# Arinc 664

## Windows BSP

AIM
RIGHT ON TARGET

Avionics Databus
Solutions

**Getting Started**

Version 19.2.1
May, 2019

# Arinc 664

## Windows BSP

**Getting Started**

Version 19.2.1
May, 2019

AIM NO. 60-15900-35-19.2.1

**AIM – Gesellschaft für angewandte Informatik und Mikroelektronik mbH**

**AIM GmbH**
Sasbacher Str. 2
D-79111 Freiburg / Germany
Phone +49 (0)761 4 52 29-0
Fax +49 (0)761 4 52 29-33
sales@aim-online.com

**AIM GmbH – Munich Sales Office**
Terofalstr. 23a
D-80689 München / Germany
Phone +49 (0)89 70 92 92-92
Fax +49 (0)89 70 92 92-94
salesgermany@aim-online.com

**AIM UK Office**
Cressex Enterprise Centre, Lincoln Rd.
High Wycombe, Bucks. HP12 3RB / UK
Phone +44 (0)1494-446844
Fax +44 (0)1494-449324
salesuk@aim-online.com

**AIM USA LLC**
Seven Neshaminy Interplex
Suite 211 Trevose, PA 19053
Phone 267-982-2600
Fax 215-645-1580
salesusa@aim-online.com

# TABLE OF CONTENTS

# 1 Introduction

## 1.1 About This Manual

This Getting Started Manual has been developed to assist first time users of the following boards with software installation, hardware set up, and starting a sample project for the following boards:

Step by step instructions cover Arinc 664 Windows Board Software Package (BSP) installation, board installation, and driver adaptation. The BSP provides users with current documentation, board drivers, and sample development projects. Additionally, the BSP provides utility files that are used to upgrade the AIM 664 on-board firmware, which is also described in this manual. The current version of all BSP is available from the download area of the AIM website.

AIM is also a leading designer and manufacturer of other high performance test and simulation modules, data bus analyzer software and systems for MIL-STD-1553 A/B, AFDX/ARINC664, ARINC429, MIL-STD-1760 and CAN/ARINC825 Applications, PANAVIA Serial Link and Fibre Channel. Supported hardware platforms include Peripheral Component Interconnect (PCI), PCI Express (PCIe), Compact PCI (cPCI), Versa Module Eurocard (VME), VME eXtensions for Instrumentation (VXI), PC104+, PC-Card, PCI Mezzanine Card (PMC), Express Card and Universal Serial Bus (USB). Information about all AIM products can be found at http://www.aim-online.com.

## 1.2 How This Manual Is Organized

This manual is organized as follows:

Section 1 "Introduction" contains an overview of this manual.

Section 2 "New BSP And Hardware Installation Procedure" describes the procedure used to install the BSP software and the Arinc 664 hardware on a system that does not have a previous Arinc 664 Windows BSP installed.

Section 3 "BSP Upgrade Procedure" describes the procedure used to upgrade the BSP and the on-board software on a system that has a previous Arinc 664 Windows BSP installed.

Section 4 "Connecting the fdXTap$^{TM}$" describes proper cable configuration to insert the fdXTap$^{TM}$ Network tap into End System-to-End System, End System-to-Switch, and Switch-to-Switch interfaces..

Section 5 "Developing Custom C/C++ Applications" describes how to create custom C/C++ based applications that make use of AIM Arinc 664 Application Programming Interface (API).

## 1.3 Applicable Documents

AIM has developed several documents that may be used to aid the user with other aspects of the ARINC 664 bus interface card. These documents and a summary of their contents are listed below (including this document):

**Arinc 664 Reference Manual** provides the ARINC 664 application developer with detailed information about all functions provided by Arinc664 API. This guide is to be used in conjunction with the ARINC 664 Programmer's Guide.

**Arinc 664 Programmer's Guide** provides the ARINC 664 application developer with high level s/w development information including user application system design concepts, function call guidelines, and sample programs. This guide is to be used in conjunction with the ARINC 664 Reference Manual.

**Arinc 664 Windows BSP Getting Started Manual** assists the first time users of the AIM PCI 664 boards with software installation, hardware set up and starting a sample project for a Windows BSP.

