ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *Abaco Systems / SBS A429-PCI-8* at our website: ***Click HERE***

SBS Avionics Technologies

Getting Started with the ARINC 429 PCI-8

1: Introduction 1

2: Before You Begin 5

3: Installing the Software 7

4: Installing the Hardware 11

5: Performing System-Specific Installation ... 17

6: Running Unit Test 21

7: Connecting to A429 Bus 41

8: What to Do Next 43

SBS Avionics Technologies
2400 Louisiana Boulevard, NE
AFC Building 5, Suite 600
Albuquerque, NM 87110-4316
Fax: 505-875-0400
Email: sbshelp@sbse.com
http://www.sbs-avionics.com
800-SBS-1553 or 505-875-0600

Applies to models:
A429-PCI-8

© 1999 SBS Technologies, Incorporated. All rights reserved.

Getting Started with the ARINC 429 PCI-8

This document is the intellectual property of SBS Technologies, Inc. (SBS), and contains proprietary and confidential information. Use, disclosure, and reproduction is permitted only under the terms of a software license agreement or explicit written permission of SBS. You should not use this document or it's contents until you have read and agreed to the applicable software license agreement.

This document and its contents are provided as is, with no warranties of any kind, whether express or implied, including warranties of design, merchantability, and fitness for a particular purpose, or arising from any course of dealing, usage, or trade practice.

In no event will SBS be liable for any lost revenue or profits or other special, indirect, incidental and consequential damage, even if SBS has been advised of the possibility of such damages, as a result of the usage of this document and the software for which this document describes. The entire liability of SBS shall be limited to the amount paid by you for this document and its contents.

SBS shall have no liability with respect to the infringement of copyrights, trade secrets or any patents by this document of any part thereof. Please see the applicable software license agreement for full disclaimer or warranties and limitations of liability.

RESTRICTED RIGHTS LEGEND

Use, duplication, or disclosure by the Government is subject to restrictions set forth in subparagraph (c)(1)(ii) of the rights in Technical Data and Computer Software clause at DFARS 252.227-7013.

SBS Technologies, Inc., 2400 Louisiana Blvd. NE, Albuquerque, NM 87110

SBS Technologies, Inc., and its logo are trademarks of SBS Technologies, Inc. All other brand names and product names contained herein are trademarks, registered trademarks, or trade names of their respective holders.

1: Introduction

1.1 Contents of *Getting Started with the A429 PCI-8*

The ARINC 429 (A429) PCI-8 is a single-card, real-time A429 interface module for PC platforms.

This manual is intended to assist you in getting the A429 PCI-8 up and running as quickly as possible. It addresses the following:

- Installing the hardware
- Installing the software
- Configuring the software for your operating system
- Testing
- Basic module operation

This manual assumes that you will be using the Integrated Avionics Programming Library provided by SBS to operate the module.

1.2 Contents of *ARINC User's Manual*

The accompanying *ARINC User's Manual* contains complete details on module programming and operation, including the following:

- A429 programming and structures
- SBS Integrated Avionics Programming Library functions
- Hardware specifications
- System software provided by SBS

1.3 Terminology

Table 1.3.1 defines some of the basic terms used throughout this manual.

Table 1.3.1: Basic Terminology

Term	Meaning
BM	Bus monitor
Bus	A single ARINC 429 unidirectional transmit or receive channel.
Channel	One unidirectional ARINC 429 bus (synonymous with "Bus").
Device	A logical entity that consists of four or eight channels and corresponds one-for-one with a device entry in the *sbs_dev.cfg* configuration file.
Firmware	Program running in the SBS digital signal processor(s) that controls all A429 operations. The firmware must be loaded upon device initialization.
Receive Channel	A single unidirectional ARINC 429 receive bus.
Transmit Channel	A single unidirectional ARINC 429 transmit bus.
Word	A 16-bit value; i.e., two bytes.

1.4 Conventions

The following conventions appear in this document. These conventions may differ from those used in other SBS publications. The subsections listed below describe each convention in more detail:

- Typographic Conventions
- Words Having Special Meaning
- Compound Keystrokes and Menu Selections
- Symbols

1.4.1 Typographic Conventions

Table 1.4.1 shows the typographic conventions used in this document.

Table 1.4.1: Typographic Conventions

Element	Use in Body Text	Use in Procedures
Italic	➢ Document, chapter, section, and topic titles and cross references. ➢ Emphasis.	➢ Filenames and directory paths.
Bold	➢ (Not used in body text.)	➢ Controls, dialogs, menus, and text or numeric fields that appear on the screen. ➢ Keys on your keyboard.
`Courier Roman`	➢ Code examples.	➢ Simulating the appearance of screens.
`Courier Bold`	➢ Library function calls and syntax. ➢ Emphasizing lines of code.	➢ Commands and other information that you type as given.
Angle brackets, e.g.,< >	➢ Enclosing variable information that you type (without the brackets) in place of a dummy variable.	➢ Enclosing variable information that you type (without the brackets) in place of a dummy variable.

The point size of the text varies depending on whether it is used in body text, code examples, notes, screens, or procedures.

1.4.2 Words Having Special Meaning

In procedures, the words "Enter" (or "enter") and "Type" (or "type") have special meanings that are indicated in Table 1.4.2.

Table 1.4.2: Words with Special Meaning

Word	Meaning
Enter	Key in the specified text or variable information and press the **Return** key.
Type	Key in the specified text. Do not press **Return**.

1.4.3 Compound Keystrokes and Menu Selections

Compound Keystrokes

Whenever a procedure instructs you to press multiple keys, a double angle bracket "»" separates the names of the keys. Table 1.4.3 shows an example.

Menu Selections

Whenever a procedure instructs you to select an item from a pull-down menu, a double angle bracket "»" separates the menu items. Table 1.4.3 shows an example.

Table 1.4.3: Examples of Notation for Compound Keystrokes and Menu Selections

Instruction	Meaning
Press **Ctrl** » **Alt** » **Delete**.	Press the **Ctrl**, **Alt**, and **Delete** keys simultaneously.
Select **File** » **Open**.	Select **Open** from the **File** menu.

1.4.4 Symbols

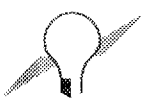
The following symbols appear throughout this manual:



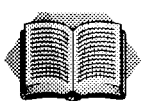
Warning: Paragraphs next to this symbol contain information critical to module operation or to your safety.



Note: Paragraphs next to this symbol contain information important to module operation.



Tip: Paragraphs next to this symbol contain useful tips.



Cross Reference: Paragraphs next to this symbol contain cross references to the *ARINC User's Manual* or to a related page in this manual.



Software Cross Reference: Paragraphs next to this symbol contain cross references to software media included with this product.

2: Before You Begin

2.1 What You Should Have Received

- ARINC 429 PCI-8 interface module
- Loopback assembly
- SBS Resource CD
- 3.5-inch Firmware disk, MS-DOS format
- This *Getting Started with the ARINC 429 PCI-8* manual
- The *ARINC User's Manual*



Note: If you are unable to use the MS-DOS format, contact SBS Technical Support for additional media options.

2.2 Unpacking the Card



Warning: This is an electronic product that is sensitive to electrostatic discharge. Take normal precautions in handling the card to prevent damage.

- Carefully unpack the card and inspect it for physical damage that might have occurred during shipping.
- If you have a damaged card, contact the SBS technical support group that handles maintenance, repairs, and warranties in Albuquerque. When you call us, give us the serial number of your card, and have the card available in case we have questions about its condition.

2.3 What You Will Need

The A429 PCI-8 product package includes everything required to operate the card on your chassis except for appropriate connectors and wiring to connect to an actual ARINC 429 bus.

These parts are available from the following vendor:

MilesTek
1506 Interstate 35W
Denton, TX 76207-2402
800-524-7444 or 940-484-9400
FAX: 940-484-9402

2.4 Power Requirements

The power requirements for the PCI-8 are listed in Table 2.4.1.

Table 2.4.1: Power Requirements for the PCI-8

Supply (volts)	Typical Idle Current (milliamps)	Maximum Current[a] (milliamps)
+5	570	630
+12	130	330
–12	90	280

Notes:
[a]For the maximum current, each channel output is loaded with 20 receiver loads, and each receiver load is assumed to be 8 kΩ.

