occur simultaneous that have GOTO actions, the order of the GOTO blocks in the state determine what the next state will be. For example, in Fig 53, if a downstream ITP and upstream ITP occur at the same time, the GOTO 2 from the downstream ITP will execute since it comes before all other matching GOTOs in the state.

To change the order of the match action units, click and drag a match action unit. A dotted box will appear where the match action unit will be moved to and the other match action units will reorder themselves accordingly (Figure 55).

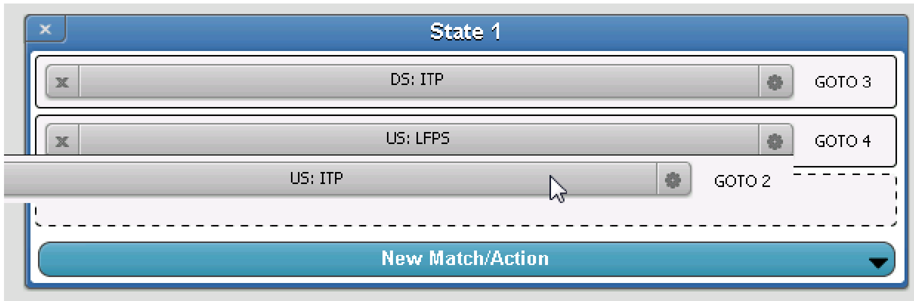



***Figure 55:*** *USB 3.0 Complex Matching - Reordering a match action unit*

### Match Action Unit Configuration

The different match action units have different configuration windows to suit their properties. In general there are three types of match action units: data match, event match, and timer match. The timer match is only available in the advanced matching system.

### Available Actions

All match action units can be configured to execute one or more actions: assert digital output, capture the trigger, or go to another state.

With the advanced matching upgrade, counters are built-in to all match action units. They can be configured in one of two ways (Figure 56): execute the action after the match has occurred a specified number of times ("Apply action on or after X matches") or execute for a set number of matches ("Apply action on the first X matches").


***Figure 56:*** *USB 3.0 Complex Matching - Match counter*

### Event Match Action Unit

The Event Match Action Unit (Figure 57) matches USB 3.0 events similar to what is available in the USB 3.0 Simple configuration.

The following events in the upstream or downstream direction can be selected: LPFS, Polarity Inversion, Receiver (RX) Termination, or Disable Scrambling. VBUS Presence and External Input can also be selected independent of stream direction. The match can be specified on the rising or falling edge.

### Timer Match Action Unit

The Timer Match Action Unit (Figure 58) matches after a specified amount of time has elapsed.

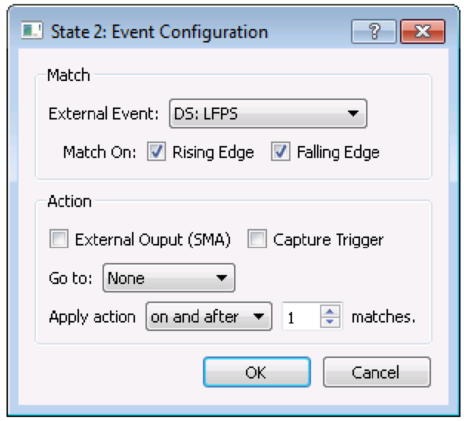



***Figure 57:*** *Event Match Action Unit*

The amount of time can be configured in seconds, milliseconds, microseconds, or nanoseconds.

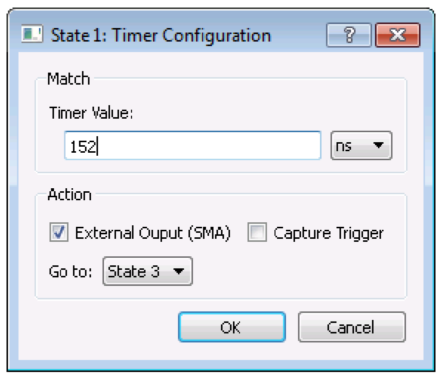



***Figure 58:*** *Timer Match Action Unit*

### Data Match Action Unit

The Data Match Action Unit is the most versatile and is configured for specifically the upstream (US) or downstream (DS) direction. A Data match action unit can be configured for any one of the following packet types: Link Command, Header Packet, Data Packet, Qualified Data Packet, and Training Sequence.

All Data match action units provide the ability to match the opposite of the selection. In this way, a match action unit can be configured to match **X** or **NOT X**.

It is important to note that data match action units are defined for a common class of packets and the negation only occurs within that type (Figure 59). For example, selecting "Any Link Command of a different Link Type" will only match any link command that is not LGOOD_0 (e.g. LGOOD_1, LCRD_A, etc.). If "Any packet besides this Link Type" is selected, the unit will match not only on every other link command, but also any packet which is not a link command (e.g. Header Packet or Data Packet Payload).

### Link Command Match Action Unit

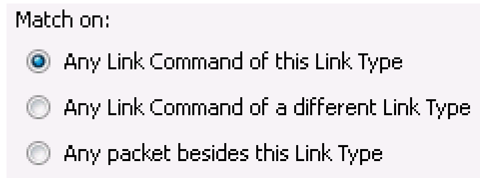



***Figure 59:*** *Data Match Action Unit - Negative Criteria*

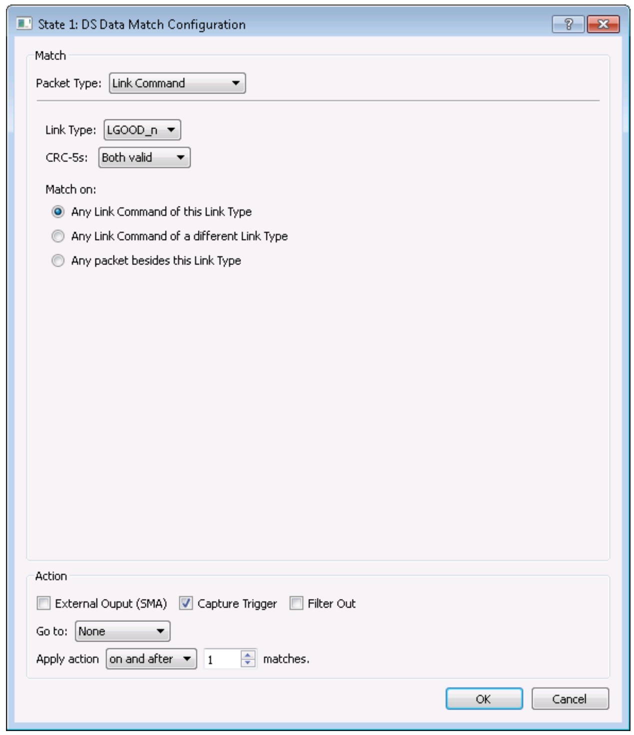



***Figure 60:*** *Link Command Data Match Action Unit*

The Link Command match action unit (Figure 60) provides the ability to match a specific link command. Checks for the validity of the CRCs can also be set.

**Header Packet Match Action Unit**

The Header Packet match action unit (Figure 61) provides the ability to set a specific data pattern to match in the header packet. The fields of the header packet follow the USB 3.0 specification. The fields available will depend on the type of header packet that is specified. The data pattern must be entered in binary format with a X indicating that the specific bit does not need to be matched.

Fields marked with a black triangle in the bottom right corner are configured with presets to facilitate data entry. By clicking on the field, a drop-down menu will appear listing the available options. Please note that it is not necessary to use the drop-down menu, a bit pattern can be specified directly.

The fields available in the interface will update on the fly to match the type of header packet selected.

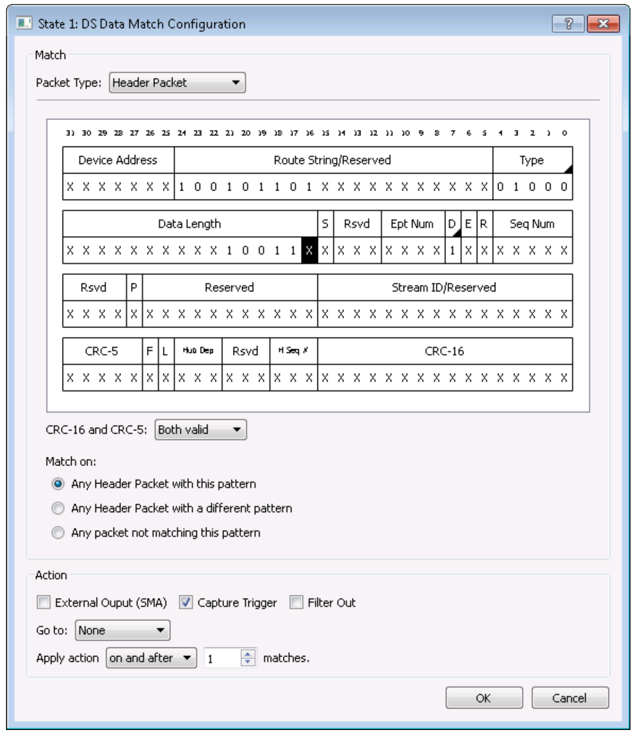



***Figure 61:*** *Header Packet Data Match Action Unit*

### Data Packet

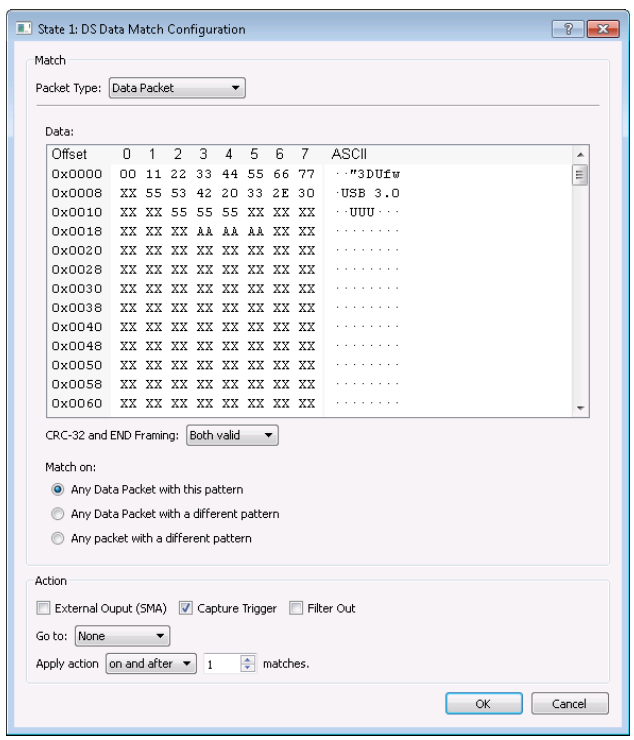



***Figure 62:*** *Data Packet Data Match Action Unit*

The Data Packet match action unit (Figure 62) can match a data pattern in the data payload. The hex editor interface for entering the data payload pattern is flexible. The user can enter data in hex, ASCII, binary, decimal, or octal. Right-clicking on the interface brings up a contextual menu that allows the user to change the input style, number of panes, radix, etc.

The data match pattern can be specified in the hex editor. **XX** can be used anywhere in the pattern as a wildcard to match any 1-byte datum at that location. The data match pattern can be up to 1024 bytes in length.

When data pattern matching is enabled, every data packet will have its data payload matched against as much of the pattern as the data payload size. So a data packet with an 8-byte data payload will be compared against the first 8 bytes of the data match pattern. And a data packet with a 64-byte payload will be compared against the first 64 bytes of the data pattern.

The CRC-32 and END Framing combo box can be specified to match a data packet which ends in a valid CRC-32 and END framing (“Both valid”), an invalid CRC-32 or EDB framing (“Either invalid”), or to ignore these fields entirely (“Both ignored”).

### Qualified Data Packet

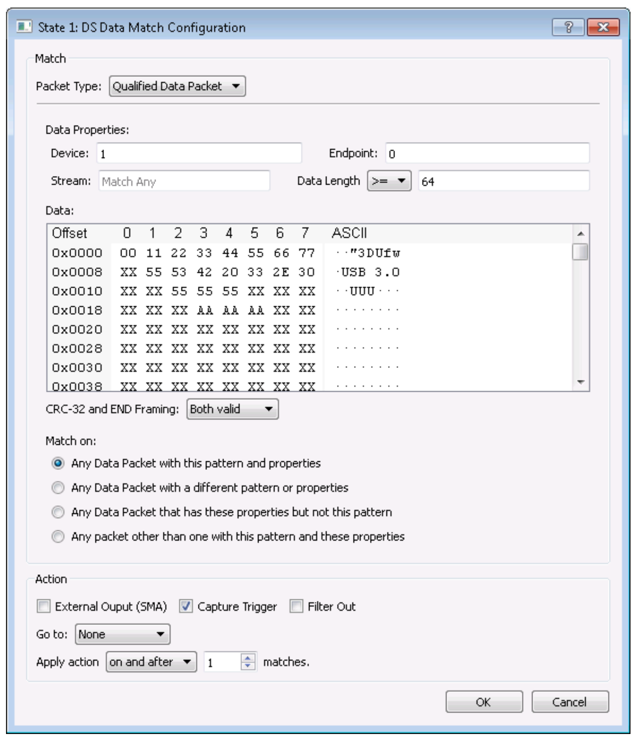



***Figure 63:*** *Qualified Data Packet Data Match Action Unit*

The Qualified Data Packet match action unit (Figure 63) provides the same capabilities as the Data Packet match, but also provides the ability to specify additional parameters such as device, endpoint, stream ID, and data length.

### Training Sequences

The Training Sequence match action unit (Figure 64) can match a TS1, TS2, or TSEQ packet.

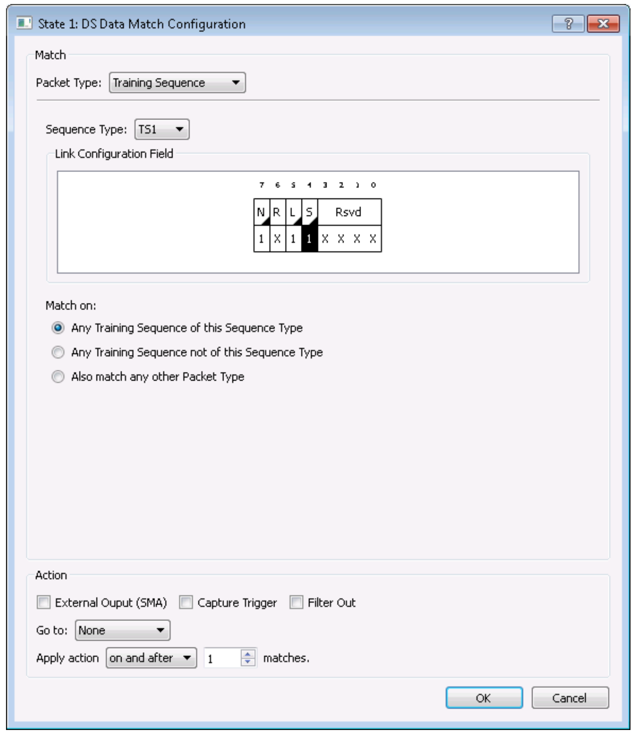



***Figure 64:*** *Train Sequence Data Match Action Unit*

For TS1 and TS2, the data pattern for the Link Configuration Field can be supplied.

**Error**

The Error match action unit (Figure 65) can match on any packet type which exhibits an error. The errors which can be matched are a CRC error, Framing error, or any unknown packet. Matches of this type cannot be filtered.

**5 Gbit Transmission**

The 5 Gbit Transmission match action unit (Figure 66) can match on the start or stop of 5 Gbit transmission. Matches of this type cannot be filtered.

### USB 3.0 Ext Out

The behavior of the USB External output (Figure 67) can be configured for the Beagle USB 5000 analyzer. The external output can be set to one of the following behaviors when it is asserted:

- Set Low
- Set High
- Positive Pulse
- Negative Pulse

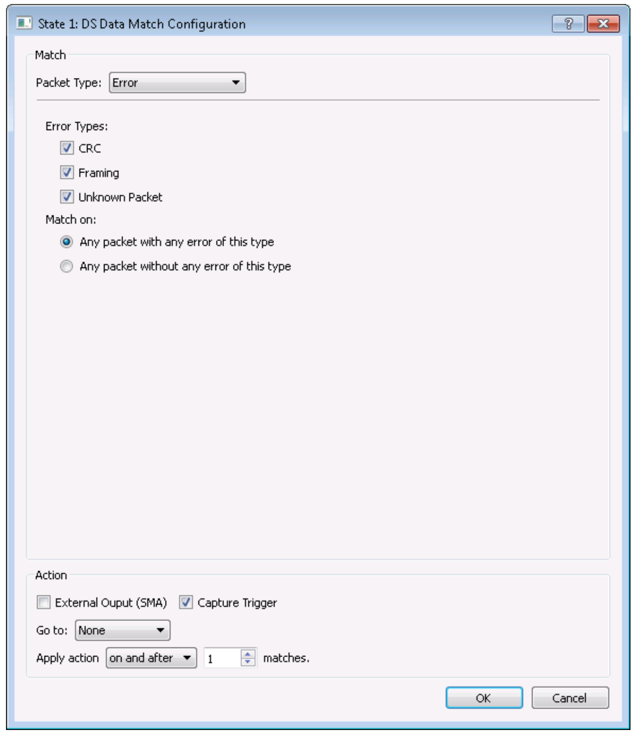



***Figure 65:*** *USB 3.0 Error Data Match Action Unit*

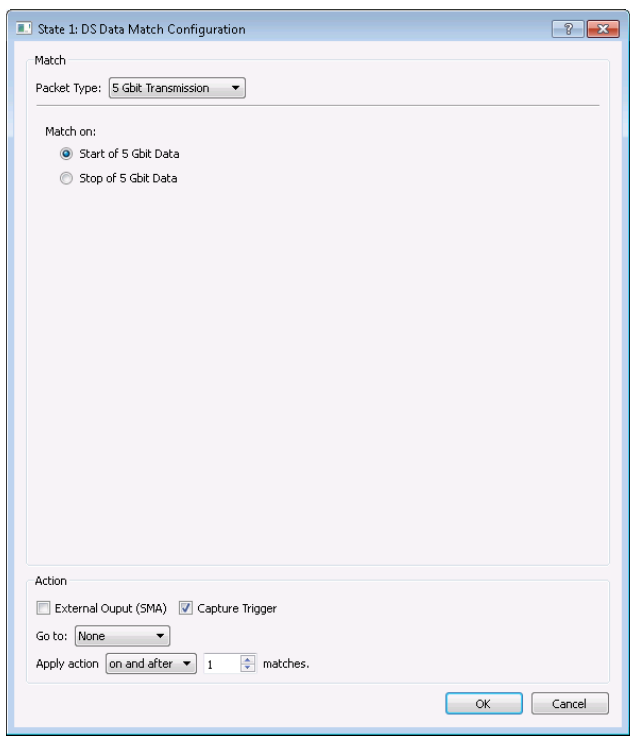



***Figure 66:*** *USB 3.0 5 Gbit Transmission Data Match Action Unit*

- Toggle (Initially Low)