**Arinc 664 Linux BSP Getting Started Manual** assists the first time users of the AIM PCI 664 boards with software installation, hardware set up and starting a sample project for a Linux BSP.

**PBA.pro Bus Analyzer User Manual** provides instruction on how to use the PC based ARINC 664 bus analyzer software package, PBA.pro, to set up and control the Arinc 664 Bus Interface Module.

**Hardware Manuals** – provide the hardware user manual for the specified modules. The documents cover the hardware installation, the board connections the technical data and a general description of the hardware architecture. The following hardware manuals are available:

- APE-FDX Hardware Manual (PCIe module)
- AXC-FDX Hardware Manual (XMC module)
- AMCX-FDX Hardware Manual (PMC module)
- ACE-FDX Hardware Manual (PXIe module)
- API-FDX Hardware Manual (PCI module)
- AMC-FDX Hardware Manual (PMC module)
- fdXTap$^{TM}$ Hardware Manual (USB device)
- APU-FDX Hardware Manual (USB device)
- APX-GNET Hardware Manual (PCIx module)
- APE-GNET Hardware Manual (PCIe module)

# 2 New BSP And Hardware Installation Procedure

This section contains the procedure used to install the BSP software and AIM Arinc 664 hardware on a system that does not have a previous BSP installed. If installing on a system that has a previous version of the BSP installed, the BSP upgrade procedure described in Section 3 "BSP Upgrade Procedure" should be used.

The BSP for the AIM Arinc 664 devices contains reference documentation, board drivers, sample development projects and upgrade utility files. These files are automatically loaded onto the system by installing the BSP software package.

The software is available on DVD from AIM or via download from www.aim-online.com.

## 2.1 Hardware Installation

Please insert the AIM Arinc 664 devices into your system according to the description within the corresponding hardware manual. Then power-on your system and boot your Windows Operating System (OS) installation.

**Note:**
**ESD Warning!!!**
The AIM interface modules are typical electronic devices that are sensitive to electrostatic discharge. It is important to observe normal precautions when handling the board to prevent damage.

## 2.2 BSP Installation

To install the BSP software from the AIM provided DVD or a download from AIM-Online

1. If using the AIM provided DVD, insert it into the drive. The Setup Application will automatically start after a few seconds.

2. If the Setup Application does not start after a few seconds, or the BSP is being installed from an AIM-Online download, the Setup Application must be manually started.

3. Once the Setup Application has started, step through the install wizard to complete the installation.

You will be given the opportunity to choose the directory where the BSP is to be installed. Default locations can also be used.

After successful BSP installation you may choose an automatic device driver and on-board Firmware update for your installed Arinc 664 devices. If you skip this update procedure, you are able to run it manually at any time (See Section 3.2 "Board Software Upgrade").