3: Installing the Software

Descriptions of the provided software disks and instructions on software installation are included in the following sections:

- Support Software
- Copying the Software to Your Host System

3.1 Support Software

SBS provides support software for its ARINC 429 products as part of the Integrated Avionics Programming Library on the following media:

- SBS Resource CD
- 3.5-inch Firmware disk

SBS Resource CD The SBS Resouce CD contains the following:

- Integrated Avionics Programmming Library, including C library source files, DLLs, sample applications, and Unit Test executable
- Device drivers necessary to support the interface between the libraries and your PC or VME computer system
- SBS PASS demos
- Product documentation

Firmware Disk The Firmware disk contains the firmware file that must be downloaded to the A429 PCI-8 card upon initialization.

3.2 Copying the Software to Your Host System

Use the following instructions to copy the software to the system in which you are installing the A429 PCI-8 card. Separate sets of instructions for PC systems and for non-PC systems are included in the following two sections:

- PC Operating Systems
- UNIX-Based (Non-PC) Operating Systems

3.2.1 PC Operating Systems

Unless you specify otherwise, the installation batch files will place the Integrated Avionics Programming Library and operating system dependent software in the *c:\sbs_ver6.10* folder (directory) in your Windows 95 or Windows NT environment.

1. Install the library by completing the following steps:
   - Insert the SBS Resource CD into your CD-ROM drive.
   - Start the Explorer and navigate to the CD.
   - Double-click on the Library folder.
   - Double-click **Setup.exe** from the Explorer.
   - Follow the instructions that appear on the screen.

   If you select all of the defaults, the directory structure shown in Figure 3.2.1 is created.

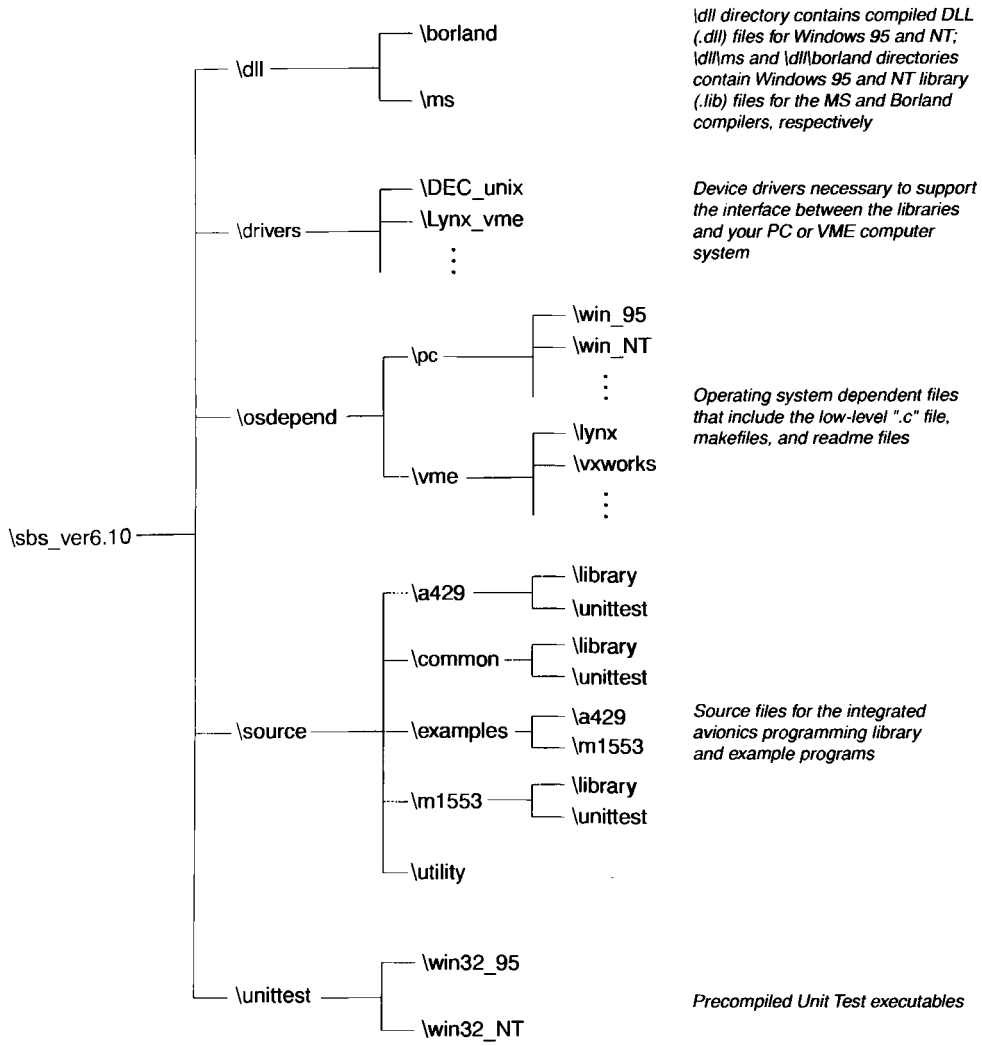


Figure 3.2.1: Default Directory Structure

2. Create the SBS user directory structure shown in Figure 3.2.3 by adding *\working* and *\firmware* directories under the *\sbs_ver6.10* directory.

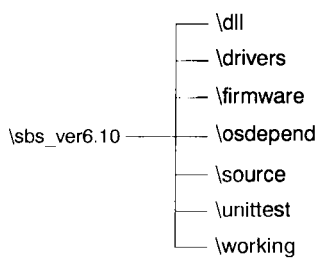
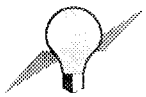


Figure 3.2.2: User Directory Structure



Note: The *\working* directory is a place for you to create and work with files without disturbing other files and directories.

3. Copy the firmware file(s) directly from the Firmware disk to the *\sbs_ver6.10\firmware* directory.



Tip: See the chapter *Performing System-Specific Installation* on page 17 to continue the software installation process.

3.2.2 UNIX-Based (Non-PC) Operating Systems

To install the interface libraries and operating system software on your UNIX-based system, you must first copy the software to a PC system on your network, then transfer it via ftp to the UNIX-based system in which you are installing the PCI card. The steps below guide you through the process.

1. Copy the software to a PC system on your network by following procedure steps 1–3 beginning on page 8 in the preceding subsection, *PC Operating Systems*. Make sure to select the files needed for your target UNIX-based system, not the PC system.

2. Enter **`ftp`** to access your host system.

3. Create the directory structure shown in Figure 3.2.3 using the ftp **`mkdir`** utility.

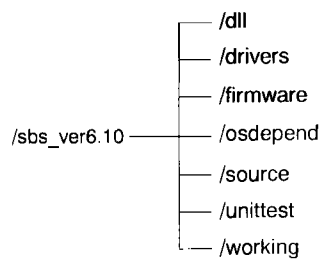


Figure 3.2.3: User Directory Structure

4. Use the ftp **`put`** or **`mput`** utility to transfer the interface library files from the PC environment to the host system (into the above tree structure).

4: Installing the Hardware

The sections in this chapter provide instructions on how to configure and install the A429 PCI-8 hardware. The sections are the following:

- Installing the Card
- Making Auxiliary Connections for External Signals
- Attaching the Loopback Assembly

4.1 Installing the Card

Warning: This is an electronic product that is sensitive to electrostatic discharge. Take normal precautions in handling the board to prevent damage.

To install the A429 PCI-8 card, follow the procedure below.

1. Ensure that power is disconnected from your computer.
2. Insert the A429 PCI-8 card into one of the PCI slots in your system.
3. Power up the host system and verify that both the status and activity LEDs are off.

   Figure 4.1.1 and Figure 4.1.2 show the locations of the LEDs on the A429 PCI-8. If either of these LEDs is not in the proper state, contact SBS Technical Support.

Note: During board operation, refer to Figure 4.1.1 and Table 4.1.1 for information on connector pinouts.