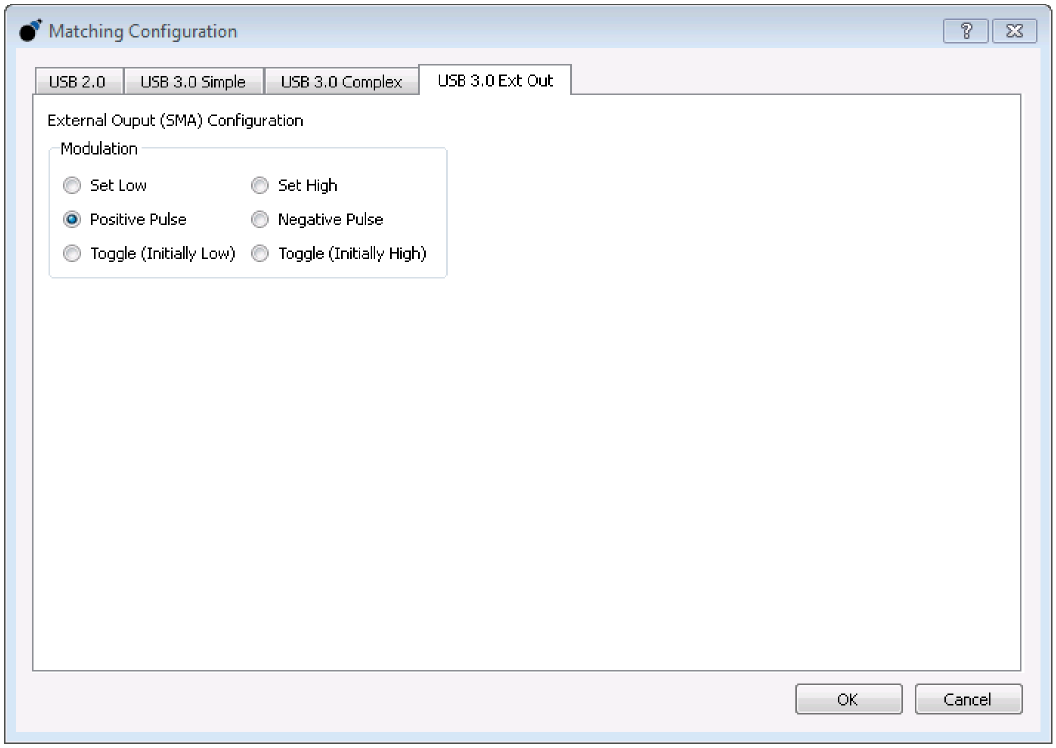



***Figure 67:*** *USB 3.0 External Out Configuration*

- Toggle (Initially High)

**WARNING: The USB 3.0 Digital Input and Output are only rated for 1.8V.** The USB 3.0 input and output of the Beagle USB 5000 analyzer have been optimized for maximum edge performance at 125 MHz. **Applying signals with higher voltage will damage your analyzer and void the warranty.**

### USB 3.0 Capture Settings

#### Beagle 5000 Capture Mode

In USB 3.0 systems, a backwards compatible USB 2.0 system exists in parallel. The two protocols essentially operate as two separate channels.

The Beagle USB 5000 analyzer is able to capture USB 3.0 and/or USB 2.0 data. In the standard analyzer, either protocol can be captured. With an optional upgrade, the analyzer is able to capture both USB 3.0 and USB 2.0 simultaneously.

From the pull-down menu, users can select: USB 3.0, USB 2.0, or both.

#### USB 3.0 Truncation Mode

The Beagle USB 5000 analyzer can optionally truncate incoming packets to 20 symbols, 36 symbols, or 60 symbols. These truncation lengths include the packet framing (4 symbols), and thus provide a means for capturing 16 symbols, 32 symbols, or 64 symbols after the packet framing.

Even if truncation is enabled, match units will still be able to compare against the full length of the packet. Additionally, the Data Center software will still provide the user with the true length of the packet transmitted on the bus.

Enabling packet truncation will limit the available class-level decoding to Mass Storage, UASP, and Bulk Transfer Grouping. Within these parsers there may be additional limitations due to the loss of data.

Note that if the configuration descriptor is truncated, class-level decoding cannot be automatically applied. These can be applied after the capture has been stopped by using the Configuration Management feature of Data Center. See Section 6.10 for more details.

**Capture Buffer**

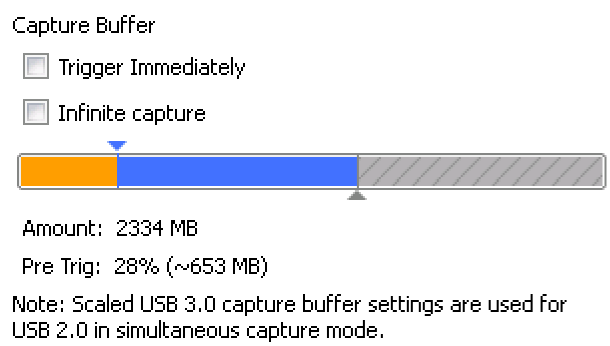



***Figure 68:*** *USB 3.0 Capture Buffer Settings*

With the Beagle USB 5000 analyzer, it is possible to limit the total size of the capture. The orange and blue bar represents the total amount of capture memory available on the analyzer (Figure 68). The orange area represents the amount of memory allocated to the pre-trigger capture. The blue area represents the amount of memory allocated to the post-trigger capture. By dragging the triangles located above and below the bar, the amount of memory used for the capture can be allocated. Not all the memory available on the analyzer needs to be used.

There is also the option of an **infinite capture** (Figure 69). Since the Beagle analyzer is constantly streaming data to the Analysis PC, it is possible for the analyzer to free up memory on the analyzer during the capture. This freed memory can be reused to capture more data. As long as the Analysis PC is able to keep up with the stream of data from the analyzer, it is theoretically possible to capture data infinitely. However, at a certain point, the memory on the analysis machine will be used up and the capture will need to be stopped.

It is also important to note that the pre-trigger buffer will operate as a circular buffer until the trigger occurs. The Beagle analyzer will not download the data to the Analysis computer until the trigger event occurs. During this time, the portion of the memory allocated to the pre-trigger capture will be reused until the trigger occurs. Once the trigger happens, the capture buffer will be sent to the Analysis PC. It is important to note that if the pre-trigger buffer is not large enough, events could be lost from the early part of the capture.

Optionally, the capture can be immediately triggered once the capture has started by checking the **Trigger Immediately** checkbox.

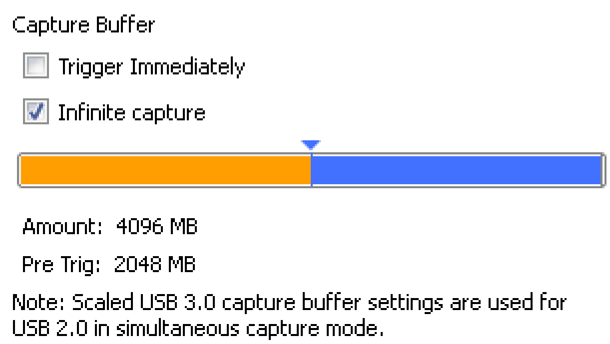



***Figure 69:*** *USB 3.0 Infinite Capture Buffer Setting*

### Front-end Settings

Given the speeds of USB 3.0, it is not possible to passively monitor the USB 3.0 bus. Consequently, the USB 3.0 data stream needs to be regenerated to send to the target receiver. It is important to note that the latency of this regeneration is only 1 ns and that the USB 3.0 signal is not retimed.

Since the signal is regenerated, it is possible to modify some aspects of the signal. The Frontend Settings dialog (Figure 70) allows the user to configure this behavior.

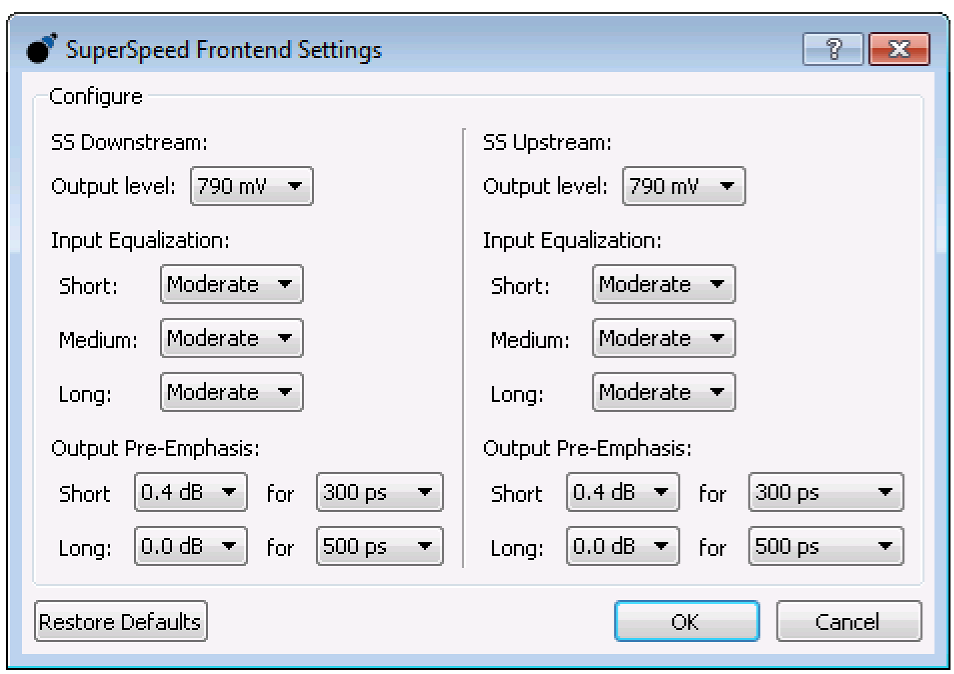



***Figure 70:*** *USB 3.0 Frontend Settings*

There are three different properties that can be set in either the upstream or downstream direction.

Output level is the signal levels sent by the transmitter in the Beagle USB 5000 analyzer. By lowering the output signal level, it is possible to test the sensitivity of the USB 3.0 receiver.

Input Equalization is used to correct for signal degradation due to transmission through a lossy channel. The equalization is broken up into three stages (short, medium, and long), which represent the size of the discontinuity causing the degradation. The equalization settings can correct for minimum, moderate, or maximum amount of degradation or can be turned off. For the most part, the default settings are adequate for most scenarios and do not need to be changed.

Output Pre-Emphasis is used to boost the signal sent by the transmitter to compensate for degradation as the signal is sent to the receiver. Pre-emphasis is broken up into two stages (short and long), which represent the size of the discontinuity expected to cause degradation. For each stage, the level (in dB) and the decay (in ps) of boost can be configured. For the most part, the default settings are adequate for most scenarios and do not need to be changed.

## 6.4 Delayed-Download Capture

This mode is only available with the Beagle USB 480 analyzer. In this capture mode, the capture data is not streamed out of the analysis port of the Beagle analyzer until after the analyzer has stopped monitoring the bus. This greatly reduces the amount of USB traffic going to the Beagle USB 480 analyzer while the capture is active, and thus is primarily useful when the Beagle analyzer and the test device share the same host controller. Please refer to the **Device Operation** section of the Beagle datasheet for more information regarding the delayed-download mode of the Beagle analyzer.

### Performing a Delayed-Download Capture

To run a delayed-download capture, select the Delayed-Download Capture Mode option in the Device Settings dialog. During the delayed-download capture, there will still be a small amount of Beagle USB 480 analyzer traffic on the capture bus since the software pings the analyzer to retrieve capture statistics. Therefore, if the monitored device is High-speed and shares its host controller with the Beagle Analyzer, it is advisable to enable the **Omit packets matching Beagle's device address** option to filter out the few Beagle packets that will remain during the delayed-download capture. In addition, because the capture will be stored in the Beagle analyzer's hardware buffer during the capture, enabling the **Hardware input filter** options may be useful to allow for a longer capture by preventing non-essential traffic from being saved in the hardware buffer.

Once the capture settings have been set, click the **Run Capture** button to open the **Delayed-Download Capture** dialog (Figure 71).

Set the polling interval for the capture. When polling during the capture, the Data Center software will check the hardware buffer usage and display it in the progress bar. The polling will generate traffic on the bus, so polling can be disabled to eliminate this traffic by choosing **Never** for the polling interval.

Click the **Start Capture** button to start the capture. If polling is enabled, the progress bar will show the portion of the hardware buffer that has been filled with capture data. The progress bar will be updated every time the Data Center software polls the Beagle USB 480 analyzer. When the hardware buffer is full, the capture will stop and the dialog will say that it is ready to download the capture from the hardware.

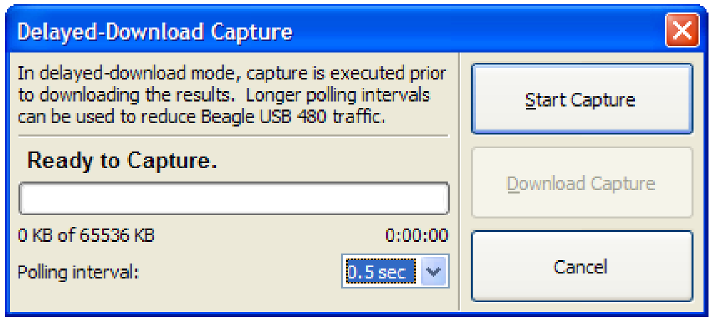



***Figure 71:*** *Delayed-Download Capture dialog*

If polling is disabled, the only way to know that the hardware buffer has filled and capture has stopped is by observing that the red activity LED on the Beagle Analyzer is no longer blinking and has turned off.

You may click the **Download Capture** button at any point to download the results. This will stop the capture if it had not already stopped. Once the download begins, the Delayed-Download Capture dialog will automatically close.

The **Cancel** button may be clicked at any time to exit the delayed-download process and close the dialog. This will completely discard any data that has been captured.

## 6.5 Capture Control Window

The Capture Control Window provides both control of a capture and visibility into the current state of the capture.

For the Beagle USB 12 analyzer and Beagle USB 480 analyzer, the Capture Control window behaves as described in the General overview Section 5.7.

The Capture Control window for a Beagle USB 5000 analyzer (Figure 72) is more complex and has three progress bars. The “USB 3” progress bar displays the status of the USB 3.0 hardware memory buffer on the Beagle analyzer. The “USB 2” progress bar displays the status of the USB 2.0 hardware memory buffer on the Beagle analyzer. The “Software Capture Buffer” progress bar displays the status of the software capture buffer as described in the General overview.

The Beagle USB 5000 analyzer has two hardware buffers. A 2 GB buffer (upgradable to 4 GB) for USB 3.0 and a separate 128 MB buffer for USB 2.0. The USB 2.0 bus and USB 3.0 bus are essentially treated as two separate channels since they operate exclusively and in parallel. If the Beagle USB 5000 analyzer is configured to only capture USB 3.0 data, the “USB 2” progress bar will be disabled and vice versa. Both the “USB 3” and “USB 2” progress bars behave similarly.

Similar to the Device Settings Dialog, the entire progress bars represents the total amount of memory available. White areas indicate the amount of memory allocated for the pre- and post-trigger buffers. Gray areas indicates memory that is not available or not in use. When memory is used for captured, the white areas are filled with orange (pre-trigger data) or blue (post-trigger data). The amount of memory allocated for pre- and post-trigger is defined in the

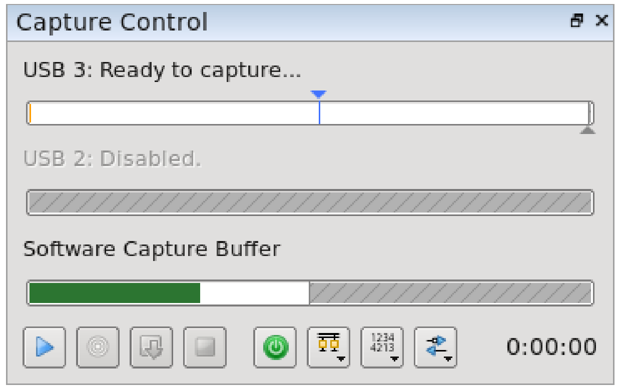



***Figure 72:*** *Capture Control Window for the Beagle USB 5000 SuperSpeed Protocol Analyzer*

Device Settings. If the capture is set to be infinite, the entire status bar is used. If the amount of capture buffer is limited in the Device Settings dialog, the unused memory is filled with gray.

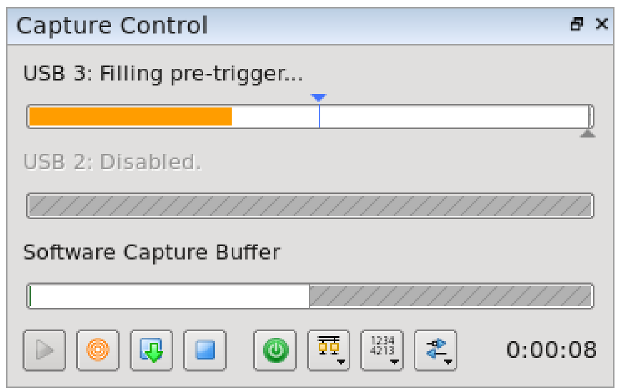



***Figure 73:*** *Capture Control Window for the Beagle USB 5000 SuperSpeed Protocol Analyzer - filling the pre-trigger buffer*

Once a capture starts, the pre-trigger buffers will fill and the orange bar in the progress bar will grow (Figure 73). The pre-trigger buffer is a circular buffer and will only fill up to the limit set in the Device Settings.

When the capture trigger occurs, a number of things happen simultaneously (Figure 74). The pre-trigger buffer will stop being filled, the post-trigger buffer will start being filled, and the analyzer will start streaming the data to the Analysis PC. Since the pre-trigger buffer has stopped the orange bar will stop growing, and any remaining white areas will be filled with gray to indicate that the pre-trigger memory is no longer available since the pre-trigger capture is complete. The orange buffer will be replaced with gray as the pre-trigger data is streamed off the analyzer. The pre-trigger capture is complete when all the orange has been replaced by gray, indicating that all the data has been downloaded to the Analysis PC.