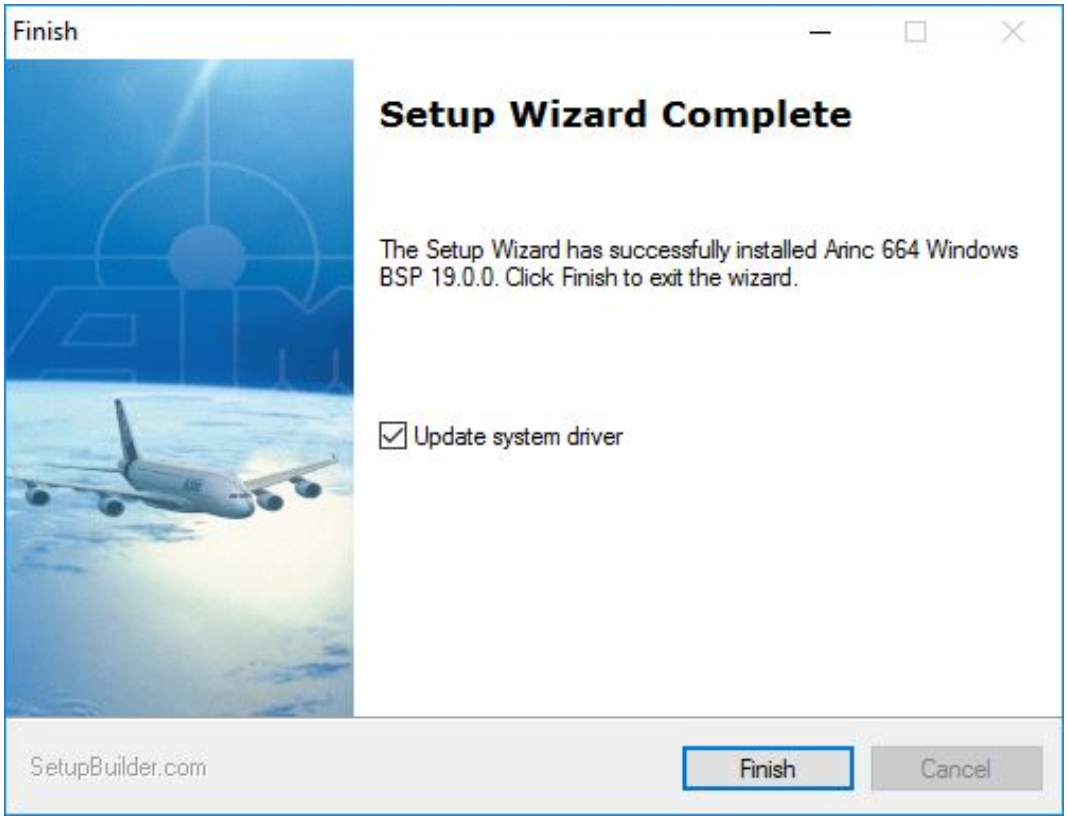


Figure 1: Automatic Driver/Firmware Update

## 2.3 Start Menu Entries

The BSP installation will result in the creation of the following Start Menue entry which can be located by going to *Start* | *All Apps* | *AIM GbmH* | *Arinc 664 x.y.z*. Clicking on this entry will open a Windows Explorer instance that will point to the BSP installation directory.

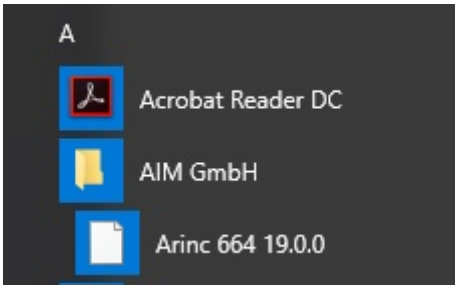


Figure 2: New entries in the start menu

## 2.4 Directory Structure

The BSP installation will result in the creation of the directory structure shown below. The default location *x:\Program Files\AIM_GmbH\Arinc 664 Windows BSP x.y.z* will contain the following directories:

**ANS664** AIM Network Server – for remote access to AIM modules.

**Doc** Software Library Reference manual, Release notes and other documents

**include** Header files used during compilation.

**LabView** LabView application DLL and Libraries

**lib32** DLL and Import Library required to develop an 32 bit application for 32 and 64 bit Windows systems.

**lib64** DLL and Import Library required to develop an 64 bit application for Windows systems.

**Onboard-SW** Update utilities that are used for updating the on-board firmware.

**Sample** Sample project and source files for Microsoft Visual Studio 2013.

**Drivers** Contains drivers for all devices for Windows 32 and 64 Bit systems.

## 2.5 Verifying Proper Installation Of The Device Drivers

To verify proper installation of Windows driver:

1. Open the Windows Device Manager. (e.g. via *Control Panel->Device Manager*)
2. In the Device Manager window, expand the *AIM Components* entry by double clicking on it.
3. Then right click on the AIM Board entry and select Properties.
4. In the Properties window you will get status information that will show if the driver reported any errors at this point.

## 2.6 Installing The Device Drivers

If you need to update the driver manually, the Arinc 664 Windows BSP provides a utility called *dpinst.exe* that will handle the installation process.

When using a *32-Bit* edition of Windows OS, you will have to run the application *x:\Program Files\AIM_GmbH\Arinc 664 Windows BSP x.y.z\Drivers\x86\dpinst.exe*.