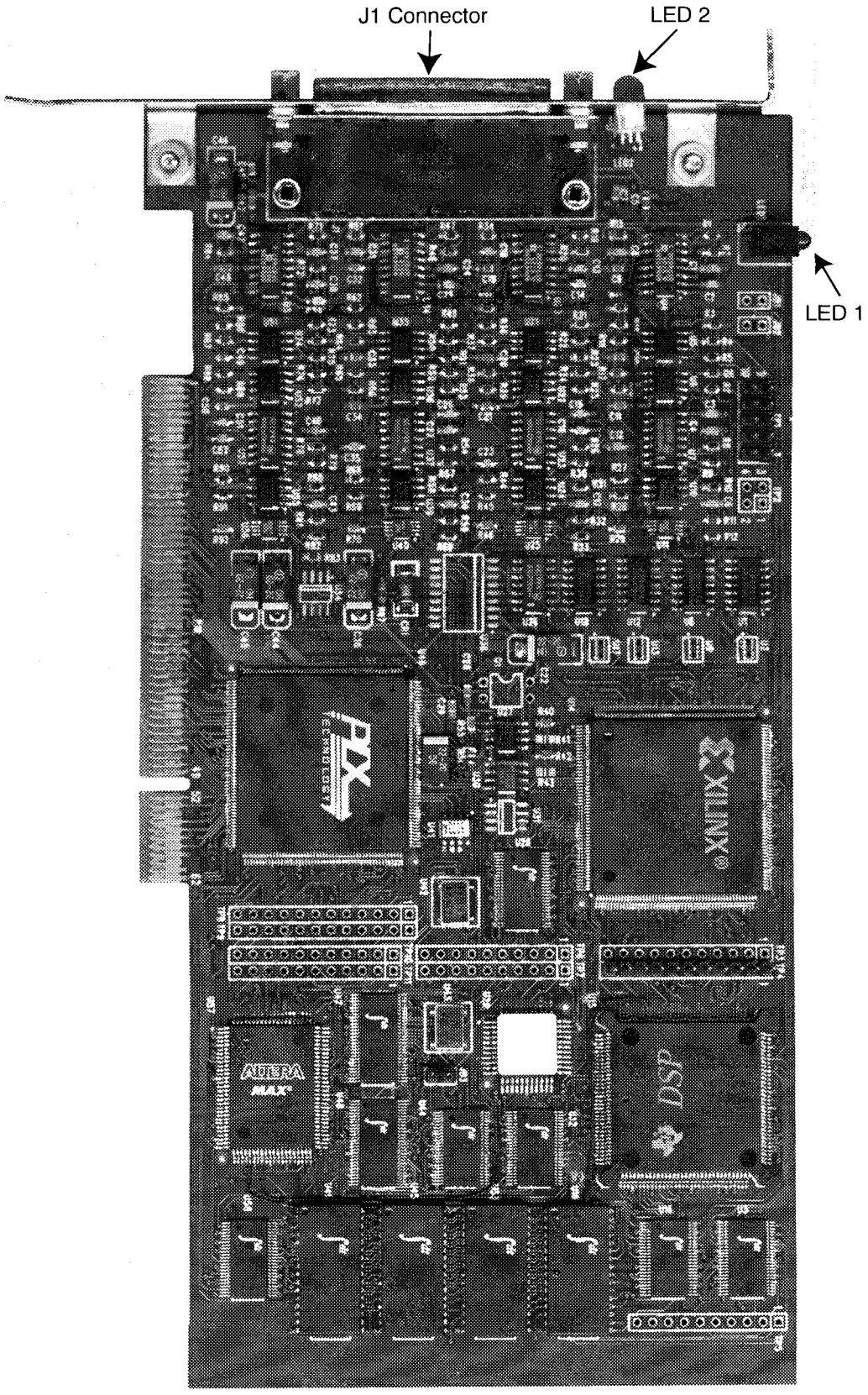


Figure 4.1.1: A429 PCI-8 Board

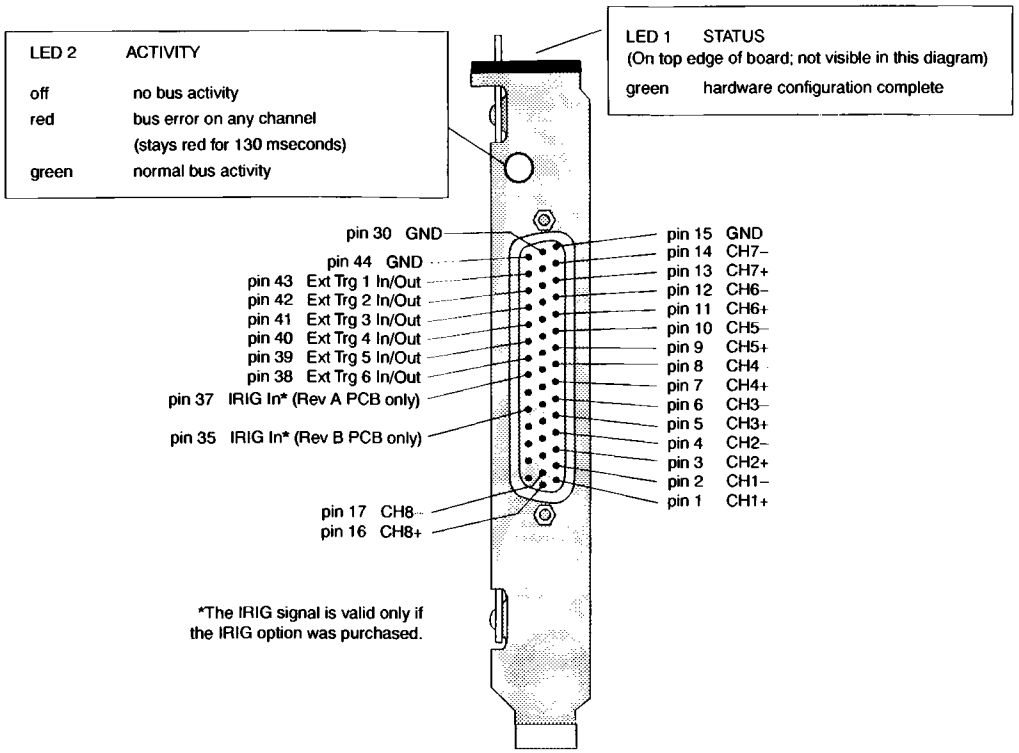


Figure 4.1.2: Single-Device A429 PCI-8, Front Panel Pinouts & LEDs

Notes: LED 2 indicates activity for all eight ARINC channels.

If the LED glows red, one or more channels have had a bus error.

If the LED glows green, one or more ARINC channels are active.

Table 4.1.1: Pinouts for Connector J1

Pin	Description	Pin	Description
1	Channel 1+	23	Unused
2	Channel 1–	24	Unused
3	Channel 2+	25	Unused
4	Channel 2–	26	Unused
5	Channel 3+	27	Unused
6	Channel 3–	28	Unused
7	Channel 4+	29	Unused
8	Channel 4–	30	Ground
9	Channel 5+	31	Unused
10	Channel 5–	32	Unused
11	Channel 6+	33	Unused
12	Channel 6–	34	Unused
13	Channel 7+	35	IRIG In (Rev B PCB only)
14	Channel 7–	36	Reserved
15	Ground	37	IRIG In (Rev A PCB only)
16	Channel 8+	38	Ext Trg 1
17	Channel 8–	39	Ext Trg 2
18	Unused	40	Ext Trg 3
19	Unused	41	Ext Trg 4
20	Unused	42	Ext Trg 5
21	Unused	43	Ext Trg 6
22	Unused	44	Ground

4.2 Making Auxiliary Connections for External Signals

In order to connect external signals to the A429 PCI-8 board, you need to make the proper connections to the 44-pin connector shown in Figure 4.1.1 on page 12.

IRIG Signal If you are using an external IRIG signal as your timing source, make the following connections:

1. Connect the IRIG signal to IRIG In (pin 35 on the Rev B PCB, or pin 37 on the Rev A PCB) on the 44-pin connector.
2. Connect the ground to one of the available ground pins on the 44-pin connector.



Note: The IRIG signal applies only if you purchased the IRIG option.

External Trigger If you are using an external trigger (in/out), make the following connections:

1. Connect the trigger signal to one of the available external trigger pins (pins 38–43) on the 44-pin connector.
2. Connect the ground to one of the available ground pins on the 44-pin connector.

4.3 Attaching the Loopback Assembly

The loopback assembly included with the A429 PCI-8 module is required for testing the board. It provides a loopback connection from channel 1 to channel 5, channel 2 to channel 6, channel 3 to channel 7, and channel 4 to channel 8. Table 4.3.1 lists the PCI-8 loopback connector pinout.

Before Testing the A429 PCI-8 Attach the loopback assembly to the P1 connector on the front panel of the board (see Figure 4.3.1).



Cross Reference: Before operating the A429 PCI-8 on an A429 bus, see the instructions on page 41 of this manual for connecting the A429 PCI-8 to an ARINC bus.