While the pre-trigger data is downloading, the post-trigger buffer will fill and the blue bar in the progress bars will grow (Figure 75). Once all the pre-trigger data has been downloaded, the post-trigger data will start being streamed off which will result in the blue buffer being replaced with gray. The post-trigger will continue to fill until the capture reaches its set limit, the Analysis

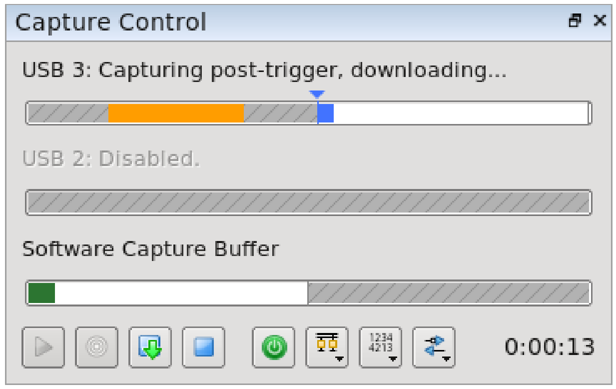



*Figure 74: Capture Control Window for the Beagle USB 5000 SuperSpeed Protocol Analyzer - downloading the pre-trigger buffer while filling the post-trigger buffer.*

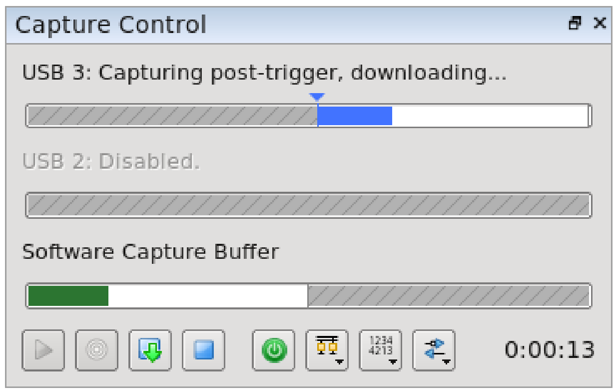



*Figure 75: Capture Control Window for the Beagle USB 5000 SuperSpeed Protocol Analyzer - pre-trigger buffer has been completely downloaded while the post-trigger is being filled*

PC runs out of memory, or the user stops the capture (Figure 76). Once the capture stops, any remaining white areas will be replaced with gray. The remaining data will download off the analyzer until the progress bars are completely gray, indicating that all data has been downloaded from the buffers.

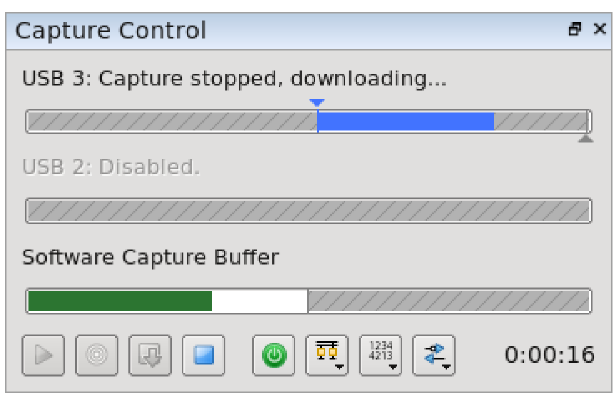



*Figure 76: Capture Control Window for the Beagle USB 5000 SuperSpeed Protocol Analyzer - the capture has been stopped and the post-trigger buffer continues to be downloaded.*

Once data starts downloading to the Analysis PC, the “Software Capture Buffer” will start to fill with green as described in the General Section.

Additional controls for the Beagle USB 5000 analyzer are available in the Capture Control window.

**Target Power** 

Use the **Target Power** button to toggle whether VBUS is passed to the target device or not. This has the same effect as pressing the Target Power button on the front of the Analyzer. When VBUS is being passed to the target device, the button will be green. When VBUS is not passed through, the button will be red.

**Receiver Termination** 

Use the **Receiver Termination** button to configure the receiver detection system. When the button is clicked, a pull-down menu provides the ability to set automatic receiver detection or to force receiver termination to be either on or off in either the upstream (UP) direction, downstream (DS) direction, or both directions. This button is only active when using a Beagle USB 5000 analyzer to capture USB 3.0 data.

When set to **Auto**, the Beagle USB 5000 analyzer will automatically detect the receiver termination as described in the datasheet. When set to **Force On**, the lines presented to the transmitter by the analyzer will always be terminated, regardless of the state of the receiver termination. When set to **Force Off**, the lines presented to the transmitter by the analyzer will not be terminated, regardless of the state of the receiver termination.

**Data Scrambling** 

Use the **Data Scrambling** button to configure the data scrambling detection. When the button is clicked, a pull-down menu provides the ability to set automatic data scrambling detection or to force data scrambling to be either on or off in either the upstream (UP) direction, downstream (DS) direction, or both directions. This button is only active when using a Beagle USB 5000 analyzer to capture USB 3.0 data.

When set to **Auto**, the Beagle USB 5000 analyzer will automatically detect the polarity settings of the USB 3.0 link being monitored during the training sequence. When set to **Force On**, the Beagle USB 5000 analyzer will always display data as if the data has been scrambled. When set to **Force Off**, the Beagle USB 5000 analyzer will always display data as if the data has not been scrambled.

**Polarity Detection** 

Use the **Polarity Detection** button to configure the polarity detection. When the button is clicked, a pull-down menu provides the ability to set automatic polarity detection or to force polarity to be either inverted or non-inverted in either the upstream (UP) direction, downstream (DS) direction, or both directions. This button is only active when using a Beagle USB 5000 analyzer to capture USB 3.0 data.

When set to **Auto**, the Beagle USB 5000 analyzer will automatically detect the polarity settings of the USB 3.0 link being monitored during the training sequence. When set to **Force Inverted**, the Beagle USB 5000 analyzer will always display data as if the polarity is inverted. When set to **Force Non-Inverted**, the Beagle USB 5000 analyzer will always display data as if the polarity is not inverted.

## 6.6 Transaction Window

The Transaction window (Figure 77) displays all the transactions as they were captured on the USB bus in real time.

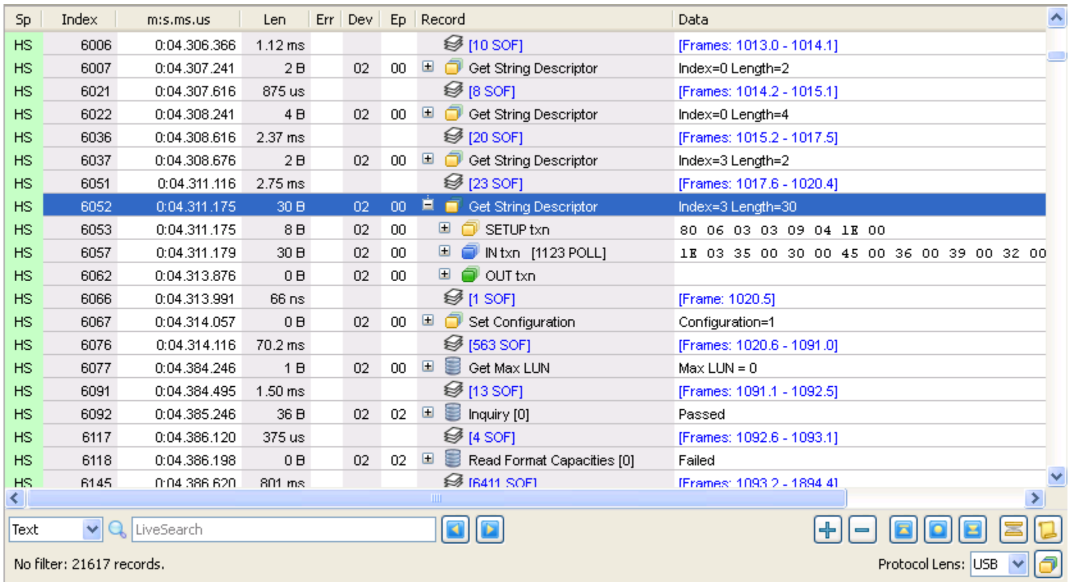



***Figure 77:*** *USB Transaction Window*

For a general description of the Transaction window, see Section 5.1. The following describes the specifics of the USB Transaction window.

**Speed (Sp)**

The bus speed of the transaction (Beagle USB 480/5000 analyzer only). The background color of the column will also indicate the bus speed. The possible values displayed are shown in Table 6. SuperSpeed transactions have an arrow next to the SS that specifies the channel on which the transaction occurred (Upstream or Downstream) or have a double-headed arrow for SuperSpeed collapsed transactions that contain traffic from both streams. The HS, FS, LS, and LF transactions always have a double-headed arrow.

***Table 6:*** *USB speed column values*

| Value | Meaning | Background color |
|---|---|---|
| SS | SuperSpeed | Blue |
| HS | High-speed | Green |
| FS | Full-speed | Yellow |
| LS | Low-speed | Red |
| LF | Low-speed over full-speed | Yellow |

**Length (Len)**

The length of the transaction in bytes is shown if the transaction has a byte value. If the transaction doesn't have a byte value, such as bus events, the duration is shown instead.

**Error codes (Err)**

Error codes listing abnormal conditions that occurred while capturing the transaction. See Table 2 for the possible error codes. In addition, there are several USB specific error codes as described in Table 7.

***Table 7:** USB-specific error code values*

| Code | Meaning | Description |
|---|---|---|
| 0 | Bad bit-stuff | The Beagle USB 12 analyzer has detected a bit-stuff error. |
| B | Bad signals | Invalid signal observed on the bus. With the Beagle USB 480/5000, this could be caused by a misaligned bit-stuff error. |
| C | Bad CRC | The CRC of the packet is invalid. |
| F | SOF/ITP frame number error | Unexpected frame number encountered. This could be caused by a discontinuity in the frame number sequence, a repeated frame number in full-speed, or greater or fewer than 8 repeated frame numbers in high-speed. |
| H | Invalid SPLIT bits | Certain bit patterns of the SPLIT packet are not allowed by the USB 2.0 Specification. Please refer to the USB 2.0 Specification section 8.4.2.2 for more information. |
| I | Invalid PID sequence | An invalid sequence of packets has been observed. |
| K | Classification error | An error occurred during class-level parsing. |
| L | Improper packet length | The packet has a length that is too large or too small for the packet's PID type. |
| Y | Unexpected PING | A PING token was seen but is unexpected. A PING token is expected to be seen only after one of these transactions: a OUT-DATA-NYET, a OUT-DATA-NAK, or a PING-NAK. |
| Z | Frame timing jitter | The frame was observed outside of the acceptable timing specification. Please refer to USB 2.0 Specification section 7.1.12 for the particular timing specifications. |
| S | Sequence error | For SuperSpeed, a Header's sequence number, a Link Good, or a Link Credit is not consecutive. |
| R | Stream error | For a SuperSpeed streaming bulk endpoint, the Transaction Packets or Data Packets did not follow the expected bulk streaming protocol. |
| E | EDB Framing | A SuperSpeed Data Packet Payload terminated with EDB framing. |
| G | Framing error | A symbol was corrupted on a SuperSpeed packet frame. |

**Dev**

The device being addressed represented as a decimal value.

**Ep**

The endpoint being addressed represented as a decimal value.

**Data**

For individual packets, the **Data** column will show the raw data bytes including the PID and CRC. For some transactions, such as SETUP transactions, the **Data** column may show a parsed representation of its corresponding packet data. For other transactions, such as IN transactions, the **Data** column will show its internal packet data without any PID or CRC.

### USB 2.0 Transaction Groups

USB packets are grouped into transactions when they are detected on the USB bus. The four transaction groups are IN, OUT, SETUP, and LPM. Each transaction group is denoted with a unique icon and can be expanded to reveal the individual packets. The timestamp of the group will match the timestamp of the first item in the group to appear on the bus.

Polling transactions that do not have a data payload, such as IN/NAK or PING/NAK, will also be included in the related transaction group. Figure 78 shows an example of this where the IN/NAKs associated with the IN transaction have been included in the IN group.

| | | | | |
|---|---|---|---|---|
| 0:05.248.965 | 83 ns | | | [1 SOF] |
| 0:05.249.026 | 13 B | 01 | 01 | ⊟ IN txn [3 POLL] |
| 0:05.249.026 | 5.30 us | 01 | 01 | [3 IN-NAK] |
| 0:05.249.033 | 3 B | 01 | 01 | IN packet |
| 0:05.249.034 | 16 B | 01 | 01 | DATA1 packet |
| 0:05.249.035 | 1 B | 01 | 01 | ACK packet |
| 0:05.249.090 | 83 ns | | | [1 SOF] |
| 0:05.249.185 | 31 B | 01 | 01 | ⊞ OUT txn |

***Figure 78:*** *IN/NAK collapsed packets included in IN transaction*

As a result of grouping the polling transactions into higher level transaction groups, there may be situations where packets are shown out of chronological order. Figure 79 shows an example of this. The collapsed PING/NAK group at index 5604 has an earlier timestamp than the collapsed SOF group at index 5602. Digital input transactions may also appear out of order since they are not included in any transaction group.

| | | | | | |
|---|---|---|---|---|---|
| 5598 | 0:05.364.175 | 512 B | 01 | 01 | ⊞ OUT txn (NAK) |
| 5602 | 0:05.364.231 | 125 us | | | [2 SOF] |
| 5603 | 0:05.364.187 | 512 B | 01 | 01 | ⊟ OUT txn [83 POLL] |
| 5604 | 0:05.364.187 | 193 us | 01 | 01 | [83 PING-NAK] |
| 5605 | 0:05.364.381 | | 01 | 01 | ⊞ PING-ACK |
| 5608 | 0:05.364.385 | 3 B | 01 | 01 | OUT packet |
| 5609 | 0:05.364.386 | 515 B | 01 | 01 | DATA0 packet |
| 5610 | 0:05.364.395 | 1 B | 01 | 01 | ACK packet |
| 5611 | 0:05.364.395 | 512 B | 01 | 01 | ⊞ OUT txn (NAK) |

***Figure 79:*** *PING/NAK out of order grouping*

Consecutive isochronous, or interrupt, transactions that are related may be grouped together as shown in Figure 80.

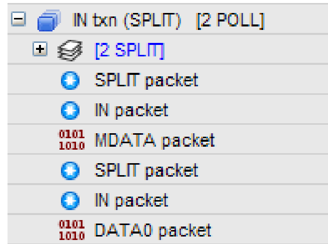



***Figure 80:*** *Grouped isochronous SPLIT transactions*

### USB 3.0 Transaction Groups

While at the lowest level, USB 3.0 data is made up of packets rather than tokens, the transaction grouping in the Data Center software follows the same convention as USB 2.0 data: IN, OUT, and SETUP transactions. Each transaction group is denoted with a unique icon and can be expanded to reveal the individual packets. The timestamp of the group will match the timestamp of the first item in the group to appear on the bus.

### Special Transaction Types

Besides the four main transaction groups, there are several different types of transactions which can appear in the Transaction window.

#### Information

Start/Stop informational transactions indicate when a capture was started or stopped. These transactions appear in blue text.

#### Events

Event transactions represent non-packet bus activity, such as host connect, target device connect, bus reset, or bus speed events. These transactions are displayed as a text description of the event that occurred with further information available in the Info pane. These transactions appear in the Transaction window in green text. Please refer to the **Device Operation** section of the Beagle datasheet for more information on specifics of bus events and their timings.

The types of USB 2.0 event transactions are:

- **Host connected** – USB cable connected to upstream port.
- **Host disconnected** – USB cable disconnected from upstream port or voltage level dropped below detection threshold.
- **Target connected** – USB cable connected to downstream port.
- **Target disconnected** – USB cable disconnected from downstream port or voltage level dropped below detection threshold.

- **Reset** – Bus put into reset state.
- **Sync error** – Bad sync observed on packet. (Beagle USB 12 only)
- **Low-speed** – The bus is operating at low-speed. (Beagle USB 480/5000 analyzer only)
- **Full-speed** – The bus is operating at full-speed. (Beagle USB 480/5000 analyzer only)
- **High-speed** – The bus is operating at high-speed. (Beagle USB 480/5000 analyzer only)
- **Suspend** – The bus has entered suspend state. (Beagle USB 480/5000 analyzer only)
- **Resume** – The bus has left suspend state. (Beagle USB 480/5000 analyzer only)
- **Keep-alive** – Low-speed keep-alive strobe detected. This signal is used by the host to keep low-speed devices from going into suspend mode. (Beagle USB 480/5000 analyzer only)
- **Chirp J** – A high-speed chirp J was observed. This signal is part of the High-speed Detection Handshake used by high-speed devices to transition from full-speed to high-speed. (Beagle USB 480/5000 analyzer only)
- **Chirp K** – A high-speed chirp K was observed. This signal is part of the High-speed Detection Handshake used by high-speed devices to transition from full-speed to high-speed. (Beagle USB 480/5000 analyzer only)
- **Tiny J** – A false J caused by a voltage divider effect between the device pulling up the D+ line with a 1.5K resistor and the host not driving the data line to ground with a sufficiently low enough output resistance. (Beagle USB 480/5000 analyzer only)
- **Tiny K** – A false K caused by a voltage divider effect between the device pulling up the D- line with a 1.5K resistor and the host not driving the data line to ground with a sufficiently low enough output resistance. (Beagle USB 480/5000 analyzer only)
- **Input line change** – Voltage change on one or more of the input lines detected. (Beagle USB 480/5000 analyzer only)
- **OTG HNP** – An On-The-Go Host Negotiation Protocol was detected. (Beagle USB 480/5000 analyzer only)
- **OTG SRP data-line pulse** – A data-line pulse of the On-The-Go Session Request Protocol was detected. (Beagle USB 480/5000 analyzer only)
- **OTG SRP Vbus pulse** – A Vbus pulse of the On-The-Go Session Request Protocol was detected. (Beagle USB 480/5000 analyzer only)

The types of USB 3.0 event transactions are:

- **VBUS Present** – $V_{BUS}$ detected between the target host and the target device.
- **VBUS Absent** – $V_{BUS}$ not detected between the target host and the target device.
- **SuperSpeed Target Connected** – USB 3.0 cable connected to the downstream port

- **SuperSpeed Target Disconnected** – USB 3.0 cable disconnected from the downstream port
- **SuperSpeed Host Connected** – USB 3.0 cable connected to the upstream port
- **SuperSpeed Host Disconnected** – USB 3.0 cable disconnected from the upstream port
- **Manual Trigger or USB2 Trigger** – USB 2.0 capture triggered
- **Manual Trigger or USB3 Trigger** – USB 3.0 capture triggered
- **LTSSM Transitions**

**Collapsed**

There are common packet sequences that are repeated frequently on the USB bus which can quickly fill up a capture and make it difficult to find the data of interest. In order to reduce this problem, the Data Center software will automatically "collapse" these sequence of packets into a single row. Packets will only be collapsed together if they share the same device and endpoint. Some of these collapsed transactions may appear in transaction groups.

These collapsed packets will collapse the following types of data.

**SOF**

Start-of-Frame. These packets are issued once every millisecond in full-speed and every 125 microseconds in high-speed to keep devices synchronized with the host.

**Keep-alive**

Low-speed keep-alive strobe. This signal is used by the host to keep low-speed devices from going into suspend mode. (Beagle USB 480/5000 analyzer only)

**IN/NAK**

Some USB devices require the host to periodically poll the device to see if any changes occurred. The host will issue an IN packet and if the device has no changes, it will send a NAK. This sequence of packets can quickly eat up capture space when a device is idle and is therefore collapsed.

**IN/ACK**

When an IN/DATA/ACK occurs on a parallel USB link, only the IN and the ACK will be observed by the Beagle analyzer. Therefore, this packet group is collapsed.

**PRE/IN, PRE/IN/NAK, PRE/IN/PRE/ACK**

When a host communicates to a low-speed device through a full-speed hub, the host must send the hub a PRE packet before every packet to the low-speed device. This alerts the hub that the packet that follows the PRE will be transmitted at the low-speed data rate. This is called low-speed over full-speed.

These packet groups are similar to IN/NAK and IN/ACK, so are similarly collapsed.

**PING/NAK**

PING packets are used in high-speed traffic to poll if a device is ready to receive data. The NAK packet indicates that the device is not yet ready to receive more data.