On *64-Bit* Windows Editions you need to run *x:\Program Files\AIM_GmbH\Arinc 664 Windows BSP x.y.z\drivers \x64\dpinst.exe*

| **Note:** |
| --- |
| The AIM Arinc 664 devices must be already installed in your system for proper device driver installation via dpinst.exe |

# 3 BSP Upgrade Procedure

This section contains the procedures used to upgrade the BSP for AIM Arinc 664 devices on a system which has a previous version of the BSP installed. This procedure describes the steps which must be taken to upgrade both the host PC driver software and the on-board Firmware.

To properly update the Arinc 664 BSP:

1. Remove the old device driver and optionally the old BSP by following the procedures described in Section 3.1 "Removing Previously Installed Drivers And BSP".
2. Install the new BSP as described in Section 2.2 "BSP Installation".
3. After successful BSP installation, choose to automatically update drivers and Firmware during last step of installation wizard (See figure 1).

**Note:**
When using legacy hardware for which automatic Firmware upgrade is not supported, you have to complete the upgrade by updating the on-board Firmware as described in Section 3.2 "Board Software Upgrade"

## 3.1 Removing Previously Installed Drivers And BSP

To remove a device driver:

1. Open the Windows *Device Manager*. (e.g. via *Control Panel->Device Manager*)
2. In the *Device Manager* window, right click over the AIM Board entry and select **Uninstall**.
3. Now confirm the device uninstall. When available, select *'Delete the driver software for this device'* as shown in figure 3


Figure 3: Device Manager: confirm device uninstall

4. Optionally: remove the BSP by going to the *Control Panel* and selecting *'Uninstall a program'*. In the *'Uninstall a program'* window, right click the Arinc 664 BSP entry and then select the Change/Remove button. Step through the uninstall wizard to complete the removal.

5. After removing the old drivers, install the new BSP and install the new driver as described in Chapter 2.

## 3.2 Board Software Upgrade

The next step is to update the on-board Firmware of your Arinc 664 devices. There are several different procedures depending on the device platform you want to update.

| **Note:** |
| --- |
| Never abort the board update process. Should the update appear to hang, please contact support@aim-online.com or your AIM sales representative. |

#### FPGA and On-Board Components Upgrade

In some cases for a proper BSP update it is necessary to update the onboard hardware components.

| **Note:** |
| --- |
| Please refer to the actual Release Notes if this step must be done or not. |

To do this it is necessary that the system drivers are properly installed and working (card starts without errors). If the system driver is not installed/loaded, please install/load the last AIM Board Software Package and drivers as described in Section 2.6 "Installing The Device Drivers" . This means the drivers which were last working with the board which shall be updated.

#### To upgrade the APE-FDX-2 / AXC-FDX-2 / AMCX-FDX-2 Interface Module Onboard Software and FPGA

- Open the file manager and navigate to:

  For APE-FDX-2:

  *C:\Program Files\AIM GmbH\Arinc 664 Windows BSP x.y.z\Onboard-SW\Ape-Fdx-2*

- Double click on the batch file: update.bat

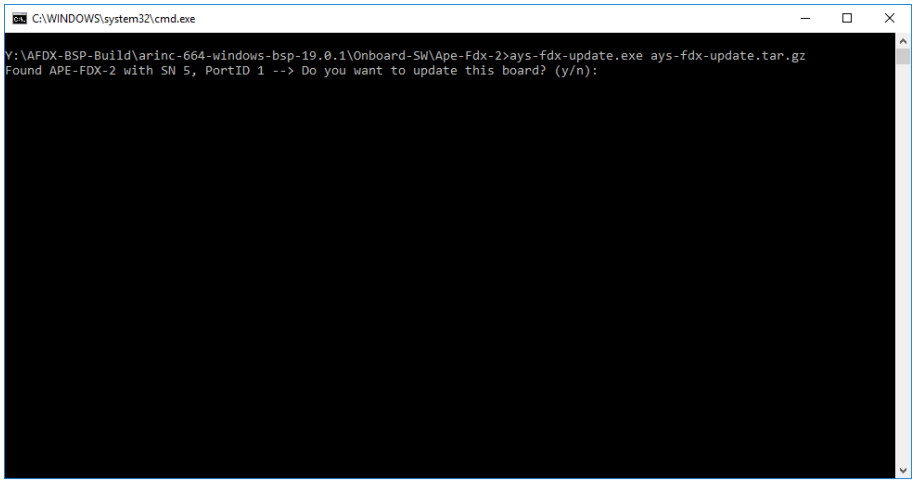


Figure 4: Onboard-SW (APE): update.bat

- Enter “y” To update the boards. This begins the onboard update. This procedure takes a few minutesbut you will get informed about the status of update every second.
- After update has finished you will see the following information.

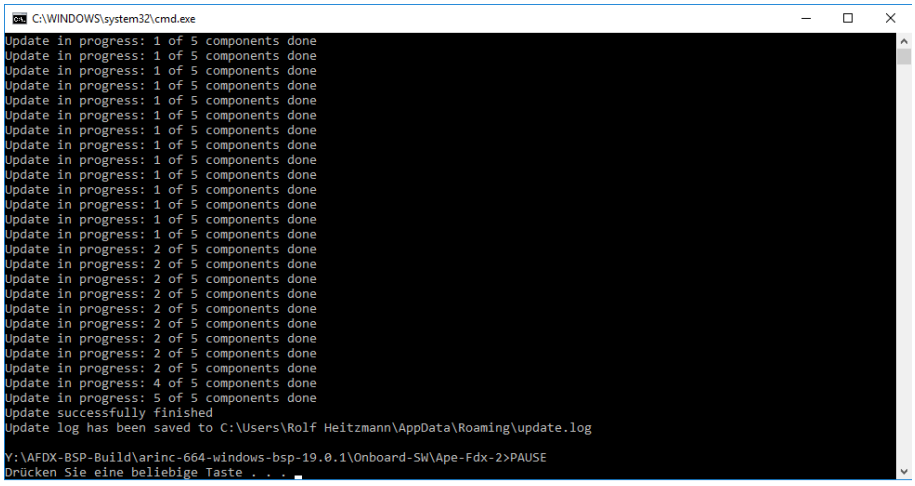


Figure 5: Onboard-SW (APE): update statis

- To get more information about update process and status you can have a look to the file update.log. The location is shown on console output.
- Shutdown the System and turn power off to reboot the card completely (a Windows restart will not reboot all parts, so power has to be turned off!)
- Restart the system, the driver should start without errors

**To upgrade the AMC-FDX-2 or API-FDX-2 Interface Module FPGA**

- Open the file manager and navigate to:

For AMC-FDX-2:

*C:\Program Files\AIM GmbH\Arinc 664 Windows BSP x.y.z\Onboard-SW\Amc-Fdx-2*

For API-FDX-2:

*C:\Program Files\AIM GmbH\Arinc 664 Windows BSP x.y.z\Onboard-SW\Api-Fdx-2*

- Double click on the batch file: update.bat

  This begins the onboard update. This procedure takes a few minutes, depending on system speed.

**GNET Board Software Upgrade**

- Open the file manager and navigate to:

  For GNET-2-4:

  *C:\Program Files\AIM GmbH\Arinc 664 Windows BSP x.y.z\Onboard-SW\GNET-2-4*

- Double click on the batch file: update.bat

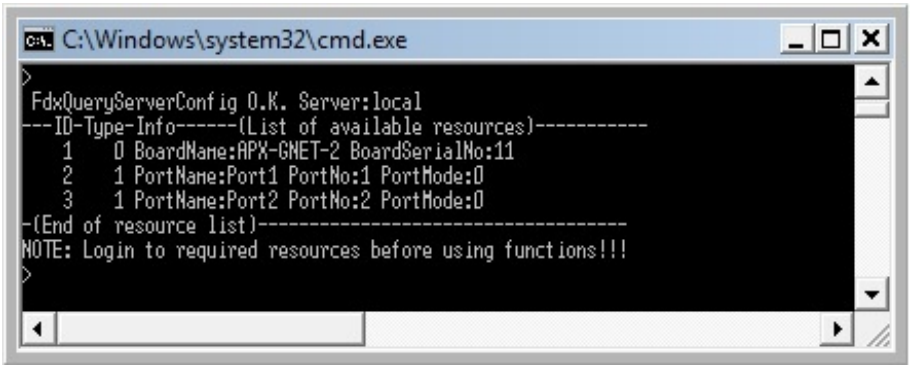


Figure 6: Onboard-SW (GNET): update.bat

- Enter LxU (x=Board Resource Id)