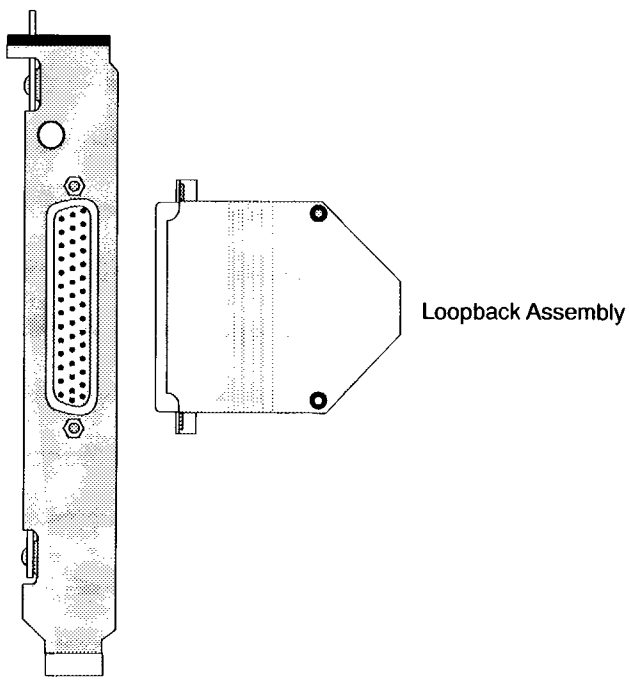


Figure 4.3.1: Attaching the Loopback Assembly to the A429 PCI-8

Table 4.3.1: A429 Loopback Connector Pinouts

Receive Channels		Connected to	Transmit Channels	
Description	**Pin**		**Description**	**Pin**
Channel 1+	1	→	Channel 5+	9
Channel 1–	2	→	Channel 5–	10
Channel 2+	3	→	Channel 6+	11
Channel 2–	4	→	Channel 6–	12
Channel 3+	5	→	Channel 7+	13
Channel 3–	6	→	Channel 7–	14
Channel 4+	7	→	Channel 8+	15
Channel 4–	8	→	Channel 8–	16

5: Performing System-Specific Installation

The sections in this chapter provide instructions on how to install the files specific to your operating system and platform. Instructions are provided for the following systems:

- Windows 95
- Windows NT
- IRIX 6.4/6.5



Note: If you are using a system not covered in this chapter, see Appendix D of the ***ARINC User's Manual*** for instructions on installing the files for your system.

5.1 Windows 95

1. After you have inserted the PCI-8 card into one of the PCI slots in your system, a **New Hardware Found** dialog appears. Select **Driver from disk provided by hardware manufacturer** and click **OK**.
2. Click the **Browse** button.
3. Navigate to the drive where you installed the SBS software.
4. Select **sbs_ver6.10\drivers\sbs.inf**, and click **OK**.
5. If you are prompted to specify the location of the virtual device driver *sbspci.vxd*, click the **Browse** button, select **sbs_ver6.10\drivers\sbspci.vxd**, and click **OK**.



Note: Parameters such as I/O base, base memory address, and interrupt vector are configured by the computer. These parameters are ignored in the *sbs_dev.cfg* (or *dev_cfg.h*) configuration file.

5.2 Windows NT



Note: You must have Administrator privileges on your account before performing this operation.

1. Copy the *sbspci32.sys* file from the *\sbs_ver6.10\drivers* directory to the *winnt\system32\drivers* directory.

2. If you are using a single card, skip to step 4. If you are using multiple cards, you must edit the *sbspci32.ini* file to include one copy of the driver for each card (the default file is set up for a single card). Open the *sbspci32.ini* file and edit it as follows:

   - Duplicate the last two lines of the file until you have one pair of lines there for each card.
   - Change the **sbspci320** line to **sbspci321** in the second pair (for the second card), to **sbspci322** in the third pair (if you have a third card), and so on.
   - Save the file.

3. Enter DOS mode and change to the *\sbs_ver6.10\drivers* directory.

4. Enter the data into the registry by using the *regini.exe* program provided on the driver disk. (This program is also supplied with the Windows NT Driver Developer's Kit.) Execute *regini.exe* under a user account that has system administration privileges, using *sbspci32.ini* as an argument (**regini sbspci32.ini**).

   All changes to the registry are printed on the screen. If nothing has changed in the registry, nothing is printed.

5. Reboot the system.

6. If there is a problem starting the driver, you may receive a Service Control Manager dialog box with the message, "At least one service or driver failed during system startup. Use Event Viewer to examine the event log for details." For details on the error, complete the following steps:

   - Select **Start** » **Settings** » **Programs** » **Administrative Tools (Common)** » **Event Viewer** to access the Event Viewer System Log dialog box.
   - Double-click on the line(s) with the red stop sign for event details.

7. After the system has rebooted successfully, select **Start** » **Settings** and then choose **Control Panel**.

8. Double-click on the **Devices** icon.

9. Scroll down the list of devices until you find **sbspci32**. The Status column for sbspci32 should say "Started," and the Startup column should say "Automatic."

10. Select **Start** » **Programs**, then **Administrative Tools (Common)**, and then **Windows NT diagnostics**.

11. Click on the **Resources** tab.

12. Click on the **Devices** button and then double-click on **sbspci32**.

13. Verify that the specified card driver is present and that all interrupts and I/O addresses are appropriate for your system. Check with your system administrator for information on the proper settings for your system.

5.3 IRIX 6.4/6.5

1. Refer to the man page for usrpci(7m) to become familiar with SGI's PCI support.
2. Locate the path to the appropriate PCI device (e.g., */hw/module/4/slot/io2/pci_xio/0/userpci*). This path is unique for each machine.
3. Create a temporary directory and copy all files in the */sbs_ver6.10/osdepend/pc/irix6_4/sbsdrvr* directory to the temporary directory.
4. Edit the *vmconfig.c* file.
   - ➢ Modify **MEM_DEV_PATH** to reflect the device path listed above.
   - ➢ Save the file.
5. Enter the following commands to compile *vmconfig.c* and create the *vmconfig* executable:

   ```
   cc vmconfig.c
   mv a.out vmconfig
   ```

6. Enter the following command to modify the privileges of all the files in PCI device directory to include read and write privileges for group and other:

   ```
   chmod go+rw <path>/*,
   ```

   where **<path>** is the device path listed above.
7. Configure the PCI card by entering the following:

   ```
   vmconfig 0 0
   ```

   The information and prompt shown in Figure 5.3.1 appear.

```
system pagesize is xxxxx bytes
This process's map window is xxxxxxxx, hex.

Command                      Action
-----------------------      --------------------------
<Return>                     read next address
- <Return>                   read previous address
<Space > < Return>           read address again
NNNNNNNN <Return>            write hex value
?                            help
q or .                       quit

Address                      Contents
----------                   -----------
00000000:                    906d10b5    .m..      >
```

Figure 5.3.1: PCI Configuration Screen

8. Press **Return** as many times as necessary until the address **0000012c:** appears.
9. Verify that the content of this address is **155312df**.
10. Address **00000018** contains the base address for the PCI card. Write this value in the space provided below for future reference:

    Base Address: _____

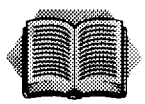
6: Running Unit Test

To verify that the A429 PCI-8 is properly installed and operational, use the instructions in the following sections to run the Unit Test application:

- Introduction
- Using the Combined (1553 and A429) Unit Test Executable
- Setting up the Device Configuration File
- Starting Unit Test
- Opening the A429 Device
- Loading the Firmware
- Running Built-In Tests (BITs)
- Initializing the A429 Device
- Exiting Unit Test



Note: You must have an ANSI compatible terminal or driver in order to run Unit Test.



Cross Reference: For more information about the Unit Test application, refer to Appendix E in the *ARINC User's Manual.*

6.1 Introduction

Executables for Windows 95 and NT for the combined (1553 and A429) Unit Test application are supplied in the *\sbs_ver6.10\unittest\win32_95* or *\sbs_ver6.10\unittest\win32_NT* directory, respectively. If you are using one of these operating systems with the A429 PCI-8 card and wish to use the combined Unit Test, proceed to the next section. If you do not have access to the distributed Unit Test executables, if there is not a precompiled executable for your operating system, or if you wish to run a stand-alone A429 Unit Test, use the instructions in Chapter 16 of the *ARINC User's Manual* to compile a new Unit Test.