**SPLIT transactions**

Split transactions are used by the host to communicate with a full- or low-speed device through a high-speed hub. (Beagle USB 480/5000 analyzer only)

In a typical situation, the host will send a START-SPLIT packet (SSPLIT) to the hub. The split packet will contain flags indicating which port to send the following packets to and what speed to send them at. Then the host will send the token packet (IN, OUT, or SETUP) to send to the full- or low-speed device. For OUT and SETUP transactions, a data packet will follow. Then the hub may or may not send an ACK to the host, depending on the transfer type.

The hub will then transmit the packets to the downstream device at the requested bus speed. The host will then periodically poll the hub to see if the hub has completed the transaction and to get the response of the device. The host does this by sending the hub a complete-split (CSPLIT) packet followed by the same token packet it sent earlier. The hub will then either respond with a NYET to indicate that it is not yet done sending the transaction, or it will respond with the device's response (data or NAK for IN tokens, a handshake packet for OUT and SETUP tokens).

Since there can be a good deal of polling with split transactions, including using IN packets to poll the downstream device, the following packet groups are collapsed into SPLIT groups:

- `SSPLIT/IN` will be shown as `[ START ]`
- `SSPLIT/IN/ACK` will be shown as `[ START ]`
- `CSPLIT/IN/NYET` will be shown as `[ NYET ]`
- `CSPLIT/IN/NAK` will be shown as `[ NAK ]`

The following packets groups are collapsed inline with their associated transaction group:

- `CSPLIT/OUT/NYET`
- `CSPLIT/SETUP/NYET`

**Orphaned packets**

It is common to observe orphaned packets when multiple devices are plugged into the same host controller, or if a USB hub is present on the bus.

Orphaned transactions are transactions which the Beagle analyzer only sees a portion of because the target device is on a different branch of the USB tree. Since all messages from the host are broadcast throughout the entire bus, the Beagle analyzer will only see one half of the conversation.

Consider the bus topology in Figure 81.

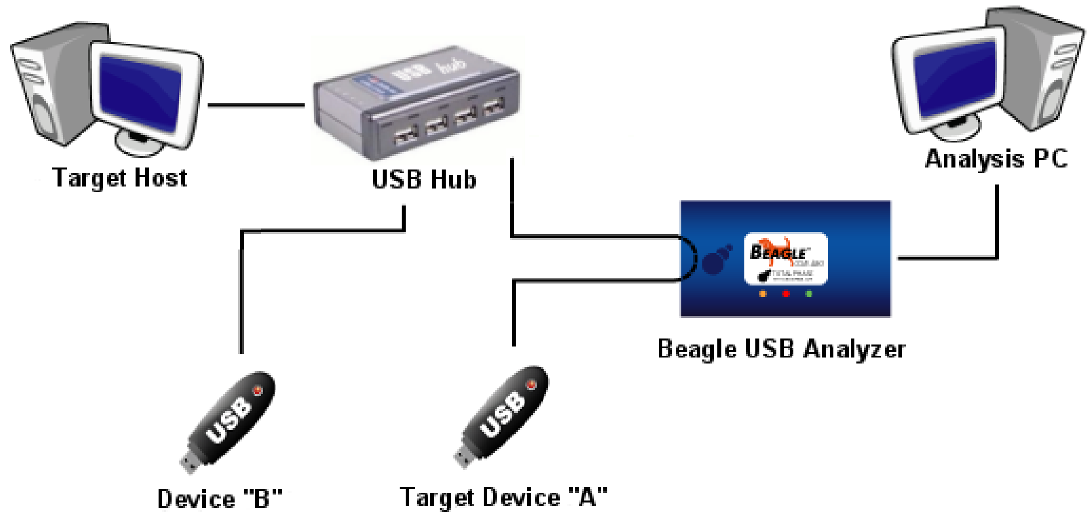



***Figure 81:*** *USB topology which causes orphaned packets*

In this configuration, the Beagle sees all traffic between the host and device A, but only downstream traffic from the host to device B. For example, if the Host sends an IN to Device B and Device B NAKs the IN, the Beagle will have only seen the IN. We consider such packets to be orphaned and group them accordingly in the Data Center software.

Note that a hub retransmits packets destined for itself. So, even if device B were not present in Figure 81, the Beagle would still see orphaned packets from the communication between the hub and the host.

Even if the hub were not present, the Beagle would observe orphaned packets if devices A and B were connected to the same host controller.

**Training Sequences**

Training sequences packets: TSEQ, TS1, and TS2 are repeated many times when a USB 3.0 device is first plugged in to establish a communication link between the transmitter and the receiver. It is not uncommon for there to be a significant number of PHY errors during the training sequence while the link is being established. Once the link is establish, PHY Errors should be uncommon.

**LFPS**

Low-Frequencey Periodic Signaling packets is a lower frequency packet sent on the SuperSpeed data lines to manage signal initiation and low power management on a link between two ports.

## 6.7 Capture View

Three unique capture views are available in Data Center when using a Beagle USB 5000 or Beagle USB 480 Protocol Analyzer. Two capture views are available when using a Beagle USB 12 Protocol Analyzer. Please note that Packet View is not available when using the Data Center software with the Beagle USB 12 Protocol Analyzer. To select a Capture View, use the Capture View Menu in the Transaction Window Controls section of the application (Section 4.3).

- **Packet** – Protocol-level decoding is performed, and records are in time-order. Collapsed groups, such as SOFs and IN-NAKs are broken up as necessary to ensure records are in timestamp order. **Only captures run in Sequential Mode** (Section 6.2) **can be viewed in Packet View**.
- **Transaction** – Protocol-level decoding is performed. Records may **not** be in time-order. Collapsed groups are not broken up for time-order preservation. Since there is no time-order restriction, captures generally appear more compact in this view than in Packet View.
- **Class** – Class-level parsing is performed. Records may **not** be in time-order. Captures generally appear high-level and compact in this view. For more information regarding class-level parsing, see Section 6.9.

## 6.8 LTSSM View

Track the progression of top-level SuperSpeed LTSSM transitions during a capture using the Data Center software's interactive LTSSM diagram (Figure 82). The view is accessible via  in the application toolbar.

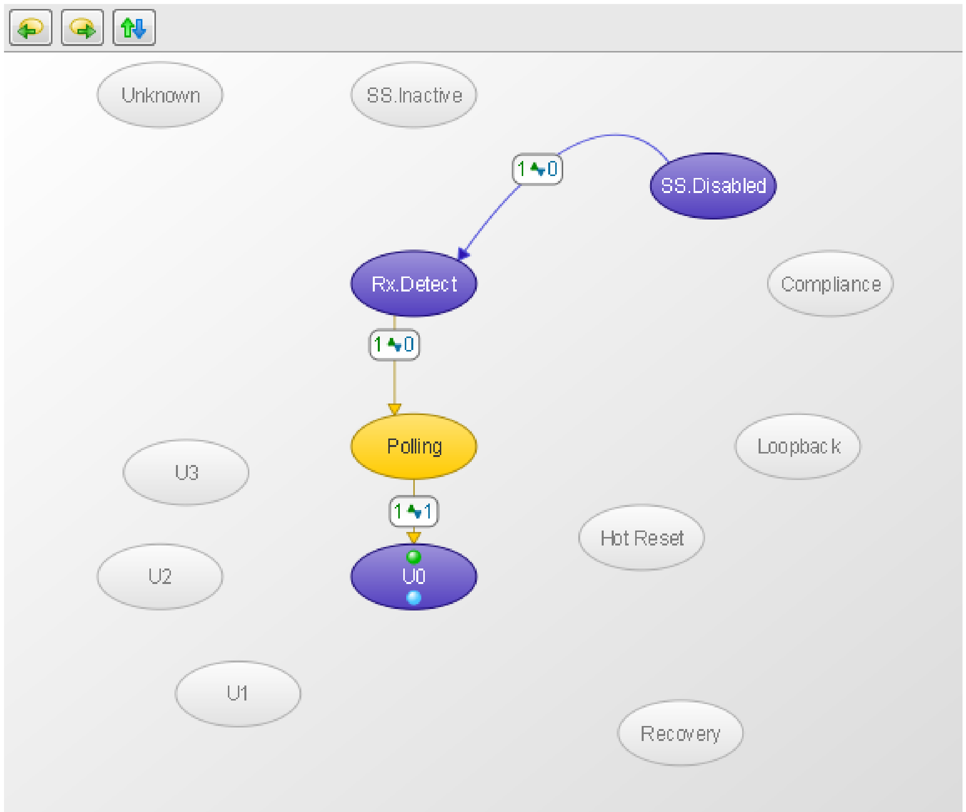
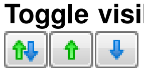



***Figure 82:*** *LTSSM View of a capture that entered four link states: SS.Disabled, Rx.Detect, Polling, and U0. Green and blue arrows on the task bar indicate visibility of both upstream and downstream state transitions. Gold highlights on Polling and adjoining arrows show that the currently selected record (not pictured) occurred while the bus was in that state, having entered it from Rx.Detect and subsequently exited towards U0. Also pictured: the count of each transition that occurred, both upstream (green) and downstream (blue). Following the same color code, the final captured states for both streams are decorated with spherical markers above and below the label (in this case, on U0).*

### Interacting with the LTSSM View

Controls in the LTSSM View enable filtering of which transitions should be represented on the diagram as well as LTSSM-centric traversal of the record list.

- **Toggle visibility of upstream and downstream state transitions** in the diagram with  on the LTSSM View toolbar.
- **Jump to the previous or next state transition** relative to the selected record with  and  on the LTSSM View toolbar, respectively.

- **Cycle through occurrences of a particular state** by *hovering* over the state ellipse of interest and *rolling the mouse wheel*. Rolling upwards navigates to the previous instance of the state relative to the currently selected record, and rolling downwards to the next instance.
  - Alternatively, repeatedly *click* the state to cycle through matching transition records, starting at the first.
- **Cycle through occurrences of a particular transition** by *hovering* over the counter embedded on the transition arrow of interest (e.g. ) and either *rolling the mouse wheel* or *clicking*. The behavior is identical to navigating states (see previous bullet point).
- **Jump to the first/previous/next/last instance of a state or transition** by *right-clicking* on either the state ellipse or transition counter of interest and *selecting* the desired option from the context menu.
- **View all valid transitions from a state according to the USB3 spec** by *hovering* the mouse pointer over the state of interest. After a short delay, any transitions originating from that state that weren't already visible will appear in faint gray.

## 6.9 Class-Level Parsing

Some USB hosts and devices may communicate with one another using device classes. The Data Center software supports parsing of these device classes. Further information about device classes may be found in the USB Background section of the Beagle datasheet.

### General Use

Class-level parsing can be enabled by selecting Class View in the Capture View menu (Section 4.3). For class-level parsing to work correctly, it is necessary for the Data Center software to capture the enumeration of the USB device. The easiest way to ensure that the enumeration is captured is to first start the capture and then to plug in the device into the analyzer. It is not possible to enable or disable classification while the capture is running. The enumeration is preserved across capture sessions. This may lead to unintended behavior if different devices share the same device address across the capture sessions.

Post-capture, It is possible to apply a configuration to a device in order to see its data parsed at the class level. More information about applying and managing configurations can be found in Section 6.10.

Note that when using circular buffer or saving a filtered view the class-level parsing ability may be lost. For more information please see the section on circular buffer (Section 4.14) and saving a capture (Section 4.10).

Also, note that the Beagle USB 12 Analyzer supports parsing of the Standard Device Requests only. Class specific parsing is not supported with the Beagle USB 12 Analyzer.

### Class-Level Transactions

The benefit of enabling classification is that class-level fields are visible for each transfer. One or more protocol-level transactions will be grouped into a class-level transaction. Additionally, for some classes, a class-level transaction may contain one or more other class-level transactions. The transaction will display information that is relevant to the device class. For instance, with a mass storage device, a class-level transaction may be named "Read", and the Data column will show the logical block address and length of the transfer. Additional information can be obtained by clicking on the class-level transaction and looking at the Info pane. The Info pane will show the parsed fields of the selected class transaction. See Figure 83 for an example.

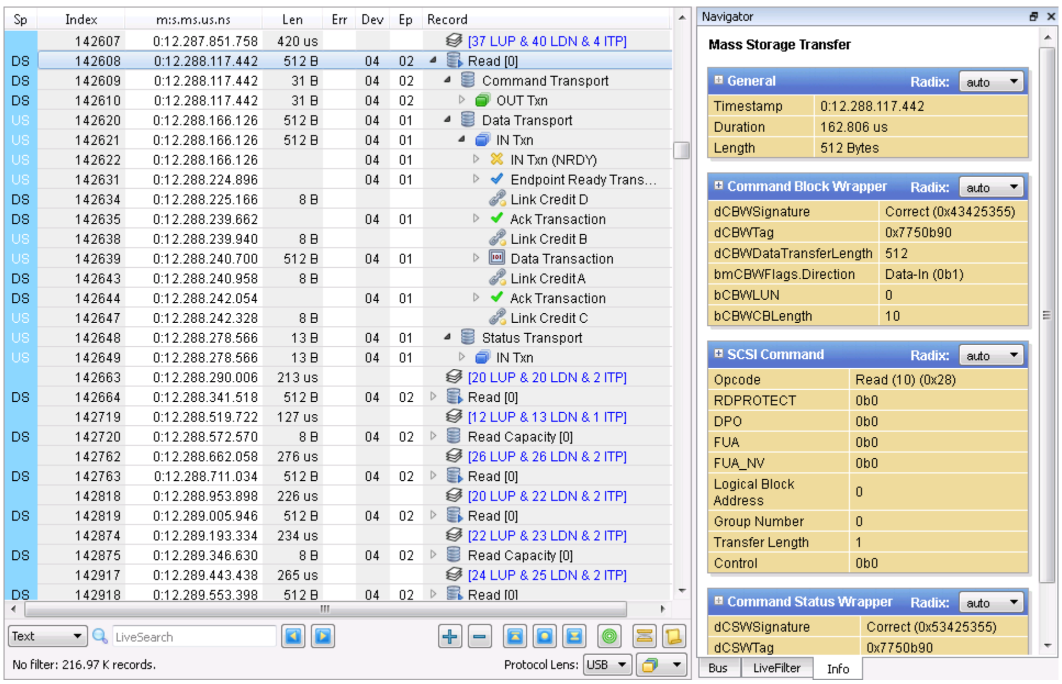


***Figure 83:*** *A parsed mass storage class transaction.*

### Control Transfers

An additional benefit of enabling classification is that the protocol-level transactions that make up a control transfer are grouped into a single class-level transaction (Figure 84). If the control transfer can be parsed, the transaction will be named accordingly. Otherwise, the transaction will show up as "Control Transfer."

### Bulk Transfer Grouping

If a device with an unrecognized class is enumerated, the Data Center can group bulk endpoint transfers based on a short packet delimitation. To enable this feature, right click on the *Universal Serial Bus* item in the USB Bus Pane and choose *Enable Bulk Ep Grouping*.

The device has to be enumerated for this feature to work. If the device has not been enumerated, the Managed Configurations (Section 6.10) feature can be used to set a custom enumeration. The bulk endpoints have to be properly listed under an interface with the proper maximum endpoint size set for proper operation of bulk transfer grouping.

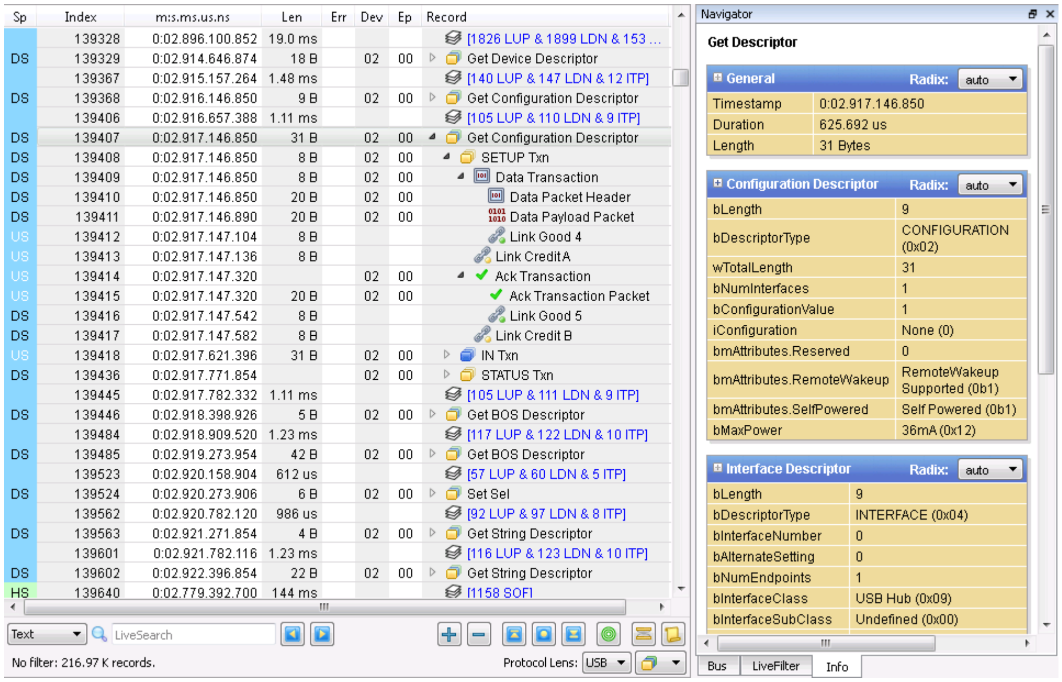


*Figure 84: A parsed control transfer.*

## 6.10 Bus Pane

The USB Bus pane (Figure 85) provides detailed information about each device on the bus. Clicking on a packet in the Transaction window will highlight the related device in the Bus pane.

When performing a simultaneous USB 3.0/2.0 capture, separate bus trees are available for USB 2.0 and USB 3.0 because both buses exists separately, but in parallel.

The Bus Pane is divided into two main sections. The top section displays the bus tree of all the USB devices that have been detected on the bus. The bottom section of the screen provides enumeration information and statistics about the bus or device selected in the bus tree.

### Real-Time Statistics Pane

The Statistics pane (Figure 39) provides a real-time count of Packets, Control Transfers, Errors, etc. as data is being captured. When an endpoint is selected in the Bus Pane, only its data is displayed in the Statistics Pane. When a device is selected, only the device data is displayed. When a bus is selected in the Bus Pane, the aggregate of the bus level data and the bus' connected devices data will be displayed in the statistics table.

When the individual statistics are expanded in the Statistics table, the sub data will vary. As it is possible for both USB 2.0 and USB 3.0 traffic to be on the same bus, data for a given bus will only display in the Statistics Pane when the bus in question is selected.