  (Here Board Id is 1 so you have to enter L1U) This begins the PCI interface update. This procedure takes a few minutes, depending on system speed. DO NOT SWITCH POWER OFF DURING UPDATE! A progress indication will appear during update and for each part a update status is.

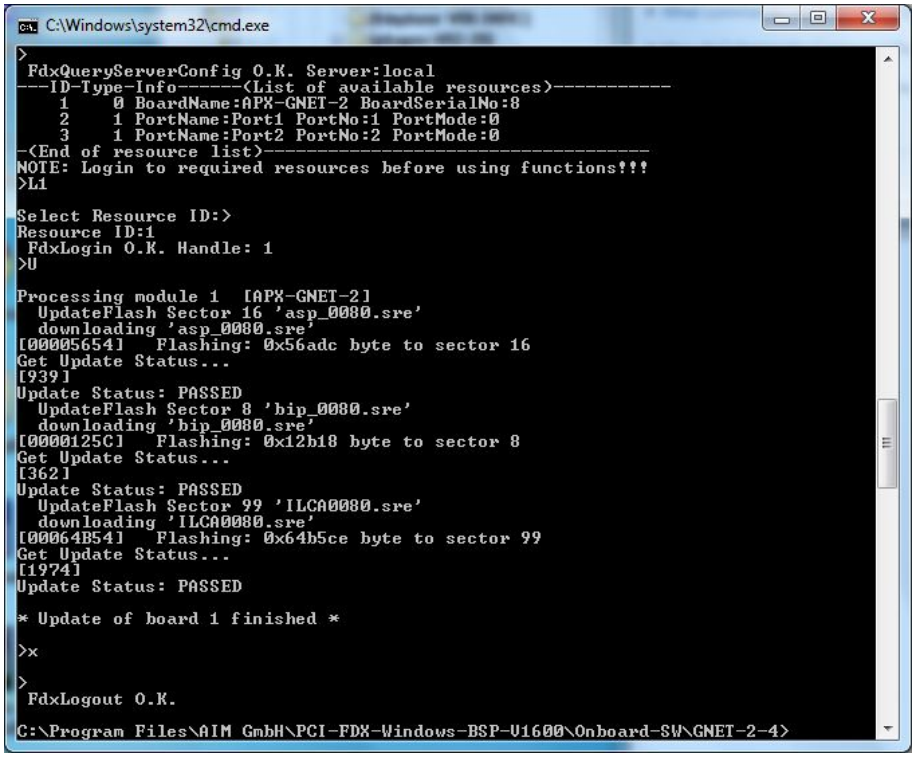


Figure 7: Onboard-SW (GNET): update status

- Shutdown the System and turn power off to reboot the card completely (a Windows restart will not reboot all parts, so power has to be turned off!)
- Restart the system, the driver should start without errors

# 4 Connecting the fdXTap$^{TM}$

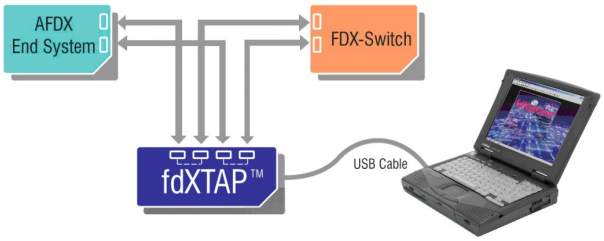


Figure 8: fdXTap: overview

The fdXTap$^{TM}$ can monitor one redundant full duplex network or two independent networks. It routes traffic internally from Port A to Port B and vice versa (for each network). The connection between the ports is internally a direct connection, which means pin 1 of Port A is routed to pin 1 of Port B, and so on. Figure 9 shows the connectivity and cables required for End System-to-End System, End System-to-Switch, and Switch-to-Switch configurations. Figure 10 shows detailed pinouts for each of these configurations. The LEDs at the front panel show the current activity on the network. Both Networks have their own LED array as described below:

**LED Indicators (for each Network)**

| | |
|---|---|
| **Link** | Illuminates if the Network Connection is established successfully |
| **Act A** | if network device connected to Port A sends data to the network device connected to Port B |
| **Act B** | Illuminates if the network device connected to Port B sends data to the network device connected to Port A |
| **Fail** | Serious error has occurred |

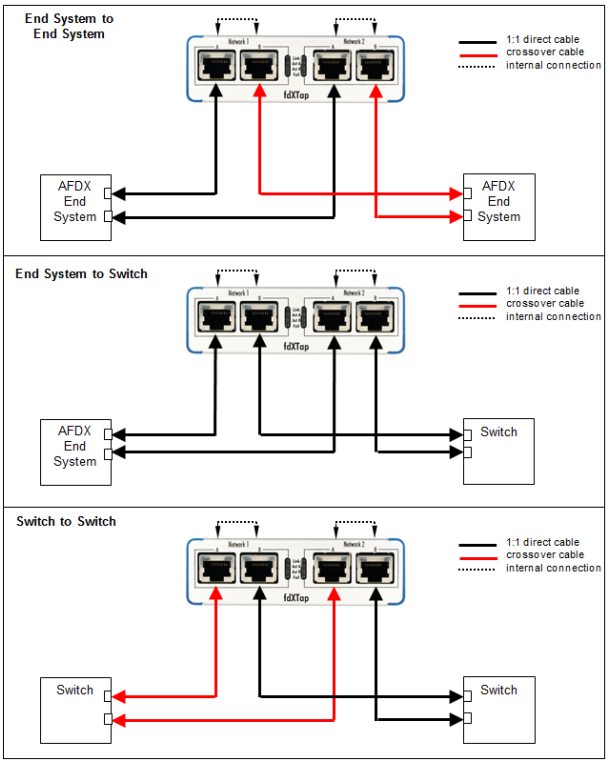


Figure 9: fdXTap: Connection/Cable Configurations

**End System ↔ Switch Connections**

| | | | fdxTap | | | |
|---|---|---|---|---|---|---|
| Pin | End System | | Port A | Port B | | Switch |
| 1 | Tx+ | Direct (1:1) cable | Tx+ In | Tx+ Out | Direct (1:1) cable | Rx+ |
| 2 | Tx- | | Tx- In | Tx- Out | | Rx- |
| 3 | Rx+ | | Rx + Out | Rx+ In | | Tx + |
| 4 | N.C. | | N.C. | N.C. | | N.C. |
| 5 | N.C. | | N.C. | N.C. | | N.C. |
| 6 | Rx- | | Rx- Out | Rx- In | | Tx- |
| 7 | N.C. | | N.C. | N.C. | | N.C. |
| 8 | N.C. | | N.C. | N.C. | | N.C. |

**End System ↔ End System Connections**

| | | | fdxTap | | | |
|---|---|---|---|---|---|---|
| Pin | End System | | Port A | Port B | | End System |
| 1 | Tx+ | Direct (1:1) cable | Tx+ In | Tx+ Out | Crossover cable | Tx+ |
| 2 | Tx- | | Tx- In | Tx- Out | | Tx- |
| 3 | Rx+ | | Rx + Out | Rx - In | | Rx+ |
| 4 | N.C. | | N.C. | N.C. | | N.C. |
| 5 | N.C. | | N.C. | N.C. | | N.C. |
| 6 | Rx- | | Rx- Out | Rx- In | | Rx- |
| 7 | N.C. | | N.C. | N.C. | | N.C. |
| 8 | N.C. | | N.C. | N.C. | | N.C. |