6.2 Using the Combined (1553 and A429) Unit Test Executable

6.2.1 Operating Systems with File Systems

To use the precompiled, combined Unit Test for Windows 95 or NT, complete the following steps:

1. Copy the Unit Test executables from the *\sbs_ver6.10\unittest\<os>* directory to the *\sbs_ver6.10\working* directory (where *<os>* is *win32_95* or *win32_NT*).
2. Copy the *sbs_dev.cfg* file from the *\sbs_ver6.10\source\common\library* directory to the *\sbs_ver6.10\working* directory.
3. Copy the firmware files from the *\sbs_ver6.10\firmware* directory to the *\sbs_ver6.10\working* directory.
4. Set up the *sbs_dev.cfg* file as described in the next section, *Setting up the Device Configuration File*.

6.2.2 Operating Systems without File Systems

1. Refer to Chapter 16 of the *ARINC User's Manual* to determine the necessary files and appropriate compiler directives. Copy the appropriate files into the *\sbs_ver6.10\working* directory.
2. Set up the *dev_cfg.h* file as described in the next section, *Setting up the Device Configuration File*.
3. Compile your Unit Test application as described in Chapter 16 of the *ARINC User's Manual*.

6.3 Setting up the Device Configuration File

Device parameter values for your SBS device(s) must be defined in one of the following two files:

- *sbs_dev.cfg*, if your operating system has a file system
- *dev_cfg.h*, if your operating system does not have a file system

The *sbs_dev.cfg* device configuration file is an ASCII text file containing information that your application uses to initialize one or more SBS devices. The SBS device's initialization procedure calls a parser function (`sbs_parse_config_file()`) that reads the information in the configuration file. The parser function must be able to correctly read and verify the information for an SBS device before further library calls can be made. Because the `sbs_parse_config_file()` function reads the *sbs_dev.cfg* file at run-time, you can modify *sbs_dev.cfg* without recompiling the application.

If your operating system does not have a file system, you should use the *dev_cfg.h* file instead of the *sbs_dev.cfg* file. The *dev_cfg.h* file is an array of strings with the same format and keywords as the *sbs_dev.cfg* file. It is parsed in the same manner as the *sbs_dev.cfg* file. However, you must recompile your application each time you edit *dev_cfg.h* for your changes to take effect.



Software Cross Reference: The *sbs_dev.cfg* and *dev_cfg.h* configuration files are delivered with the SBS Integrated Avionics Library distribution disks under the *\sbs_ver6.10\source\common\library* directory.

A description and examples of the *sbs_dev.cfg* and *dev_cfg.h* files are provided in the following sections:

- Format
- Keywords
- sbs_dev.cfg File
- dev_cfg.h File

6.3.1 Format

The format of the *sbs_dev.cfg* and *dev_cfg.h* files resembles an *.ini* file found on most PCs. Values contained in this file vary based on your SBS device(s), system hardware, and operating system. Every device requires a separate set of keyword entries in the configuration file. (Thus, each single-device card requires a single set of keyword entries, and each multidevice card requires multiple sets.)

The file is set up as follows:

- An equal sign (=) links each keyword with a value.
- The variable **<num>** represents a numerical value.
- The variable **<name>** represents a case-sensitive character string.
- The parser recognizes the following characters appended to the value string:
  - **b** (binary)
  - **o** (octal)
  - **h** (hex)
  - **d** (decimal; the default if no character is present)
- A semicolon at the beginning of a line denotes a comment line, and the parser ignores the entire line.

6.3.2 Keywords

Table 6.3.1 provides a description of the keywords in the *sbs_dev.cfg* file that are needed for configuring the A429 PCI-8 card.

Table 6.3.1: Required Configuration File Keywords for the PCI-8

Keyword=<Specifier>	Required for	Description
[DEVICE=<num>]	All operating systems and platforms	This line is required before all other keywords for a device. **<num>** represents a unique number, starting at 1, that identifies the SBS device.
device_type=<name>	All operating systems and platforms	**<name>** specifies the type of SBS device being used. Set the value to **A429_PCI_8**.
filename=<name>	Windows NT and some UNIX systems (not required for vxWorks)	**<name>** specifies the UNIX or Windows NT device driver name. It contains a text string with the name of the device driver node associated with the SBS device. For UNIX, **<name>** is the filename of the driver the */dev* directory. For IRIX 6.4/6.5 systems, **<name>** is the XIO pathname that points to the memory space of the PCI bus for your card. For Windows NT, this value must correspond exactly to the name of the instantiated driver for the device, which can be found in the device directory using the Device Manager. It is of the form etc.
firmware=<name>	All operating systems and platforms	**<name>** specifies the firmware filename. If you specify **NO_FILE_SYSTEM** in *sbs_sys.h*, the firmware must be embedded in the *firmware.h* file, which contains one or more **firmware_<num>[FIRMWARE_LENGTH_<num>]** data arrays (where **<num>** is a digit from 1 to 9). Set **<name>** equal to **firmware_<num>**. You must use the firmware **<num>** value appropriate for that device. Devices with flash memory still require this field to reprogram the flash memory.  **Note:** Verify that the firmware filename listed in the *sbs_dev.cfg* file is the same as that of the firmware file provided with the software distribution.
num_receive=<num>	All operating systems and platforms	**<num>** specifies the number of receive channels for an A429 device. This is set by the user in the range of 0 to 8. Receive channels are numbered starting with 1. The sum of **num_transmit** and **num_receive** must equal the total number of channels for that device.
num_transmit=<num>	All operating systems and platforms	**<num>** specifies the number of transmit channels for an A429 device. This is set by the user in the range of 0 to 8. Transmit channels are numbered starting at **num_receive** +1. The sum of **num_transmit** and **num_receive** must equal the total number of channels for that device.
relative_position=<num>	Windows 95 only	**<num>** is the sequence number of the PCI card as found in the PCI bus slots. The first board is 0, the second board is 1, etc.

6.3.3 *sbs_dev.cfg* File



Note: If your operating system does not have a file system, you must use the *dev_cfg.h* file instead of *sbs_dev.cfg*. See page 31 for information on *dev_cfg.h*.

If your operating system has a file system, the device parameter values for each SBS device must be defined in the *sbs_dev.cfg* file. Modify *sbs_dev.cfg* as follows:

1. Change directories to the *\sbs_ver6.10\working* directory if you are not already there.
2. Edit the *sbs_dev.cfg* file and verify that the settings are correct for the device type, firmware file(s), and number of devices you are using.
3. Save the file.

Examples of the *sbs_dev.cfg* file appear on the following pages for an A429 PCI-8 card running under the following systems:

- Windows 95
- Windows NT
- IRIX 6.5

In these examples, the arrows denote the required parameters. All remaining lines are commented out (preceded by a semicolon). If you are running your card under an operating system other than those in the examples, refer to Table 6.3.1 to determine the keywords that your operating system requires.

Windows 95

```
;*=====================================================================
;* Copyright (c) 1999 by SBS Technologies, Inc.
;*                       2400 Louisiana Blvd., NE
;*                       AFC Building 5, Suite 600
;*                       Albuquerque, New Mexico  87110
;*                       Technical support (toll free)
;*                       877-TECHSBS (877-832-4727)
;*=====================================================================
;*  SBS INTEGRATED AVIONICS LIBRARY  Version 6.1  (08 Mar. 1999)
;*=====================================================================
;*
;* NAME:  sbs_dev.cfg
;*
;* DESCRIPTION:
;*    This file contains the information required to initialize one or
;*    more SBS device(s).
;*
;* FUNCTIONS:   None
;*
;*=====================================================================

[DEVICE=1]
```

```
; Base Memory Address Segment in hex (0-FFFFFFFFh)
; base_address=D0000h

; PCMCIA DSP Bootloader program filename (default bootload.txt)
; bootloader=bootload.txt

; Device Type (M1553_cPCI_3U,     A429_PC8,
;              M1553_cPCI_6U_1,   A429_PC16_1,
;              M1553_cPCI_6U_2,   A429_PC16_2,
;              M1553_FW5000,      A429_PC104,
;              M1553_PC104,       A429_PCMCIA,
;              M1553_PC3_1,       A429_V2_1,
;              M1553_PC3_2,       A429_V2_2,
;              M1553_PCI_1,       A429_cPCI_3U,
;              M1553_PCI_2,       A429_cPCI_6U_1,
;              M1553_PCMCIA,      A429_cPCI_6U_2,
;              M1553_PMC,         A429_PCI_8,
;              M1553_V6_1,        A429_PCI_16_1,
;              M1553_V6_2,        A429_PCI_16_2,
;              M1553_V6_3,
;              M1553_V6_4)
device_type=A429_PCI_8

; PCMCIA DSP BIT/Application program filename
; dsp_file=PCMDSP.TXT

; UNIX/NT Device Driver Name
; filename=sbspci320

; Firmware filename
firmware=f050b.dat

; VME Interrupt Vector in hex (0-FFh)
; int_vector=FFh

; PC Base I/O Address in hex (300h-3FFh)
; io_base=380h

; Interrupt Request Level in hex (0-0Fh)
; irq_level=0Ah

; Number of Receive channels
num_receive=4

; Number of Transmit channels
num_transmit=4

; Product Type (ABI,ASF)
; product=ABI

; Relative position of PCI card under Windows 95
relative_position=0

; PCMCIA Socket number (0-7)
; socket=0

; PC Memory Window Size (16k, 64k)
; window_size=16k

; PCMCIA Xilinx downloadable filename (.bit extension)
; xilinx_file=00670002.BIT
```