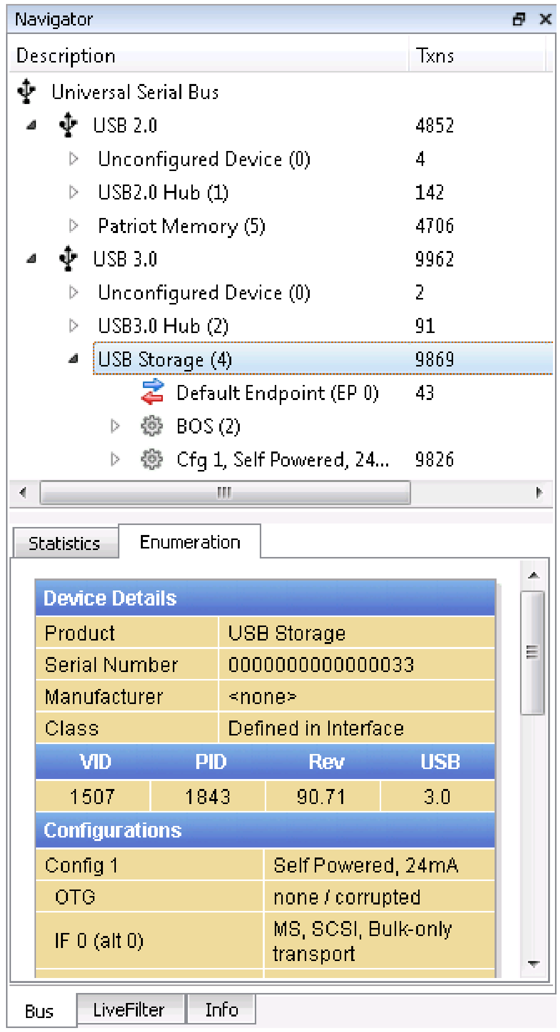



***Figure 85:*** *USB Bus Pane*

#### Enumeration

The Bus pane will only display descriptor information for devices whose enumeration was captured. It is possible to have missing or incomplete descriptor information if a capture is stopped prematurely or is interrupted. In these cases, it is possible to manually apply a device configuration to the device.

Descriptor information is stored in a cache for an entire capture session and is saved in the capture file as an “Enumerated Config”. The descriptor information associated to a device is based on the device’s address. Therefore all packets that are sent or received to the same address are considered to be interacting with the same device.

However, if the USB control message “SET ADDRESS” is seen, the software will parse all new descriptor information corresponding to a different target device, even if that device has the same USB address as a previously connected target device.

Please note that this feature can cause some strange behaviors. When appending to an older capture file, different devices may share the same address. The Data Center software may

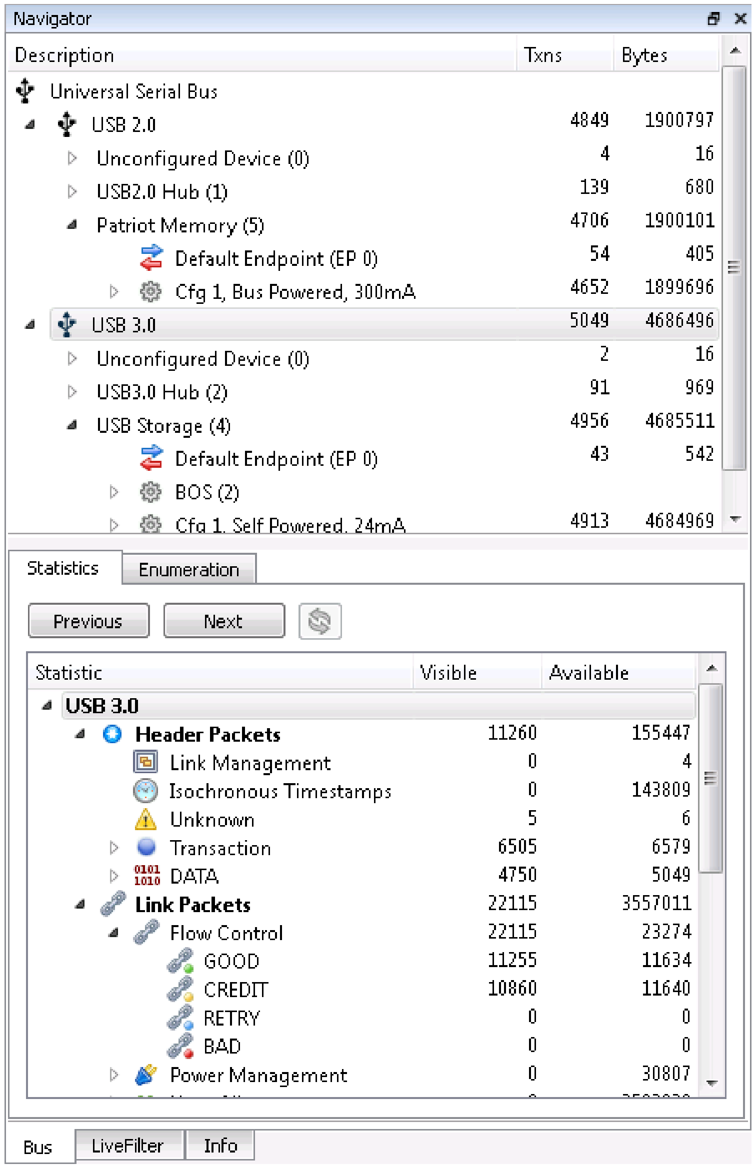



*Figure 86: Real-time Statistics Pane provides a quick and easy way to access types of packets*

become confused and display the wrong descriptor information if a `SET ADDRESS` is not seen.

Based on how the operating system assigns device addresses, there may be duplicate addresses for different devices when devices are disconnected and plugged in.

Clicking on a device will show a summary of the descriptor information below the tree in the **Enumeration** tab. Expanding a device will reveal a hierarchy of descriptor information from the device, configuration, interface, and endpoint descriptors. Clicking on any level of the tree will show a parsed view of those descriptors and any child descriptors.

The packets and bytes columns list the number of each that have been sent or received from each endpoint, interface, configuration, and entire device. The byte count includes only the size of the data payload, excluding PIDs, CRCs, etc.

Right clicking in the Bus tree will reveal a popup menu that gives the user the option to apply a filter so that specific devices can be shown or hidden in the Transaction window. Also in the popup menus is a **Manage Configs** options. It is possible to manage the configuration information for the device and/or apply a new configuration.

### Configuration Management

By default, Data Center uses configuration descriptor information captured during the enumeration phase to configure class-level decoding of USB traffic. However, with the **Configuration Management** interface, users can apply arbitrary configuration descriptors to the captured USB device data. This provides a custom class-level decoding experience within the set of USB classes supported by Data Center.

To clarify, this feature does not add support for decoding custom USB *classes*, only for specifying custom *configuration descriptors*.

For step-by-step instructions on how to perform common tasks with this interface, refer *Common Tasks* at the end of this section.

The **Configuration Management** interface provides ways for the user to:

- **Create, Edit, and Delete** custom configuration descriptors that persist either on the user's machine or in the active capture file.
- **Assign** a custom configuration descriptor to an arbitrary `bConfigurationValue` of any device on the bus, which can change the way data is parsed during a capture.
- **Remove** one or all previously assigned custom configuration descriptors from a device on the bus, exposing the originally enumerated configuration descriptor where available.

These functions are provided through the **Configuration Management Window** (Figure 87) and the **Bus Pane**'s right-click context menu (Figure 88) through which the window is accessed.

The **Configuration Management Window** is broken up into three functional areas.

**Configurations Pane**

The **Configurations Pane** is a hierarchical list providing all available configurations separated into four categories:

- **Enumerated Configs** are the configurations enumerated during the capture. Being part of the capture, they can be copied but not modified.
- **Assignable Configs** are custom configurations that have been saved with the capture file. They are the only non-enumerated configurations that the user can directly assign to a device for customized class level decoding.
- **User Configs** are custom configurations that are saved in the user's preferences. These configurations are available to any capture opened with the user's Data Center software.

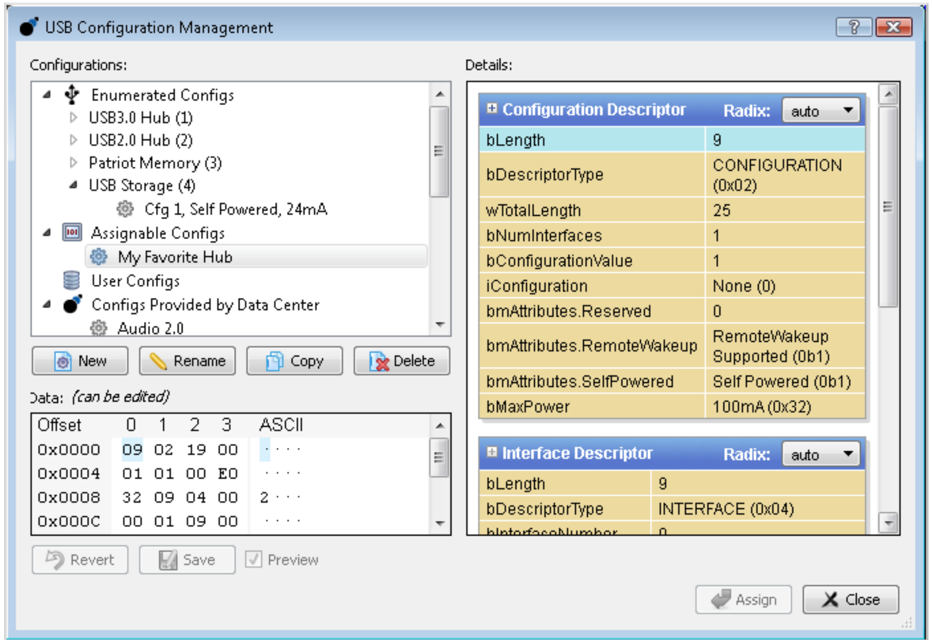



***Figure 87:*** *USB Configuration Management Window*

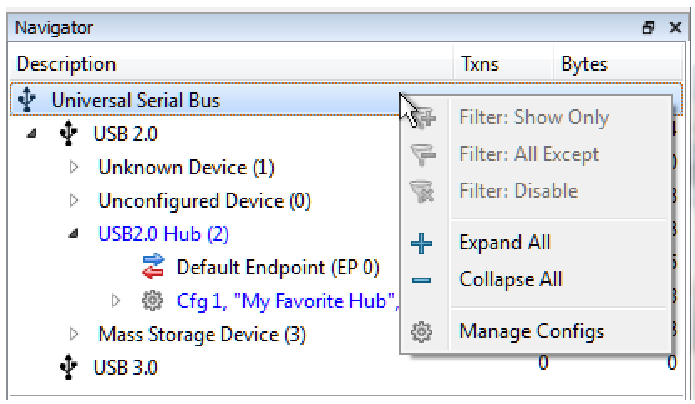



***Figure 88:*** *Access the configuration management interface through the* ***Bus Pane*** *context menu.*

- **Configs Provided by Data Center** are immutable custom configurations that were packaged with the application. Use them as examples or templates for making new configurations.

The user can create [New] custom configurations using the controls below the **Configurations Pane**

The user can also [Rename], [Copy], or [Delete] the selected configuration using the same controls.

**Data Pane**

Clicking a configuration in the **Configurations Pane** displays the descriptor's raw data in the bottom-left of the window. This data is editable in a variety of formats (e.g. hexadecimal, ASCII) for **Assigned** and **User** configurations. Configure the data display by right-clicking anywhere in the pane.

The **Data Pane** will highlight relevant regions of the data when a specific parameter is clicked in the **Details Pane**. Likewise, when changes are made in the data, the **Details Pane** shows the results immediately if Preview at the bottom is checked.

The user can also Save or Revert edits to a configuration using the controls beneath the **Data Pane**.

**Details Pane**

The configuration data is parsed into higher level parameters in the **Details Pane**. This display is similar to the **Enumeration** tab of the Bus Pane. When a specific parameter is highlighted in the **Details Pane**, the corresponding data is highlighted in the **Data Pane**. As the configuration data is modified in the **Data Pane**, the parameters in the **Details Pane** will update if Preview is checked.

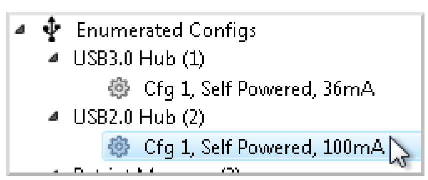
**Common Tasks**

***How to Save an Enumerated Configuration for Later***

Rather than re-enumerate a frequently used device on every new capture, it can be time-efficient to save the device's configuration descriptor and assign it in future captures using **Configuration Management**.

To save an enumerated configuration:

1. Right-click on any Bus, Device, or Configuration in the **Bus Pane** (Figure 88).
2. Click **Manage Configs** to open the **Configuration Management Window** (Figure 87).
3. Find and click on the enumerated configuration descriptor of interest in the **Enumerated Configs** section of the **Configurations Pane**:



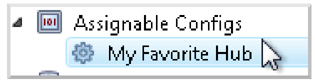

4. Click Copy and follow the instructions to name the new custom configuration.

***How to Assign a Custom Configuration to a Device***

The following steps will assign an existing custom configuration to a target device and `bConfigurationValue`:

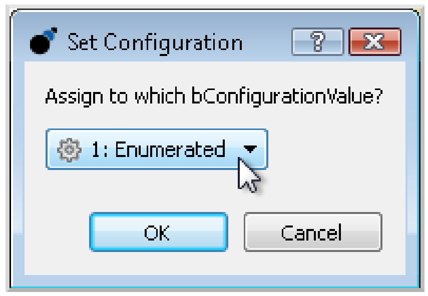
1. Right-click on the device of interest in the **Bus Pane** (Figure 88).
2. Click **Manage Configs**. This will open the **Configuration Management Window** (Figure 87).

3. Locate and click the configuration you would like to assign to the device in the **Assignable Configs**, **User Configs**, or **Configs Provided by Data Center** categories of the **Configurations Pane**:




4. Click   at the bottom right of the window. This will open a dialog allowing selection of a `bConfigurationValue` for assignment.

5. Select the desired `bConfigurationValue` and click **OK**:

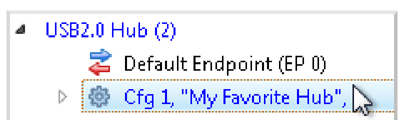



The custom configuration is now assigned and will appear blue in the **Bus Pane**. Until this configuration is removed, Data Center will class-decode all transfers against the device as if the custom configuration actually occupied the device's `bConfigurationValue`.

#### *How to Remove One Assigned Configuration from a Device*

The following steps will remove a specific previously assigned custom configuration from active class-level decoding:

1. Right-click on the configuration to remove in the **Bus Pane** (Figure 88).




2. Click **Remove Assigned Config** on the context menu opened by the previous step.

The custom configuration is removed, exposing any originally enumerated configuration on that `bConfigurationValue`.

#### *How to Remove All Assigned Configurations from a Device*

The following steps will remove all of a device's previously assigned custom configurations from active class-level decoding:

1. Right click on the target device in the **Bus Pane** (Figure 88).

2. Click **Remove Assigned Configs** on the context menu opened by the previous step.

All custom configurations are removed, exposing the device's originally enumerated configurations where present.

#### Details Window

Refer to section 5.2 for an overview of the Details View. Note that for USB captures, a bit-level timing view of the data is not available.

## 6.11 Filtering a USB Capture

A USB capture can be filtered in real time or after it has been completed. The Data Center software offers an extensive list of filters (Figure 89) to help developers filter out extraneous data. All filters are non-destructive and users are free to apply filters multiple times without losing data.

For a description of the General filters and how filtering works, refer to Section 5.5.

#### Bus Filters

##### Reset/Suspend/Connect Events

Unchecking this option will hide all bus events.

##### Collapsed

Unchecking this option will hide all collapsed transactions.

Note:

1. This setting will not affect collapsed SOFs and keep-alives, as they are bus events and not transactions.
2. When the Filter Protocol is in Packet mode, this setting will not affect any collapsed types that are still checked.

##### Digital Inputs

Unchecking this option will hide all digital input events.

#### USB 2.0

##### Devices

An integer value, or list of values, that filters transactions based on their device address. This filter only applies to transactions that have a device address. Addresses should be expressed as decimal values as they are shown in the **Dev** column. Multiple addresses can be listed separated by commas or spaces.

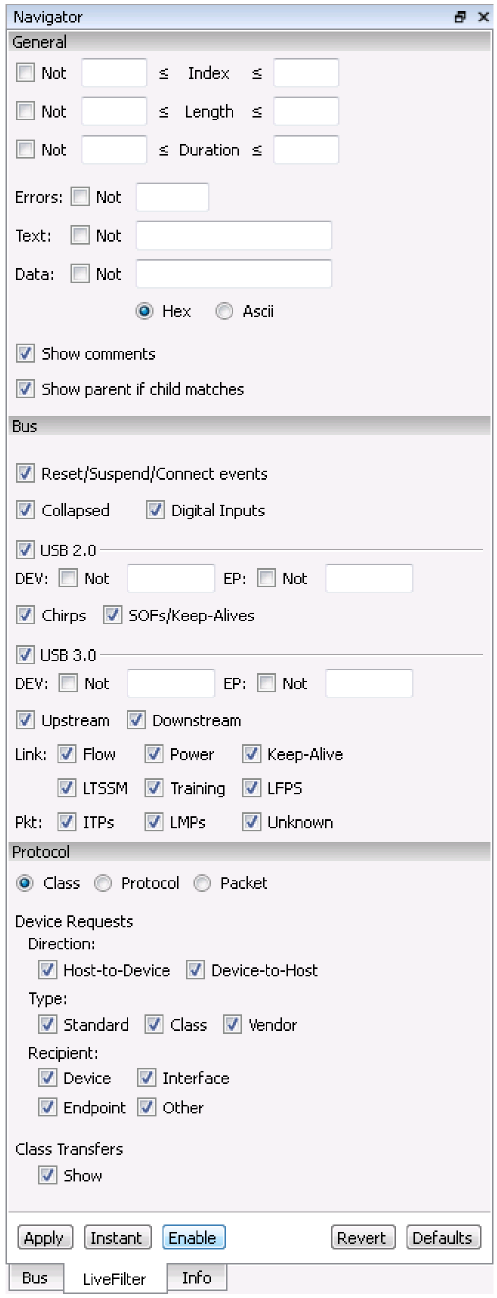



***Figure 89:*** *USB Filters*

### Endpoints

An integer value, or list of values, that filters transactions based on their endpoint address. This filter only applies to transactions that have an endpoint address. Addresses should be expressed as decimal values as they are shown in the **Ep** column. Multiple addresses can be listed separated by commas or spaces.

#### Chirps

Unchecking this option will hide all chirp J and chirp K events.

#### SOFs/Keep-Alives

Unchecking this option will hide all SOF and keep-alive events.

### USB 3.0

#### Devices

An integer value, or list of values, that filters transactions based on their device address. This filter only applies to transactions that have a device address. Addresses should be expressed as decimal values as they are shown in the **Dev** column. Multiple addresses can be listed separated by commas or spaces.

#### Endpoints

An integer value, or list of values, that filters transactions based on their endpoint address. This filter only applies to transactions that have an endpoint address. Addresses should be expressed as decimal values as they are shown in the **Ep** column. Multiple addresses can be listed separated by commas or spaces.

#### Upstream

Unchecking this option will hide all upstream traffic (from the Target Device to the Target Host).