**Switch ↔ Switch Connections**

| | | | fdxTap | | | |
|---|---|---|---|---|---|---|
| Pin | Switch | | Port A | Port B | | Switch |
| 1 | Rx+ | Crossover cable | Tx+ In | Tx+ Out | Direct (1:1) cable | Rx+ |
| 2 | Rx- | | Tx- In | Tx- Out | | Rx- |
| 3 | Tx + | | Rx + Out | Rx - In | | Tx + |
| 4 | N.C. | | N.C. | N.C. | | N.C. |
| 5 | N.C. | | N.C. | N.C. | | N.C. |
| 6 | Tx- | | Rx- Out | Rx- In | | Tx- |
| 7 | N.C. | | N.C. | N.C. | | N.C. |
| 8 | N.C. | | N.C. | N.C. | | N.C. |

Figure 10: fdXTap: Pinout Connectivity for Different Configurations

# 5 Developing Custom C/C++ Applications

This section describes the basic steps for configuring the build of an application that attaches to the AIM Arinc 664 API. For further information regarding the usage and feature set of this API, refer to Arinc 664 Reference Manual and Arinc 664 Programmer's Guide.

## 5.1 Precompiled Sample

To execute a pre-compiled sample program already included in the BSP run:
*C:\Program Files\AIM GmbH\Arinc 664 Windows BSP x.y.z\Sample\Sample.exe* The sample program opens a board and provides a selection of a sub-sample functions to execute. Its source code is included in the BSP and offers a first lead for 664 API programming.

## 5.2 Building Your Own Application

In this demonstration, the Microsoft Visual Studio 2013 Integrated Development Environment (IDE) is used . If you are using other IDEs or compilers, refer to the appropriate documentation on how to add header and library search paths and library dependencies.The general procedure follows these steps:

1. Include the *AiFdx_def.h* header file in your application's modules that want to make use of Arinc 664 functionality.

```
//Include the AIM Arinc 664 API header file
#include "AiFdx_def.h"

int main(void)
{


}
```

**Adding 664 API include path**

2. Add the Arinc 664 BSP's *include* directory to the *"Additional Include Directories"* property in your project's *C/C++ -> General* configuration section.

**Adding _AIM_WINDOWS preprocessor macro**

3. Add the preprocessor macro *_AIM_WINDOWS* to the *"Preprocessor Definitions"* property in your project's *C/C++ -> Preprocessor* configuration section:

**Adding aim_fdx.19.lib dependency**

4. Now add the *aim_fdx.19.lib* library to the *"Additional Dependecies"* property in your project's *Linker -> Input* configuration section:

**Adding library search path**

5. As a last step, append the search path for the Arinc 664 API library to the *"Additional Library Directories"* property in your project's *Linker -> General* configuration section. If building a 32Bit application you have to add the *lib32* directory of the BSP installation directory. In case of building a 64Bit application, *lib64* has to be specified instead.

# 6 Using The LabVIEW VI Interface

The BSP contains a LabVIEW wrapper for the API. For every function of the API there is a LabVIEW Virtual Instrument (VI) with the same name. To avoid redundant information, there is no explicit manual for the LabVIEW VI interface. Information about functionality of the VI can be found in the C/C++ API Reference Manual delivered with this BSP.

## 6.1 Using The Reference Manual

If information about a VI is required, the help of it shows a link to the reference manual. Searching for the name of the VI will bring you to the description of the function.

The VI have connections with the same name for every parameter of the described function and one for each member of every struct used.

Now an example: the fictive function
**AiReturn ApiGetValueFromStruct( AiHandle module, struct somestruct element_in, AiUInt32 *pul_Value)**, where somestruct consists of **AiUInt32 a** and **AiUInt32 b** would have **moduleHandle**, **a** and **b** as input parameters for the VI, in addition to **error in**. The output parameters would be **value** in addition to **error out** and optionally the return value.

# List of Abbreviations

**API** Application Programming Interface

**BSP** Board Software Package

**cPCI** Compact PCI

**IDE** Integrated Development Environment

**OS** Operating System

**PCI** Peripheral Component Interconnect

**PCIe** PCI Express

**PMC** PCI Mezzanine Card

**USB** Universal Serial Bus

**VME** Versa Module Eurocard

**VXI** VME eXtensions for Instrumentation

**VI** LabVIEW Virtual Instrument

# LIST OF FIGURES