Windows NT

```
;*=====================================================================
;* Copyright (c) 1999 by SBS Technologies, Inc.
;*                       2400 Louisiana Blvd., NE
;*                       AFC Building 5, Suite 600
;*                       Albuquerque, New Mexico  87110
;*                       Technical support (toll free)
;*                       877-TECHSBS (877-832-4727)
;*=====================================================================
;*  SBS INTEGRATED AVIONICS LIBRARY  Version 6.1  (08 Mar. 1999)
;*=====================================================================
;*
;* NAME:  sbs_dev.cfg
;*
;* DESCRIPTION:
;*    This file contains the information required to initialize one or
;*    more SBS device(s).
;*
;* FUNCTIONS:   None
;*
;*=====================================================================

[DEVICE=1]

; Base Memory Address Segment in hex (0-FFFFFFFFh)
; base_address=D0000h

; PCMCIA DSP Bootloader program filename (default bootload.txt)
; bootloader=bootload.txt

; Device Type (M1553_cPCI_3U,     A429_PC8,
;              M1553_cPCI_6U_1,   A429_PC16_1,
;              M1553_cPCI_6U_2,   A429_PC16_2,
;              M1553_FW5000,      A429_PC104,
;              M1553_PC104,       A429_PCMCIA,
;              M1553_PC3_1,       A429_V2_1,
;              M1553_PC3_2,       A429_V2_2,
;              M1553_PCI_1,       A429_cPCI_3U,
;              M1553_PCI_2,       A429_cPCI_6U_1,
;              M1553_PCMCIA,      A429_cPCI_6U_2,
;              M1553_PMC,         A429_PCI_8,
;              M1553_V6_1,        A429_PCI_16_1,
;              M1553_V6_2,        A429_PCI_16_2,
;              M1553_V6_3,
;              M1553_V6_4)
device_type=A429_PCI_8

; PCMCIA DSP BIT/Application program filename
; dsp_file=PCMDSP.TXT

; UNIX/NT Device Driver Name
filename=sbspci320

; Firmware filename
firmware=f050b.dat

; VME Interrupt Vector in hex (0-FFh)
; int_vector=FFh

; PC Base I/O Address in hex (300h-3FFh)
; io_base=380h

; Interrupt Request Level in hex (0-0Fh)
; irq_level=0Ah

; Number of Receive channels
num_receive=4
```

```
    ; Number of Transmit channels
    num_transmit=4

    ; Product Type (ABI,ASF)
    ; product=ABI

    ; Relative position of PCI card under Windows 95
    ; relative_position=0

    ; PCMCIA Socket number (0-7)
    ; socket=0

    ; PC Memory Window Size (16k, 64k)
    ; window_size=16k

    ; PCMCIA Xilinx downloadable filename (.bit extension)
    ; xilinx_file=00670002.BIT
```

IRIX 6.5

```
;*=========================================================================
;* Copyright (c) 1999 by SBS Technologies, Inc.
;*                       2400 Louisiana Blvd., NE
;*                       AFC Building 5, Suite 600
;*                       Albuquerque, New Mexico  87110
;*                       Technical support (toll free)
;*                       877-TECHSBS (877-832-4727)
;*=========================================================================
;*  SBS INTEGRATED AVIONICS LIBRARY  Version 6.1  (08 Mar. 1999)
;*=========================================================================
;*
;* NAME:  sbs_dev.cfg
;*
;* DESCRIPTION:
;*    This file contains the information required to initialize one or
;*    more SBS device(s).
;*
;* FUNCTIONS:   None
;*
;*=========================================================================

[DEVICE=1]

; Base Memory Address Segment in hex (0-FFFFFFFFh)
; base_address=D0000h

; PCMCIA DSP Bootloader program filename (default bootload.txt)
; bootloader=bootload.txt

; Device Type (M1553_cPCI_3U,     A429_PC8,
;              M1553_cPCI_6U_1,   A429_PC16_1,
;              M1553_cPCI_6U_2,   A429_PC16_2,
;              M1553_FW5000,      A429_PC104,
;              M1553_PC104,       A429_PCMCIA,
;              M1553_PC3_1,       A429_V2_1,
;              M1553_PC3_2,       A429_V2_2,
;              M1553_PCI_1,       A429_cPCI_3U,
;              M1553_PCI_2,       A429_cPCI_6U_1,
;              M1553_PCMCIA,      A429_cPCI_6U_2,
;              M1553_PMC,         A429_PCI_8,
;              M1553_V6_1,        A429_PCI_16_1,
;              M1553_V6_2,        A429_PCI_16_2,
;              M1553_V6_3,
;              M1553_V6_4)
device_type=A429_PCI_8

; PCMCIA DSP BIT/Application program filename
; dsp_file=PCMDSP.TXT

; UNIX/NT Device Driver Name
filename=sbspci320

; Firmware filename
firmware=f050b.dat

; VME Interrupt Vector in hex (0-FFh)
; int_vector=FFh

; PC Base I/O Address in hex (300h-3FFh)
; io_base=380h

; Interrupt Request Level in hex (0-0Fh)
; irq_level=0Ah

; Number of Receive channels
num_receive=4
```

```
; Number of Transmit channels
num_transmit=4

; Product Type (ABI,ASF)
; product=ABI

; Relative position of PCI card under Windows 95
; relative_position=0

; PCMCIA Socket number (0-7)
; socket=0

; PC Memory Window Size (16k, 64k)
; window_size=16k

; PCMCIA Xilinx downloadable filename (.bit extension)
; xilinx_file=00670002.BIT
```

6.3.4 *dev_cfg.h* File



Note: If your operating system has a file system, you should use the *sbs_dev.cfg* file instead of *dev_cfg.h*. See page 25 for information on *sbs_dev.cfg*.

This file contains the device information to be used in lieu of *sbs_dev.cfg* for vxWorks and other operating systems that do not have a file system. Modify *dev_cfg.h* as follows:

1. Change directories to the *\sbs_ver6.10\working* directory if you are not already there.
2. Initialize the **dev_cfg** string array with the configuration parameters for each card. The keywords in *dev_cfg.h* are identical to those in *sbs_dev.cfg*; see Table 6.3.1 on page 24 for the keyword descriptions.
3. Save the file.
4. Refer to Chapter 16 of the *ARINC User's Manual* for information on compiling your application.