#### Downstream

Unchecking this option will hide all downstream traffic (from the Target Host to the Target Device).

#### Link

The following settings are specific to Link Commands

#### Flow

Unchecking this option will hide all Link Flow Commands such as LGOOD and LCRD.

#### Power

Unchecking this option will hide all Link Power Commands such as LPM

#### Keep-Alive

Unchecking this option will hide all Link Keep-Alive Packets such as LUP and LDN.

#### LTSSM

Unchecking this option will hide all LTSSM transitions such as Polling.Idle -> U0.

#### Training

Unchecking this option will hide all Training sequences such as TSEQ, TS1, and TS2.

#### LFPS

Unchecking this option will hide all Low Frequency Packets.

#### Pkt

The following settings apply to Packets.

#### ITPs

Unchecking this option will hide all Isochronous Timestamp Packets.

#### LMPs

Unchecking this option will hide all Link Management Packets.

#### Unknown

Unchecking this option will hide all Unknown Packets.

## Protocol Filters

The Protocol filters are separated into three types: class filtering, transaction filtering, and packet filtering.

### Class

When Class is selected, the Device Requests and the Class Transfers options apply only to class-level transactions, not to the protocol-level transactions or the packets inside the class-level transactions.

The Device Requests options filter class-level transactions on the Default Control Pipe based on the `bmRequestType` field of the transfer. For example, unchecking the Host-To-Device option would hide all the Set device requests such as the Set Configuration request.

The Class Transfer option decides whether to show all of the class transactions which are not on the Default Control Pipe. For example, unchecking this option will hide all of the class transactions that occur on endpoints other than 0.

When all of the options under the Device Requests and the Class Transfers are selected, all transactions packets are matched. None of the protocol-level transactions, or individual packets will match unless all of the options under the Device Requests and the Class Transfers are selected.

#### Transactions

When Transactions is selected, the Token and Handshake options apply only to the protocol-level transactions as a whole, and not to the class-level transactions, or the individual packets inside the transactions.

Any protocol-level transaction that has BOTH a selected Token and a selected Handshake will match. None of the individual packets will match, unless all the Token and Handshake options are selected.

For example, selecting only IN, SETUP, ACK and NAK will show all IN/ACK, IN/NAK, SETUP/ACK and SETUP/NAK transactions. All other transaction groups will not be shown.

The Token and Handshake options are also applied to collapsed transactions. For example, unchecking the NAK handshake will hide all IN/NAK collapsed transactions.

#### Packets

When Packets is selected, only the selected packets match, and all of the class-level transactions and the protocol-level transactions do not match. This means that even with all the individual packet options checked, every transaction will appear in the Transaction window as a soft match. This is helpful for isolating specific packet types apart from transactions when used in conjunction with the **Show parent if child matches** option (Section 5.5).

# 7 I²C Monitoring

The Beagle I²C/SPI Protocol Analyzer is capable of non-intrusively monitoring I²C at up to 4 MHz.

Please note that captured I²C data is 9 bits wide because the ninth bit is the ACK/NACK bit to indicate whether the data was received properly. For this reason, I²C data will appear differently in the General views.

## 7.1 Performing an I²C Capture

Here are the steps for starting an I²C capture.

1. Start the Data Center application.
2. Connect the Beagle I²C/SPI analyzer to the analysis computer. Make sure that the green indicator LED has illuminated.
3. Connect the Beagle I²C/SPI analyzer to the I²C bus. The 10-pin ribbon cable can be connected directly, or the 10-pin split cable can be used to provide individual flying leads.
4. Click the **Connect to Analyzer...** button in the toolbar and connect to a Beagle I²C/SPI analyzer.
5. Make sure **I²C** is selected in the Protocol Lens pull-down menu under the Transaction window.
6. Click **Device Settings** in the toolbar and set the I²C capture settings. Make sure **I²C** is selected in the Capture Protocol pull-down menu.
7. Connect the Beagle I²C/SPI analyzer to the target device.
8. Click the **Run Capture** button to start the data capture. Once the capture has started, the capture indicator will turn green and an informational transaction will appear in the Transaction window which notes when the capture was started.
9. To stop the capture, click on the **Stop Capture** button.

## 7.2 I²C Device Settings

The I²C device settings described below can be configured in the Device Settings dialog (Figure 90). To open this dialog, click on the **Device Settings...** button.

The Device Settings dialog can also be accessed through **Analyzer | Device Settings...**.

### Sampling Rate

There are three different sampling rates which can be used to monitor the I²C bus. As a rule of thumb, it is recommended that the sampling rate should be at least 4 times faster than the data rate of the monitored bus. For a 400 kHz I²C bus, a sampling rate of 10 MHz would suffice.

To select a sampling rate, simply select the desired rate from the pull-down menu.

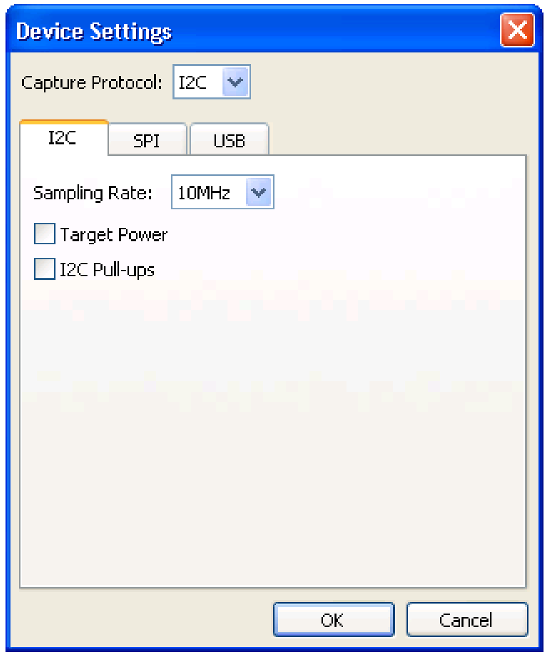



***Figure 90:*** *I²C Tab of the Device Settings Dialog*

**Target Power**

It is possible to power a downstream target, such as an I²C or SPI EEPROM with the Beagle analyzer's power (which is provided by the USB port). It is ideal if the downstream device does not consume more than 20-30 mA.

To enable or disable target power, check or uncheck the box in the Settings window.

**I²C Pull-ups**

There is a 2.2K resistor on each I²C line (SCL, SDA). The lines are effectively pulled up to 3.3V, so that results in approximately 1.5 mA of pull-up current. For more information about the pull-up resistors, please consult the Beagle I²C/SPI Protocol Analyzer datasheet.

To enable or disable the I²C pull-ups, check or uncheck the box in the Settings window.

## 7.3 Transaction Window

The I²C Transaction window (Figure 91) displays all the transactions that were captured on the I²C bus in real time. When a transaction is selected in the Transaction window, detailed information about that transaction is displayed in the Info pane.

For a general description of the Transaction window, see Section 5.1. The general description encompasses the behavior of the I²C Transaction window, with the following modifications:

**Start/Stop (S/P)**
This column is unique to the I²C Transaction window. It indicates whether the start and stop conditions were observed for each record. `S` indicates the start condition; `P` indicates the stop

| Index | m:s.ms.us | Dur | Len | Err | S/P | Addr | Record | Data |
|---|---|---|---|---|---|---|---|---|
| 0 | 0:00.000.000 | | | | | | Capture started | [12/16/09 10:11:17] |
| 1 | 0:44.186.201 | 1.78 ms | 1 B | | S | 50 | Write Transaction | 00 |
| 2 | 0:44.187.986 | 23.7 ms | 256 B | | SP | 50 | Read Transaction | 00 01 02 03 04 05 06 |
| 3 | 0:52.383.979 | 2.06 ms | 1 B | | S | 50 | Write Transaction | 00 |
| 4 | 0:52.386.043 | 23.7 ms | 256 B | | SP | 50 | Read Transaction | 00 01 02 03 04 05 06 |
| 5 | 0:52.710.997 | 2.32 ms | 1 B | | S | 50 | Write Transaction | 00 |
| 6 | 0:52.713.325 | 23.7 ms | 256 B | | SP | 50 | Read Transaction | 00 01 02 03 04 05 06 |
| 7 | 0:52.967.024 | 2.17 ms | 1 B | | S | 50 | Write Transaction | 00 |
| 8 | 0:52.969.203 | 23.7 ms | 256 B | | SP | 50 | Read Transaction | 00 01 02 03 04 05 06 |
| 9 | 0:53.185.063 | 2.16 ms | 1 B | | S | 50 | Write Transaction | 00 |
| 10 | 0:53.187.223 | 23.7 ms | 256 B | | SP | 50 | Read Transaction | 00 01 02 03 04 05 06 |
| 11 | 0:53.439.043 | 2.11 ms | 1 B | | S | 50 | Write Transaction | 00 |
| 12 | 0:53.441.157 | 23.7 ms | 256 B | | SP | 50 | Read Transaction | 00 01 02 03 04 05 06 |
| 13 | 0:53.655.087 | 2.13 ms | 1 B | | S | 50 | Write Transaction | 00 |
| 14 | 0:53.657.218 | 23.7 ms | 256 B | | SP | 50 | Read Transaction | 00 01 02 03 04 05 06 |
| 15 | 0:54.007.111 | 2.26 ms | 1 B | | S | 50 | Write Transaction | 00 |
| 16 | 0:54.009.374 | 23.7 ms | 256 B | | SP | 50 | Read Transaction | 00 01 02 03 04 05 06 |

***Figure 91:*** *I$^2$C Transaction Window*

condition. Transactions that have no stop condition (in the case of repeated start conditions) will have only S displayed.

**Address (Addr)**
The I$^2$C address of the slave device that was the target of the transaction. This number is in hexadecimal. “None” is displayed for transactions that are zero bytes long, and thus have no address field. An asterisk (*) following the address indicates that the address byte was NACK’ed. In certain situations, an I$^2$C transaction may not specify the lowest 8 bits of a 10-bit slave address. In these situations, the **Address (Addr)** column will render the incomplete addresses as 0XX, 1XX, 2XX, or 3XX, depending on the value of the first two address bits.

**Data**
In the I$^2$C **Data** column, NACK’ed bytes are followed by an asterisk (*) to differentiate them from ACK’ed bytes.

## 7.4 Details Window

The Details window has some extra features to accommodate the I$^2$C protocol. Refer to section 5.2 for an overview of the Details View, including the Data and Timing panes.

### Data Pane

The I$^2$C Data pane (Figure 92) provides a hexadecimal and ASCII dump of the contents of the transaction. Please note that it does not include the byte(s) which are composed of the slave address and read/write bit. In the I$^2$C Data pane, NACK’ed bytes are rendered in red text to differentiate them from ACK’ed bytes.

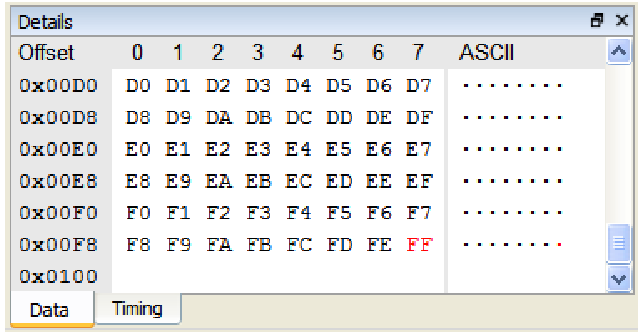



***Figure 92:*** *I²C Details Window - Data Pane*

### Timing Pane

In the I²C Timing pane (Figure 93), all the bytes from the transaction will be displayed in the pane, including start and stop conditions.

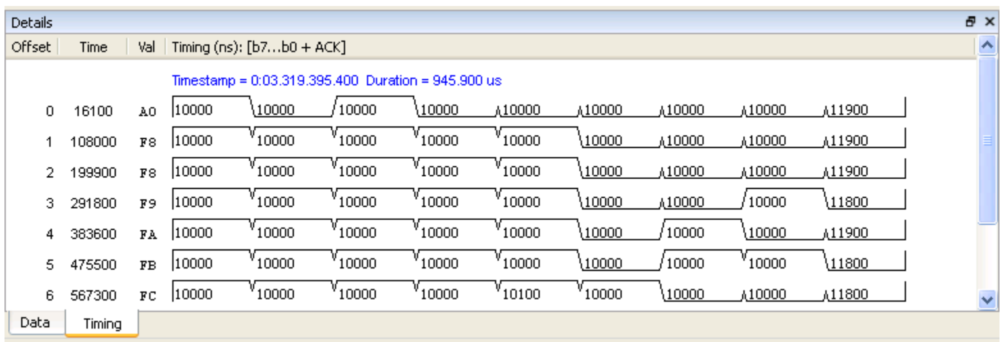



***Figure 93:*** *I²C Details Window - Timing Pane*

There are a few additional things to note:

- I²C data is sent MSB first and LSB last. In the column header for the Timing column, the bit order is indicated to be b7 . . . b0.
- The timing display for I²C actually shows 9 bits. The last bit is the ACK/NACK bit.

## 7.5 Filtering an I²C Capture

The following is a description of the parameters that are specific to the I²C protocol. For a description of the General parameters, or for information on how to operate the Filter pane, refer to Section 5.5. The I²C Filter pane (Figure 94) has protocol-specific filtering options under the **Bus** caption in the pane.

### 7-bit Addresses

Filter the transactions based on the I²C slave address of the message. The addresses should be specified in hexadecimal format. Multiple device addresses should be separated by commas

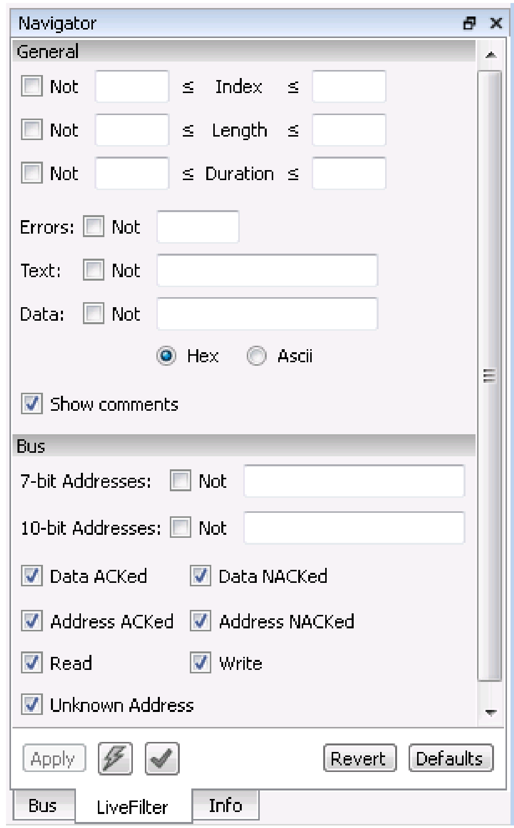



***Figure 94:*** *I²C Filter Pane*

or spaces. Note that this parameter only filters transactions that were addressed to slaves with 7-bit addresses.

**10-bit Addresses**

Filter the transactions based on the I²C slave address of the message. The addresses should be specified in hexadecimal format. Multiple device addresses should be separated by commas or spaces. Note that this parameter only filters transactions that were addressed to slaves with 10-bit addresses.

Partial 10-bit addresses can be specified as well using the 0XX, 1XX, 2XX, or 3XX notation as seen in the Transaction window.

**Data ACKed**

Unchecking this option will hide all transactions in which no data was NACK'ed

**Data NACKed**

Unchecking this option will hide all transactions in which any data was NACK'ed.

**Address ACKed**

Unchecking this option will hide all transactions in which the address was ACK'ed.

**Address NACKed**

Unchecking this option will hide all transactions in which the address was NACK'ed.

**Read**

Unchecking this option will hide all Read transactions.

**Write**

Unchecking this option will hide all Write transactions.

**Unknown Address**

Unchecking this option will hide all transactions that have an unknown address. An unknown address can occur when a transaction did not contain any data or encountered an error while transmitting the address.

# 8 SPI Monitoring

The Beagle I²C/SPI Protocol Analyzer is capable of non-intrusively monitoring SPI at up to 24 MHz. However, the Beagle analyzer may have difficulty monitoring continuous transactions at a sustained rate of 24 MHz. Please see the Beagle Analyzer datasheet for more details.

Please note that SPI is a full duplex protocol. For this reason, two bytes are recorded by the Data Center application during every 1-byte clock period. When the Data Center application displays these two bytes together, the first byte will be the MOSI byte and the second byte will be the MISO byte. There is no standard higher level protocol for SPI data.

## 8.1 Performing an SPI Capture

Here are the steps for starting an SPI capture.

1. Start the Total Phase Data Center application.
2. Connect the Beagle I²C/SPI analyzer to the analysis computer. Make sure that the green indicator LED has illuminated.
3. Connect the Beagle I²C/SPI analyzer to the SPI bus. The 10-pin ribbon cable can be connected directly, or the 10-pin split cable can be used to provide individual flying leads.
4. Click **Connect to Analyzer...** in the toolbar and connect to a Beagle I²C/SPI analyzer.
5. Select **SPI** from the Protocol Lens pull-down menu under the Transaction Window.
6. Click **Device Settings...** in the toolbar and set the SPI capture settings. Make sure **SPI** is selected in the Capture Protocol pull-down menu.
7. Connect the Beagle I²C/SPI analyzer to the target device.
8. Click the **Run Capture** button to start the data capture. Once the capture has started, the capture indicator will turn green and an informational transaction will appear in the Transaction window which notes when the capture was started.
9. To stop the capture, click on the **Stop** button.

## 8.2 SPI Device Settings

The SPI device settings described below can be configured in the Device Settings dialog (Figure 95). To open this dialog, click on the **Device Settings...** button.

### Sampling Rate

There are three different sampling rates which can be used to monitor the SPI bus. As a rule of thumb, it is recommended that the sampling rate should be at least 4 times faster than the data rate of the monitored bus. For a 1000 kHz SPI bus, a sampling rate of 10 MHz would suffice.

To select a sampling rate, simply select the desired rate from the pull-down menu.

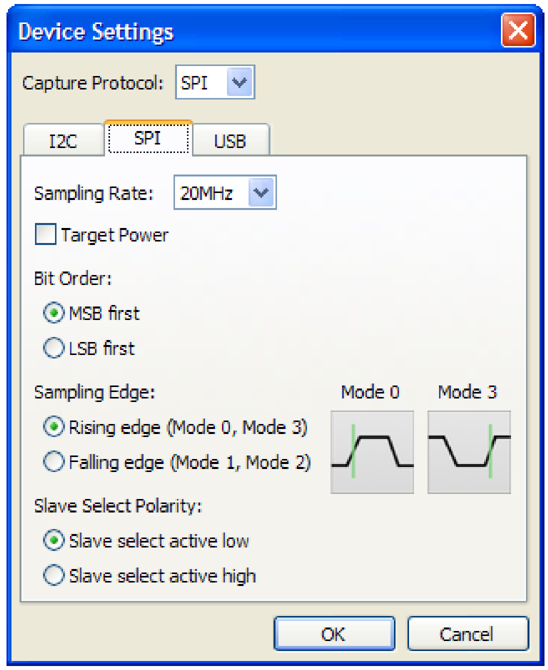



***Figure 95:*** *SPI Tab of the Device Settings Dialog*

### Target Power

It is possible to power a downstream target, such as an SPI flash or SPI EEPROM with the Beagle analyzer's power (which is provided by the USB port). It is ideal if the downstream device does not consume more than 20-30 mA.

To enable or disable target power, check or uncheck the box in the Settings window.

### Bit Order

Since SPI does not have a high level protocol, it is necessary for the user to specify the bit order of the data bytes in order to have the Data Center software properly parse the captured data.

**MSB first** means that the Most Significant Bit (MSB) is transmitted first. The byte order would be b7 . . . b0.

**LSB first** means that the Least Significant Bit (LSB) is transmitted first. The byte order would be b0 . . . b7.

### Sampling Edge

SPI has multiple modes (0, 1, 2, 3) which define the data frame for data transmission. In order for the Data Center software to correctly parse the captured data, the sampling edge of the data frame must be specified.

Mode 0 and 3 are sampled on the **Rising edge** of the clock and Mode 1 and 2 are sampled on the **Falling edge** of the clock.

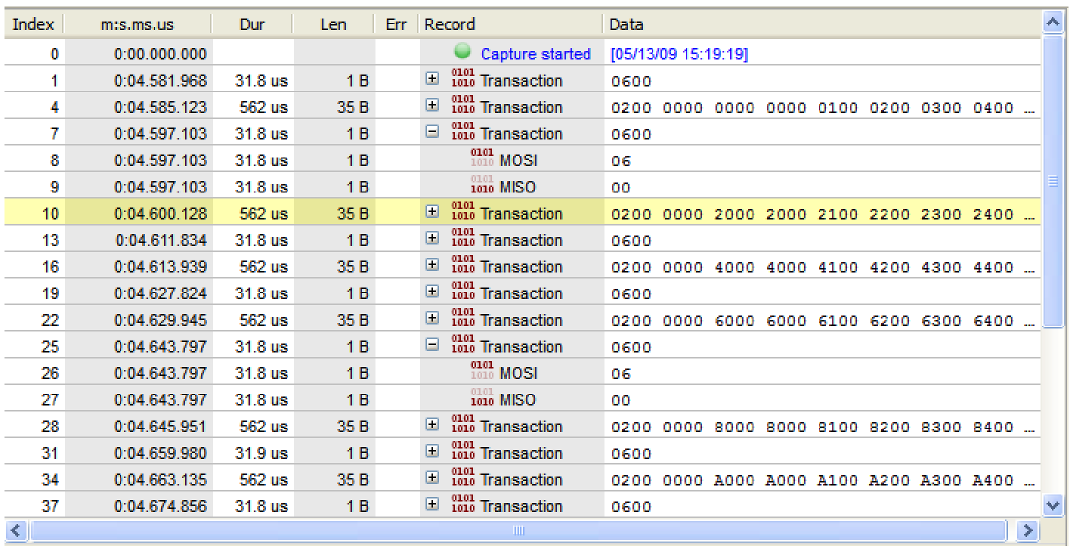
For more information about SPI modes, please refer to the SPI Background section of the Beagle Protocol Analyzer datasheet.

### Slave Select Polarity

Different SPI devices use different polarities on Slave Select to activate an SPI slave device. Slave select can be pulled low to activate the SPI slave or it can be pulled high to activate the SPI slave.

## 8.3 Transaction Window

The SPI Transaction window (Figure 96) displays all the transactions that were captured on the SPI bus in real time. When an transaction is selected in the Transaction window, detailed information about that transaction is displayed in the Info pane.



| Index | m:s.ms.us | Dur | Len | Err | Record | Data |
|---|---|---|---|---|---|---|
| 0 | 0:00.000.000 | | | | Capture started | [05/13/09 15:19:19] |
| 1 | 0:04.581.968 | 31.8 us | 1 B | | Transaction | 0600 |
| 4 | 0:04.585.123 | 562 us | 35 B | | Transaction | 0200 0000 0000 0000 0100 0200 0300 0400 ... |
| 7 | 0:04.597.103 | 31.8 us | 1 B | | Transaction | 0600 |
| 8 | 0:04.597.103 | 31.8 us | 1 B | | MOSI | 06 |
| 9 | 0:04.597.103 | 31.8 us | 1 B | | MISO | 00 |
| 10 | 0:04.600.128 | 562 us | 35 B | | Transaction | 0200 0000 2000 2000 2100 2200 2300 2400 ... |
| 13 | 0:04.611.834 | 31.8 us | 1 B | | Transaction | 0600 |
| 16 | 0:04.613.939 | 562 us | 35 B | | Transaction | 0200 0000 4000 4000 4100 4200 4300 4400 ... |
| 19 | 0:04.627.824 | 31.8 us | 1 B | | Transaction | 0600 |
| 22 | 0:04.629.945 | 562 us | 35 B | | Transaction | 0200 0000 6000 6000 6100 6200 6300 6400 ... |
| 25 | 0:04.643.797 | 31.8 us | 1 B | | Transaction | 0600 |
| 26 | 0:04.643.797 | 31.8 us | 1 B | | MOSI | 06 |
| 27 | 0:04.643.797 | 31.8 us | 1 B | | MISO | 00 |
| 28 | 0:04.645.951 | 562 us | 35 B | | Transaction | 0200 0000 8000 8000 8100 8200 8300 8400 ... |
| 31 | 0:04.659.980 | 31.9 us | 1 B | | Transaction | 0600 |
| 34 | 0:04.663.135 | 562 us | 35 B | | Transaction | 0200 0000 A000 A000 A100 A200 A300 A400 ... |
| 37 | 0:04.674.856 | 31.8 us | 1 B | | Transaction | 0600 |

***Figure 96:*** *SPI Transaction Window*

For a general description of the Transaction window, see Section 5.1. The general description encompasses the behavior of the SPI Transaction window, with the following caveats for each column:

### Data

For the top level SPI Transactions, data is displayed as a sequence of 2-byte words. The first byte of the word is the MOSI data and the second byte is the MISO data. The data is paired because SPI is a bidirectional protocol and the MOSI and MISO bytes appear on the bus at the same time. SPI transactions can be expanded into separate MISO and MOSI records, both of which contain the normal sequence of 1-byte words.

## 8.4 Details Window

The Details window has some extra features to accommodate the SPI protocol. Refer to section 5.2 for an overview of the Details window, including the Data and Timing panes.

### MOSI and MISO Data Panes

The SPI Details window separates the transaction data into the MOSI Data Pane (Figure 97) and the MISO Data Pane (Figure 98).

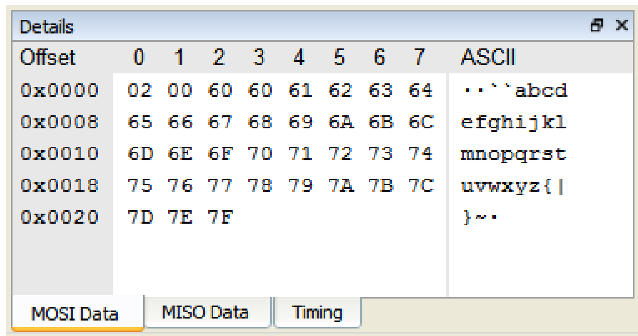



***Figure 97:*** *MOSI Data Pane of the SPI Details Window*

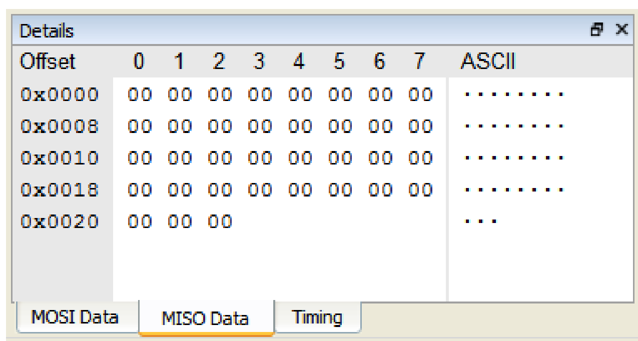



***Figure 98:*** *MISO Data Pane of the SPI Details Window*

Each of these Data panes behaves as the Data pane described in Section 5.2.

### Timing Pane

The SPI Timing Pane (Figure 99) overlays the bit timing diagram of the MOSI line with the MISO line. The MOSI line is displayed in red and the MISO line in blue. If a **Transaction** record is selected, the red and blue MOSI and MISO timing lines will overlap in the diagram. If either a **MOSI** or a **MISO** record is selected, only the MISO or MOSI line will be drawn in the Timing pane.

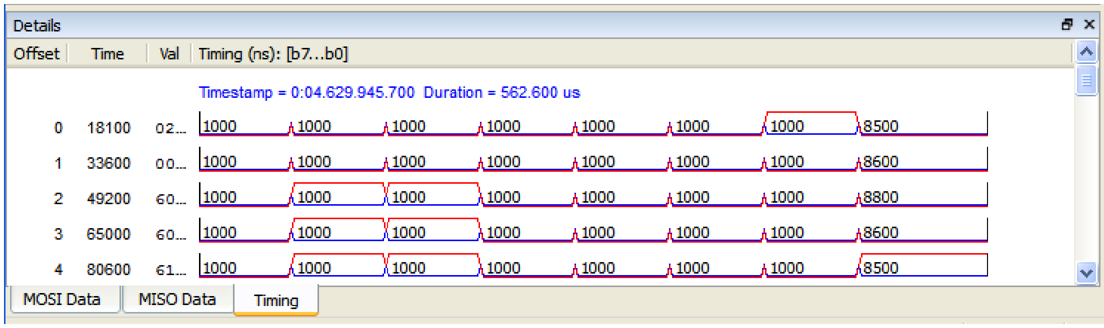



*Figure 99: Timing Pane of the SPI Details Window*

## 8.5 Filtering an SPI Capture

The following is a description of the parameters that are specific to the SPI protocol. For a description of the General parameters, or for information on how to operate the Filter Pane, refer to Section 5.5. The SPI Filter Pane (Figure 100) has protocol-specific filtering options under the **Bus** caption in the pane.

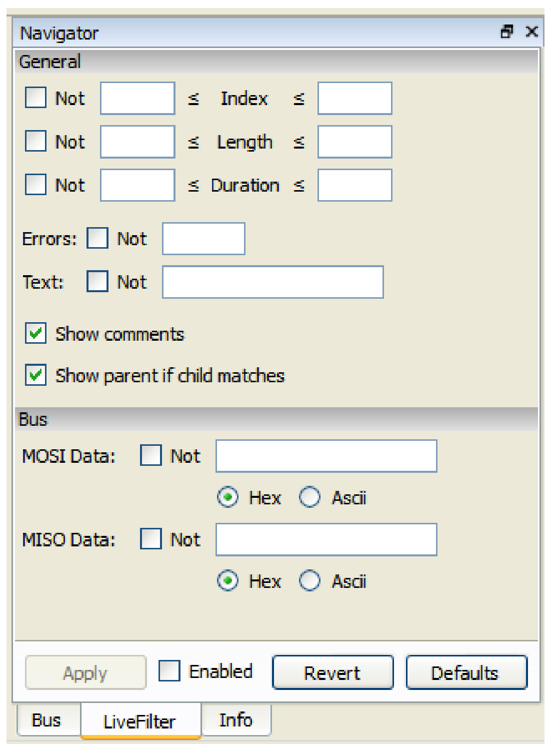



*Figure 100: Filter Pane located in the SPI Navigator Window*

### MOSI Data and MISO Data

In the SPI Filter pane, there is no **Data** field in the General parameters section. It is replaced by two Data fields, one that matches only MOSI Data and one that matches only MISO Data.

These Data parameters accept the same syntax described in Section 5.5.

# 9 CAN Monitoring

The Komodo CAN Duo Interface is a CAN interface capable of active CAN data transmission as well as non-intrusive CAN bus monitoring.

The Komodo CAN Duo Interface has two independent, customizable CAN channels along with eight configurable GPIOs. The Komodo interface also has two virtual USB ports (via a single physical USB port).

The two CAN channels make simultaneous communication on and/or monitoring of two separate CAN buses possible using a single Komodo CAN Duo Interface. GPIO, General Purpose IO, allows users to synchronize external logic with a CAN channel, as well as output events to external devices, such as oscilloscopes. The two virtual USB ports allow users to communicate with a single Komodo interface simultaneously from two software applications.

## 9.1 Performing a CAN Capture

Here are the steps for starting a CAN capture.

1. Start the Total Phase Data Center application.
2. Connect the Komodo CAN Duo Interface to the analysis computer. Make sure that the green indicator LED has illuminated.
3. Connect the Komodo CAN DUO Interface to the CAN bus.The Komodo CAN Duo Interface features two connectors for each CAN channel: a common DB-9 connector and a block screw terminal which wires can easily connect to. Ensure the CAN bus is properly terminated, otherwise the Komodo is saturated with CAN errors.
4. Click **Connect to Analyzer...** in the toolbar and connect to a Komodo CAN Duo Interface.
5. Select **CAN** from the Protocol Lens pull-down menu under the Transaction Window.
6. Click **Device Settings...** in the toolbar and set the CAN capture settings. Make sure **CAN** is selected in the Capture Protocol pull-down menu.
7. Connect the Komodo CAN Duo Interface to the target device.
8. Click the **Run Capture** button to start the data capture. Once the capture has started, the capture indicator will turn green and an informational transaction will appear in the Transaction window which notes when the capture was started.
9. To stop the capture, click on the **Stop** button.

## 9.2 CAN Device Settings

The CAN device settings described below can be configured in the Device Settings dialog (Figure 101). To open this dialog, click on the **Device Settings...** button.

The Device Settings dialog can also be accessed through **Analyzer | Device Settings...**.

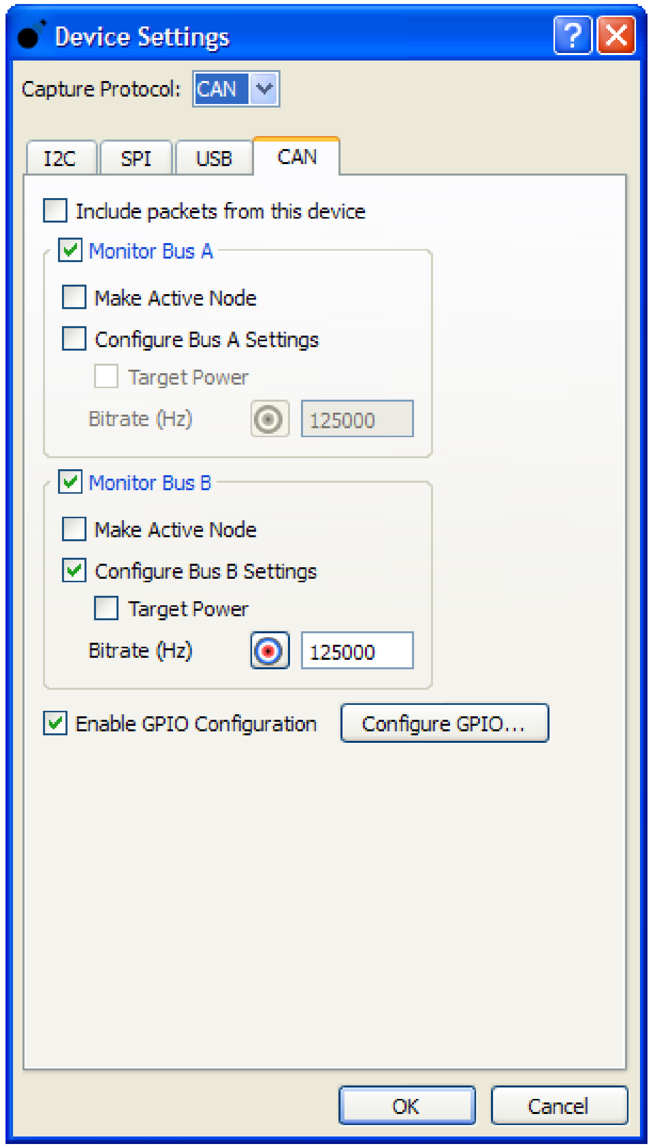



***Figure 101:*** *CAN Tab of the Device Settings Dialog*

#### Monitor channel

**Monitor channel** checkbox enables monitoring selected channel (A or B) on the connected Komodo CAN Duo Interface port. Data Center software can monitor either or both of the channels.

#### Make active node

**Make active node** checkbox enables packet ACKing on the corresponding channel on the connected Komodo port. When the box is checked, the application will acquire the appropriate Control feature from the Komodo port. When the box is unchecked, the port will act as a passive monitor (listen-only) on the CAN bus. For more information, refer to the Komodo CAN Duo Interface datasheet. This option is valid only when **Monitor channel** is checked for this channel.

**Include packets from this unit**

If checked, CAN traffic generated by this Komodo unit will be includes

**Configure Bus Settings**

**Configure Bus Settings** checkbox enables congiguring the corresponding channel on the connected Komodo port. When the box is checked, the application will acquire the appropriate Config feature from the Komodo port. When the box is unchecked, all configuration options will be disabled on this channel. This option is valid only when **Monitor channel** is checked for this channel. Only one application can configure the bus settings at a time.

**Target power**

It is possible to power one or more downstream CAN nodes using the V+ pin. The Komodo CAN Duo Interface can source a maximum of 73mA per CAN channel with V+.

To enable or disable target power on corresponding channel, check or uncheck the box in the Settings window. This option is valid only when **Monitor channel** and **Acquire config resource** are checked for this channel.

**Bitrate (Hz)**

The bitrate can be manually configured for each channel by entering a bitrate value in the **Bitrate Field** of the dialog. The resulting bitrate may slightly differ from the entered value, as only certain discrete bitrates are permitted. The application will set the bitrate to the value nearest the entered bitrate.

The bitrate can be set automatically for each channel by pressing the **Auto Bitrate Button** . Once the bitrate detection operation completes, the result will appear in the corresponding **Bitrate Field**.

These options are valid only when **Monitor channel** and **Acquire config resource** are checked for this channel.

**Enable GPIO Configuration**

**Enabled GPIO Configuration** checkbox enables configuring GPIO pins. When the box is checked, the application will acquire GPIO Control feature from the Komodo port. When the box is unchecked, configuring GPIO pins is disabled. Only one application can configure the GPIO interface at a time.

**Configure GPIO** button opens the **Komodo GPIO configuration** dialog.

**Komodo GPIO configuration**

The **Komodo GPIO configuration** dialog offers a way to configure general purpose input and output (GPIO) pins. Each of the pins labeled with IN 1, IN 2, IN 3, IN 4 can be configured as an input by dragging one of the inputs from **Configure Input** section and dropping it on one of

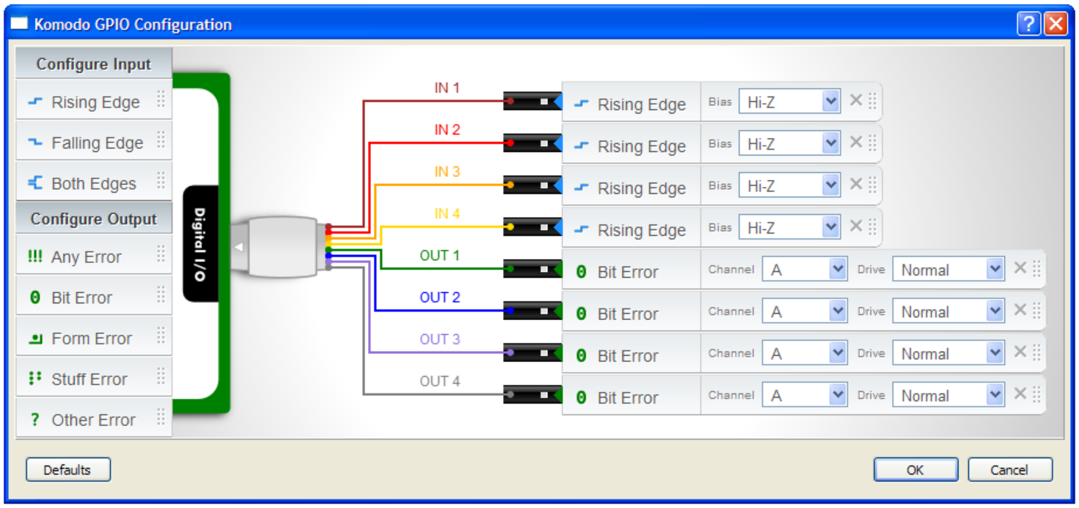



***Figure 102:*** *Komodo GPIO configuration Dialog*

these pins. Each of the pins labeled with OUT 1, OUT 2, OUT 3, OUT 4 can be configured as an output by dragging one of the outputs from **Configure Output** section and dropping it on one of these pins. The various options for each pin type are described below.

### Configure Input

Digital inputs allow users to synchronize external logic with a CAN channel. Whenever the state of an enabled digital input changes, an event will be sent to the analysis PC and displayed in the transaction log.

- **Rising Edge** - Report change on rising edge.
- **Falling Edge** - Report change on falling edge.
- **Both Edges** - Report change on both edges.

The digital input options are as follows:

#### Bias

Specify a voltage bias for an input pin.

- **Pull-Down** - Pulls down input voltage using high impedance resistor to GND.
- **Pull-Up** - Pulls up input voltage using high impedance resistor to 3.3V.
- **Hi-Z** - No modification to input voltage.

### Configure Output

Digital outputs allow users to output events to external devices. A common use for this feature is to trigger an oscilloscope or logic analyzer to capture data. The output pins can be activated on the various conditions below. Refer to the Komodo datasheet for details on the output signal characteristics and refer to the CAN specification for details on the different error types.

- **Any Error** - Output pulse on any error.

- **Bit Error** - Output pulse on bit error.
- **Form Error** - Output pulse on form error.
- **Stuff Error** - Output pulse on stuff error.
- **Other Error** - Output pulse on other error.

The digital output options are as follows:

**Channel**

Specify a source channel for an error activated output pin.

- **A** - Active on CAN A error.
- **B** - Active on CAN B error.
- **Both** - Active on CAN A or B error.

**Drive**

Specify the voltage drive for an output pin.

- **Normal** - Active is 3.3V; Inactive is GND.
- **Inverted** - Active is GND; Inactive is 3.3V.
- **Open Drain** - Active is GND; Inactive is floating.
- **+Pullup** - Equivalent to **Open Drain** with a high impedance pullup.

**Defaults**

Set the default configuration for all of the pins. This will configure IN pins as inputs with pull-downs that report changes on both edges and OUT pins as outputs that activate on any error.

## 9.3 Transaction Window

The CAN Transaction window (Figure 103) displays all the transactions that were captured on the CAN bus in real time. When a transaction is selected in the Transaction window, detailed information about that transaction is displayed in the Info pane.

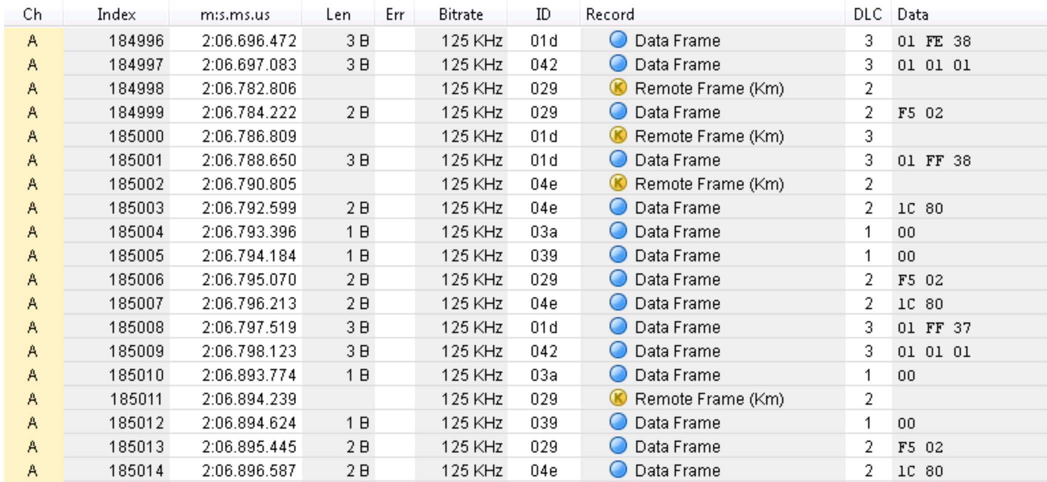
For a general description of the Transaction window, see Section 5.1. The general description encompasses the behavior of the CAN Transaction window, with the following modifications:

**Ch**
The channel on which the packet or event occurred.

**Error codes (Err)**
Error codes listing abnormal conditions that occurred while capturing the transaction. See Table 2 for the possible error codes. In addition, there are several CAN specific error codes as described in Table 8.

**Bitrate**
The bitrate of the CAN bus in kHz.

**ID**
The ID of the source CAN node of the CAN packet. When a packet is marked as RTR, the ID, instead, corresponds to the destination CAN node (the requestee).



| Ch | Index | m:s.ms.us | Len | Err | Bitrate | ID | Record | DLC | Data |
|---|---|---|---|---|---|---|---|---|---|
| A | 184996 | 2:06.696.472 | 3 B | | 125 KHz | 01d | Data Frame | 3 | 01 FE 38 |
| A | 184997 | 2:06.697.083 | 3 B | | 125 KHz | 042 | Data Frame | 3 | 01 01 01 |
| A | 184998 | 2:06.782.806 | | | 125 KHz | 029 | Remote Frame (Km) | 2 | |
| A | 184999 | 2:06.784.222 | 2 B | | 125 KHz | 029 | Data Frame | 2 | F5 02 |
| A | 185000 | 2:06.786.809 | | | 125 KHz | 01d | Remote Frame (Km) | 3 | |
| A | 185001 | 2:06.788.650 | 3 B | | 125 KHz | 01d | Data Frame | 3 | 01 FF 38 |
| A | 185002 | 2:06.790.805 | | | 125 KHz | 04e | Remote Frame (Km) | 2 | |
| A | 185003 | 2:06.792.599 | 2 B | | 125 KHz | 04e | Data Frame | 2 | 1C 80 |
| A | 185004 | 2:06.793.396 | 1 B | | 125 KHz | 03a | Data Frame | 1 | 00 |
| A | 185005 | 2:06.794.184 | 1 B | | 125 KHz | 039 | Data Frame | 1 | 00 |
| A | 185006 | 2:06.795.070 | 2 B | | 125 KHz | 029 | Data Frame | 2 | F5 02 |
| A | 185007 | 2:06.796.213 | 2 B | | 125 KHz | 04e | Data Frame | 2 | 1C 80 |
| A | 185008 | 2:06.797.519 | 3 B | | 125 KHz | 01d | Data Frame | 3 | 01 FF 37 |
| A | 185009 | 2:06.798.123 | 3 B | | 125 KHz | 042 | Data Frame | 3 | 01 01 01 |
| A | 185010 | 2:06.893.774 | 1 B | | 125 KHz | 03a | Data Frame | 1 | 00 |
| A | 185011 | 2:06.894.239 | | | 125 KHz | 029 | Remote Frame (Km) | 2 | |
| A | 185012 | 2:06.894.624 | 1 B | | 125 KHz | 039 | Data Frame | 1 | 00 |
| A | 185013 | 2:06.895.445 | 2 B | | 125 KHz | 029 | Data Frame | 2 | F5 02 |
| A | 185014 | 2:06.896.587 | 2 B | | 125 KHz | 04e | Data Frame | 2 | 1C 80 |

***Figure 103:*** *CAN Transaction Window*

***Table 8:*** *CAN-specific error code values*

| Code | Meaning | Description |
|---|---|---|
| B | Bit | The observed state (level) of a transmitted bit was different from the known transmitted value. |
| F | Form | A fixed-form bit field contained one or more illegal bits. |
| O | Other | An error other than bit, form or stuff was observed on the bus. |
| S | Stuff | A bit stuff error occurred – more than 5 consecutive bits with the same level were received. |

**DLC**

**DLC** (Data Length Code) is the specified number of bytes transmitted in a single CAN packet.

**Data**

The data payload for CAN packets, and a textual description for CAN events, errors, and capture events.

## 9.4 Filtering a CAN Capture

The following is a description of the parameters that are specific to the CAN protocol. For a description of the General parameters, or for information on how to operate the Filter pane, refer to Section 5.5.

**ID**

An ID that filters the transactions based on the ID of the source CAN node.

**Extended ID**

An extended ID that filters the transactions based on the extended ID of the source CAN node.

**Data Length Code**

An iteger that filters the transactions based on the DLC field.

**Traffic**

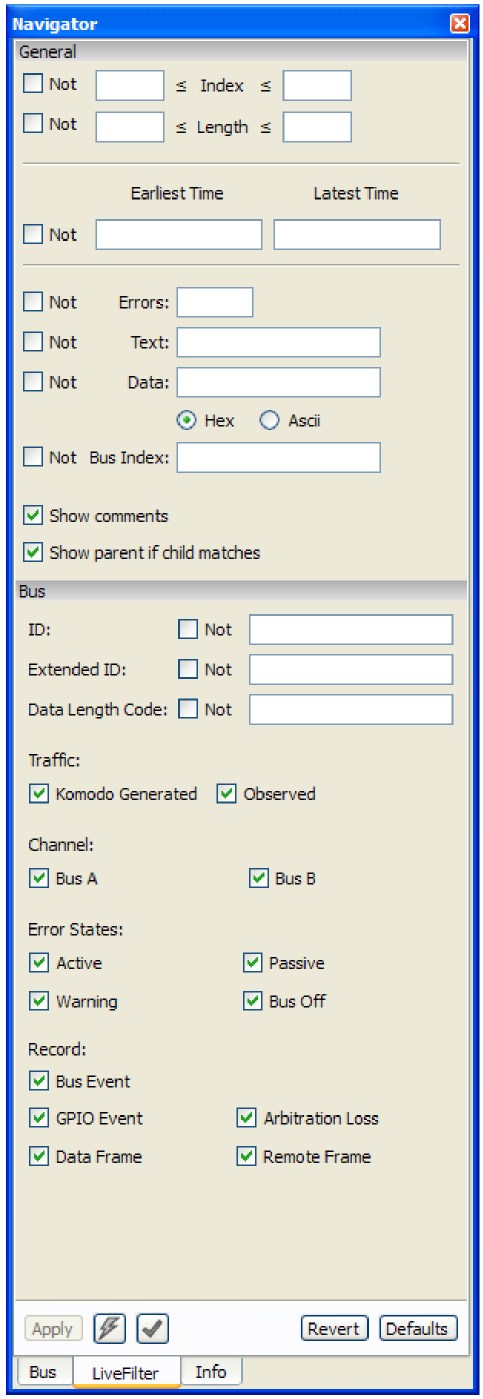



***Figure 104:*** *CAN Filter Pane*

Filters traffic that originated from the Komodo or traffic that was observed from other sources.

**Channel**
Filters transactions to include packets and events that occured only on the selected channels.

**Error States**
Filters instances of entry into one of the following error states: [ Active, Passive, Warning, or

Bus Off]

**Bus Event**
Filters traffic that is considered a bus event such as a change in bitrate.

**GPIO Event**
Filters traffic that is considered a GPIO event (Digital Input).

**Arbitration Loss**
Filters instances where the Komodo is aware that it has lost an arbitration event.

**Data Frame**
Filters Data Frames from view.

**Remote Frame**
Filters Remote Frames from view.

## 9.5 Bus Pane

**Bus**
Clicking on any device in the Bus Pane will display bus specific statistics information in the stats pane and allow filtering according to that specific bus.

**Device**
Clicking on any device in the Bus Pane will display device specific statistics information in the stats pane.

**Transaction Type**
Clicking on a transaction type (Data/Remote) in the Bus Pane will not display data in the statistics pane, but will allow filtering according to the transaction type for the device.

**Traffic Origin**
Clicking a given traffic origin (Komodo/Observed) in the Bus Pane will display statistics and allow to filter traffic corresponding to the bus-device-transaction-origin.

# 10 Troubleshooting

## 10.1 General

**When attempting to open the Connection Dialog, I receive the following error message: "Could not detect the attached Beagle analyzers for the following reason: Incompatible driver- Please check your CD or the Total Phase website for an updated driver."**

A driver newer than the version installed is required. Please refer to the Beagle analyzer datasheet for instructions on upgrading the Beagle analyzer USB driver.

**I've connected my device to a Beagle analyzer. When I try to capture data, no packets are shown.**

Please try the following:

- Make sure that you are viewing the correct protocol. To change the Protocol Lens, select the correct protocol from the Protocol Lens pull-down menu under the Transaction window.
- Disable the filter to make sure you are seeing all the packets.
- For $I^2C$ and SPI, make sure that you have selected the correct protocol from the Device Settings Dialog, as either protocol may be used with the Beagle $I^2C$/SPI Analyzer. You will have to stop your capture in order to change this setting.
- For $I^2C$, make sure that the $I^2C$ pullups are set correctly for your target device.
- For SPI, make sure that your capture settings are set to the correct sampling edge, bit order and slave select polarity.
- If the downstream target requires power from the Beagle $I^2C$/SPI analyzer, please make sure that target power has been turned on in the settings.
- For USB, if you are testing a high-speed device, make sure you connect the device to the Beagle USB 12 analyzer through a full-speed USB hub or you are using a Beagle USB 480 analyzer.

**I've set some filters, but the contents of the transaction window have not changed.**

Filters are not applied to the transaction window until the **Apply** button has been pressed. After settings all your filters, make sure you click on the **Apply** button.

## 10.2 USB

**I've plugged in a device into the Beagle USB 480 analyzer and it is acting strangely.**

Be sure that the analysis end of the Beagle USB 480 analyzer is plugged in prior to plugging in any devices on the target end. This ensures that the devices in the analyzer hardware that isolate the USB bus on the target end are functioning and the target device can communicate properly.

**The descriptor information does not appear for my device even though I am able to capture data from the device and it works fine on the host computer.**

In order for the Data Center software to correctly parse and display the descriptor data for a target device, the entire enumeration process must be captured. In order to ensure that this entire sequenced is captured, we recommend that the user start the capture before connecting the target device.

**I'm running a capture with the Beagle USB 480 analyzer and I'm seeing a lot of IN packets with no data or handshake response.**

Because the USB protocol is broadcast in the downstream direction, it is possible to see packets from parallel USB links. But only the downstream packets from the host to other USB devices will be observed; upstream packets from other devices to the host will not be seen.

It is possible that the IN packets observed may be directed to the Beagle USB 480 analyzer itself. Methods for dealing with these packets are described in Section 6.4.

**I've plugged in my target device into the target device port of the Beagle USB 12 analyzer. When I try to capture data, no packets are shown.**

The Beagle USB 12 protocol analyzer can only capture full-speed and low-speed USB and cannot capture high-speed USB directly. Please make sure that the target device is not a high-speed USB device.

If you would like to capture the USB data of a High-speed device with the Beagle USB 12 analyzer, connect the device to the Beagle Analyzer through a full-speed hyb in order to downgrade the speed of the data.

**I get a lot of sync errors when capturing USB data with the Beagle USB 12 analyzer.**

Sync errors can be caused by a poor USB connection or an analysis computer that has insufficient resources available for the Beagle Data Center application.

Here are some possible ways to eliminate sync errors:

- Use only USB ports that are mounted directly on the computer's motherboard. USB ports that are not mounted directly may perform poorly due to cable or connector quality.
- For best performance, it is recommended that a Beagle analyzer does not share its USB host controller. All other USB devices should be connected to separate controllers.
- Make sure that your computer has adequate physical memory. The Data Center software can become unstable if your computer starts to swap into virtual memory.
- Make sure that your computer is not running any other performance or resource hungry applications.
- You may want to consider using one computer as the analysis computer and a separate computer as the target host computer.
- It may be possible that the USB signals between the target host and the target device are at the very edge of compliance. If this is the case, the Beagle analyzer may encounter

errors when trying to capture the data. One way to test this is to use a USB hub in-line between the Beagle analyzer and the target device. The hub will retransmit the USB data. If this resolves the problem, the electrical signals of the target device should be examined in further detail.

- If the error is due to USB signals on the edge of compliance, you may be able to mitigate this issue by using shorter USB cables.

# 11 Legal / Contact

## 11.1 Disclaimer

All of the software and documentation provided in this datasheet, is copyright Total Phase, Inc. ("Total Phase"). License is granted to the user to freely use and distribute the software and documentation in complete and unaltered form, provided that the purpose is to use or evaluate Total Phase products. Distribution rights do not include public posting or mirroring on Internet websites. Only a link to the Total Phase download area can be provided on such public websites.

Total Phase shall in no event be liable to any party for direct, indirect, special, general, incidental, or consequential damages arising from the use of its site, the software or documentation downloaded from its site, or any derivative works thereof, even if Total Phase or distributors have been advised of the possibility of such damage. The software, its documentation, and any derivative works is provided on an "as-is" basis, and thus comes with absolutely no warranty, either express or implied. This disclaimer includes, but is not limited to, implied warranties of merchantability, fitness for any particular purpose, and non-infringement. Total Phase and distributors have no obligation to provide maintenance, support, or updates.

Information in this document is subject to change without notice and should not be construed as a commitment by Total Phase. While the information contained herein is believed to be accurate, Total Phase assumes no responsibility for any errors and/or omissions that may appear in this document.

Screengrabs in this document may not accurately reflect the latest software version.

## 11.2 Life Support Equipment Policy

Total Phase products are not authorized for use in life support devices or systems. Life support devices or systems include, but are not limited to, surgical implants, medical systems, and other safety-critical systems in which failure of a Total Phase product could cause personal injury or loss of life. Should a Total Phase product be used in such an unauthorized manner, Buyer agrees to indemnify and hold harmless Total Phase, its officers, employees, affiliates, and distributors from any and all claims arising from such use, even if such claim alleges that Total Phase was negligent in the design or manufacture of its product.

## 11.3 Additional Licenses

- Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## 11.4 Contact Information

Total Phase can be found on the Internet at http://www.totalphase.com/. If you have support-related questions, please email the product engineers at support@totalphase.com. For sales inquiries, please contact sales@totalphase.com.