An example of the *dev_cfg.h* file appears on the following pages for card running under vxWorks.

```
/*=============================================================================
 * Copyright (c) 1999 by SBS Technologies, Inc.
 *                       2400 Louisiana Blvd., NE
 *                       AFC Building 5, Suite 600
 *                       Albuquerque, New Mexico  87110
 *                       Technical support (toll free)
 *                       877-TECHSBS (877-832-4727)
 *============================================================================
 *  SBS INTEGRATED AVIONICS LIBRARY  Version 6.1  (08 Mar. 1999)
 *============================================================================
 *
 * NAME:  dev_cfg.h
 *
 * DESCRIPTION:
 *    This file contains the device information to used be in lieu of
 *    sbs_dev.cfg. This file supports vxWorks and other operating
 *    systems that don't have a file system. To utilize this option,
 *    add #define NO_FILE_SYSTEM to the sbs_sys.h file. This compiler
 *    directive also causes the firmware to be loaded from the data
 *    array(s) firmware_n[FIRMWARE_LENGTH_n] found in firmware.h. The
 *    firmware.h file can be generated using the setupfw.c utility. You
 *    must put the appropriate value for firmware_n in the "firmware="
 *    field in the dev_cfg_array which follows.
 *
 * FUNCTIONS:   None
 *
 *============================================================================*/

char *dev_cfg_array[] = {

  "[DEVICE=1]",
```

```
    "; Base Memory Address Segment in hex (0-FFFFFFFFh)",
    "; base_address=D0000h",

    "; PCMCIA DSP Bootloader program filename (default bootload.txt)",
    "; bootloader=bootload.txt",

    "; Device Type",
    "device_type=A429_PCI_8",

    "; PCMCIA DSP BIT/Application program filename",
    "; dsp_file=PCMDSP.TXT",

    "; UNIX/NT Device Driver Name",
    "; filename=sbspci320",

    "; Firmware filename",
    "firmware=firmware_1",

    "; VME Interrupt Vector in hex (0-FFh)",
    "; int_vector=FFh",

    "; PC Base I/O Address in hex (300h-3FFh)",
    "; io_base=380h",

    "; Interrupt Request Level in hex (0-0Fh)",
    "; irq_level=0Ah",

    "; Number of Receive channels",
    "num_receive=4",

    "; Number of Transmit channels",
    "num_transmit=4",

    "; Product Type (ABI,ASF)",
    "; product=ABI",

    "; Relative position of PCI card under Windows 95",
    "; relative_position=0",

    "; PCMCIA Socket number (0-7)",
    "; socket=0",

    "; PC Memory Window Size (16k, 64k)",
    "; window_size=16k",

    "; PCMCIA Xilinx downloadable filename (.bit extension)",
    "; xilinx_file=00670002.BIT"

    };
```

6.4 Starting Unit Test



Tips:
Press the carriage return key (**Return**) to complete menu selections and enter responses to prompts.

Press **Return** to restore a Unit Test menu following a failure.

1. Execute the Unit Test application.

   As soon as you execute Unit Test, it parses the configuration file (*sbs_dev.cfg* or *dev_cfg.h*). If your configuration file parses without error, the menu shown in Figure 6.4.1 appears.

```
        SBS Technologies, Inc.
   Integrated Avionics Library Unit Test
      Version 6.1  Build 03.08.99

      a - Arinc 429 Unit Test
      m - MIL 1553 Unit Test
      q - Quit Unit Test

      Selection ? >
```

Figure 6.4.1: Integrated Avionics Library Unit Test Menu

If a parser error occurs, an error screen may appear instead of the Integrated Avionics Library Unit Test Menu. Figure 6.4.2 shows a sample error screen. Table 6.4.1 describes the common parser error messages that you may encounter.

```
        SBS Technologies, Inc.
   Integrated Avionics Library Unit Test
      Version 6.1  Build 03.08.99

    Failure parsing sbs_dev.cfg file!
    Device #1 missing "firmware=" keyword.

    'Q' to quit, return to reparse

    Selection ? >
```

Figure 6.4.2: Sample Parser Error Screen

Table 6.4.1: Parser Error Messages

Error Message	Diagnosis
Failure parsing configuration file! **Error opening sbs_dev.cfg file!**	Your application was unable to open the configuation file. Verify that this file is in your working directory.
Failure parsing configuration file! **<keyword>=<num> for device #xx is invalid.**	The parser found a value out of limits in the configuration file. Correct the line containing **<keyword>**.
Failure parsing configuration file! **Application does not support <name> devices!**	The application type **<name>_APP** (where **<name>** is the avionics bus type) is not defined in the *sbs_sys.h* file.
Failure parsing configuration file! **Device #xx missing "<keyword>=" keyword.**	The specified **<keyword>**, which is required for the specified device (**Device #xx**, where **xx** is in the range 1 to **SBS_MAX_DEV**), was not found.
Failure parsing configuration file! **No device defined in configuration file!**	The **Device** keyword was not found in the configuration file.
Failure parsing configuration file! **"<name>" is not a valid device type.**	The **<name>** specified for the **device_type** keyword is not valid. Choose a valid device type from the list included in the configuration file.
Failure parsing configuration file! **xx recv + yy xmit != zz channels**	The sum of the values for the receive (**xx**) and transmit (**yy**) channels (**num_receive** + **num_transmit**) did not equal the total number of channels (**zz**) for an A429 device.
Failure parsing configuration file! **"<keyword>=<num>" on line xx: duplicate entry.**	A value for the **<keyword>** specified on line **xx** has already been parsed for this device number.
Failure parsing configuration file! **Line number xx has more than 132 characters.**	The number of characters on line **xx** exceeds the maximum number that can be read.
Failure parsing configuration file! **"Device=0" on line xx must be greater than 0.**	The **Device** keyword must have a value greater than zero.
Failure parsing configuration file! **"Device=<num>" on line xx exceeds SBS_MAX_DEV.**	The value **<num>** is greater than the **SBS_MAX_DEV** value specified in the *dev_mgmt.h* file.

2. If a parser error occurs, correct the error in the configuration file. If you are using *sbs_dev.cfg*, press **Return** from the Unit Test error screen to reparse. If you are using *dev_cfg.h*, exit and recompile your code.

3. After the configuration file parses without error (i.e., once you see the menu shown in Figure 6.4.1), enter **a** to access the ARINC 429 Interface Library Unit Test Menu.

   The menu shown in Figure 6.4.3 appears.

```
                SBS Technologies, Inc.
          ARINC 429 Interface Library Unit Test
            Version 6.1  Build 03.08.99

             1 - Device Management Tools
             2 - Interrupt Management Tools
             3 - Transmit Management Tools
             4 - Receive Management Tools
             5 - Bus Monitor Management Tools
             q - Quit Unit Test

          Selection? >
```

Figure 6.4.3: ARINC 429 Interface Library Unit Test Menu

6.5 Opening the A429 Device

1. To select **Device Management Tools**, enter **1**.

   The menu shown in Figure 6.5.1 appears.

```
                          Device Management Tools

     1 - Initialize Device (Steps 2-7)    i - Get Device Information
     2 - Open Device                      b - Execute Built-In Tests
     3 - Load Firmware                    e - Enable Hardware Interrupts
     4 - Start Application                d - Disable Hardware Interrupts
     5 - Initialize Device Memory         p - Reprogram Flash Memory
     6 - Initialize Interrupt Queue       s - Execute Soft Reset
     7 - Create Seq. Mon. Buffers         m - Set 1553 ASF Mode
     8 - Process Interrupt Queue          g - Get A429 Channel Info
     9 - Start Device I/O
     0 - Stop Device I/O                  r - Read/Write RAM
     c - Close Device                     q - Return to Main Menu
--------------------------------------------------------------------------------
Selection:

Messages:
--------------------------------------------------------------------------------
Menu Selection ? >
```

Figure 6.5.1: Device Management Menu



Note: The Device Management menu options are described in detail in Chapter 17 of the *ARINC User's Manual*.

2. To select **Open Device**, enter **2**.

   The prompt shown in Figure 6.5.2 appears.

```
--------------------------------------------------------------------------------
Selection: Open Device
Device Number? >


Messages:
--------------------------------------------------------------------------------
```

Figure 6.5.2: Open Device Prompt

3. To select and open device 1, enter **1**.

   The area between the dashed lines clears, and the message "Device **#1** successfully opened!" appears.

4. To continue, press **Return**.

   The area between the dashed lines clears.

LEDs If you have *not* loaded firmware on this card before:

- ➢ LED 1 should be off.
- ➢ LED 2 should be off.

If you *have* loaded firmware on this card before:

- ➢ LED 1 should be on.
- ➢ LED 2 may be on if the card was previously running.



Cross Reference: For the locations and descriptions of the LEDs, see Figure 4.1.1 on page 12 and Figure 4.1.2 on page 13.

6.6 Loading the Firmware

1. To select **Load Firmware**, enter **3**.

   The prompt shown in Figure 6.6.1 appears between the two dashed lines.

```
--------------------------------------------------------------------------
 Selection: Load ARINC Firmware
 Device Number? >



 Messages:
 --------------------------------------------------------------------------
```

Figure 6.6.1: Load ARINC Firmware Prompt

2. Enter **1** for the device number.
3. If you do not receive a firmware source prompt, the firmware loads from the firmware file, and the message and prompt shown in Figure 6.6.2 appear.

```
--------------------------------------------------------------------------
 Selection: Load ARINC Firmware
 Device Number? >1


           Load Firmware Successful
 Messages:  Hit CR to continue
 --------------------------------------------------------------------------
```

Figure 6.6.2: Load ARINC Firmware Verification Screen (without Firmware Source Prompt)

4. If you receive a firmware source prompt, enter the appropriate number to load from flash, from file, or via driver (depending on the type of card and operating system you are using, you may see only two of these options at the prompt).

If the firmware loads successfully, a message and prompt similar to those in Figure 6.6.3 appear.

```
------------------------------------------------------------------------
 Selection: Load ARINC Firmware
 Device Number? >1

'0' = FILE; '1' = FLASH
Firmware Source? >

            Load Firmware Successful
 Messages:  Hit CR to continue
 ------------------------------------------------------------------------
```

Figure 6.6.3: Load ARINC Firmware Verification Screen (with Firmware Source Prompt)

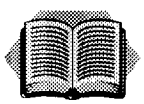
5. To continue, press **Return**.

LEDs If you have *not* loaded firmware on this card before:

- LED 1 should be off.
- LED 2 should be off.

If you *have* loaded firmware on this card before:

- LED 1 should be on.
- LED 2 may be on if the card was previously running.



Cross Reference: For the locations and descriptions of the LEDs, see Figure 4.1.1 on page 12 and Figure 4.1.2 on page 13.

6.7 Running Built-In Tests (BITs)

Note: The device must be open and you must not be in application mode.

1. To select **Execute Built-In Test**, enter **b**.

   The prompt shown in Figure 6.7.1 appears between the dashed lines.

```
--------------------------------------------------------------------------
 Selection: Execute Built in Tests
 Device Number? >


 Messages:
 --------------------------------------------------------------------------
```

Figure 6.7.1: Execute Built-in Tests Prompt

2. Enter **1** for the device number.

   If the tests are successful, the messages and prompt shown in Figure 6.7.2 appear between the dashed lines.

```
--------------------------------------------------------------------------------
Selection: Execute Built in Tests


Please Wait (Takes 6 to 9 seconds)...
          BIT passed!
          Device closed. Initialize before running.
Messages:  Hit CR to continue
--------------------------------------------------------------------------------
```

Figure 6.7.2: Execute Built-in Tests Passed Screen

   If the tests are unsuccessful, error messages appear between the dashed lines.

3. To continue, press **Return**.

   The area between the dashed lines clears.

LEDs

- LED 1 should remain on.
- LED 2 should flash once, then remain off.

Cross Reference: For the locations and descriptions of the LEDs, see Figure 4.1.1 on page 12 and Figure 4.1.2 on page 13.

6.8 Initializing the A429 Device

1. To select **Initialize Device**, enter **1**.

   The messages and prompt shown in Figure 6.8.1 appear between the dashed lines.

```
--------------------------------------------------------------------------
 Selection: Initialize device
 Device Number? >



 Messages:
 --------------------------------------------------------------------------
```

Figure 6.8.1: Initialize ARINC Device Prompts

2. Enter **1** for the device number.

3. At the Queue Length prompt, press **Return** to select the displayed default value, or enter a different value.

4. If you do not receive a firmware source prompt, proceed to step **5**. If you do receive the firmware source prompt, enter the appropriate number to load from flash, from file, or via driver (depending on the type of card and operating system you are using, you may see only two of these options at the prompt).

5. At the following prompts, press **Return** to select the displayed default value, or enter a different value:

   - # of Global SM Buffers
   - Global Buffer Size
   - Loop
   - # of Channel SM Buffers
   - Channel Buffer Size
   - Loop

   If the process is successful, the messages and prompt shown in Figure 6.8.2 appear between the dashed lines.

```
--------------------------------------------------------------------------
 Selection: Initialize device
 Device Number? >1                      Global Buffer Size [4096]? >
 Queue Length? [4] >                    Loop [Y]? >
 '0' = [FLASH]; '1' = FILE              # of Channel SM Buffers [2]? >
 Firmware Source ? >1                   Channel Buffer Size [256]? >
 # of Global SM Buffers [2]? >          Loop [Y]? >
           Initialize device successful!
 Messages:  Hit CR to continue
 --------------------------------------------------------------------------
```

Figure 6.8.2: Initialize ARINC Device Passed Screen



Note: The Firmware Source prompt on your screen may be absent or show different options than the screen shown in Figure 6.8.2.

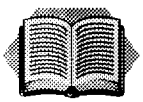
If the process is unsuccessful, error messages appear between the dashed lines.

6. To continue, press **Return**.

LEDs

➢ LED 1 should remain on.

➢ LED 2 should remain off.



Cross Reference: For the locations and descriptions of the LEDs, see Figure 4.1.1 on page 12 and Figure 4.1.2 on page 13.

6.9 Exiting Unit Test

1. To return to the main menu, enter **q**.

   The screen clears, and the ARINC 429 Interface Library Unit Test menu appears.

2. To exit from the Unit Test application, enter **q**.

   The prompt shown in Figure 6.9.1 appears.

```
Are you sure you want to quit? ([y]/n)
```

Figure 6.9.1: Prompt to Quit

3. Enter **y**.

7: Connecting to A429 Bus

If the Unit Test procedures (conducted in the previous chapter) ran without error, you may proceed with ARINC 429 operations. However, you must first connect each channel of the card to an actual A429 bus(es) via the connector (see Figure 7.0.1), as appropriate for your system. Pinouts for the PCI-8 connector are listed in Table 4.1.1 on page 14.

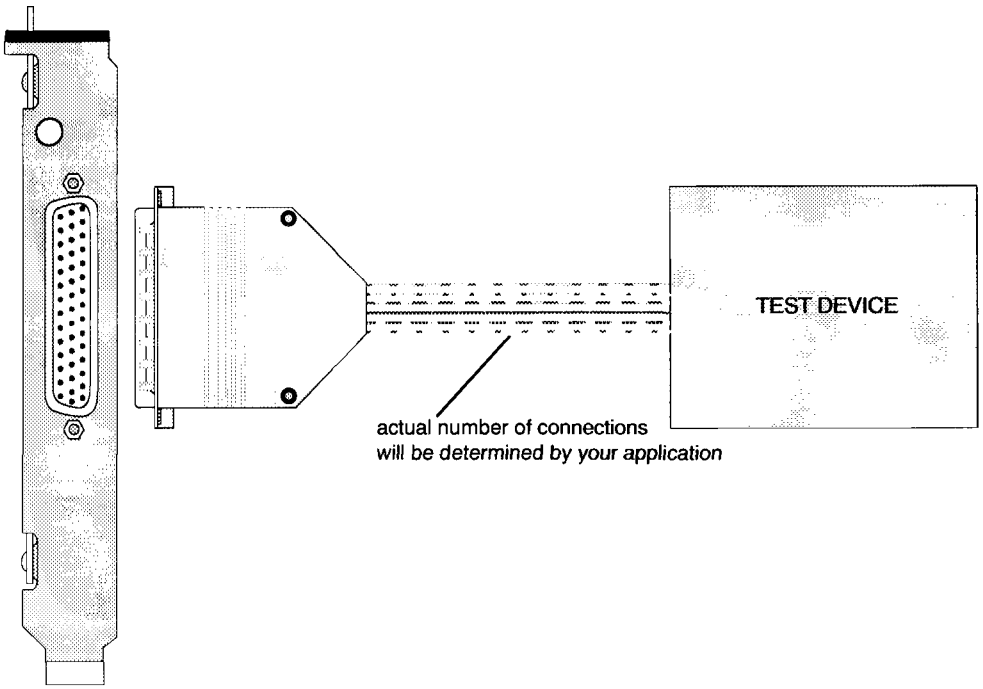


Figure 7.0.1: Connecting the A429 PCI-8 to an A429 Bus

8: What to Do Next

If your card is properly connected to an A429 bus, you may proceed in one or more of the following ways:

➢ Explore the Unit Test menus for A429 operations.



Cross Reference: You can use the Unit Test application to set up and execute basic A429 operations and to troubleshoot problems. See Appendix E of the *ARINC User's Manual* for more details.

➢ Run sample A429 applications.



Cross Reference: If you are a first-time user, SBS recommends that you execute the sample A429 applications that are included with the A429 PCI-8 product shipment. These applications are detailed in the *ARINC User's Manual.*

➢ Build a custom A429 application.



Cross Reference: If you are an advanced user, you may wish to begin designing your own A429 application. See the *ARINC User's Manual* for complete details.

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP