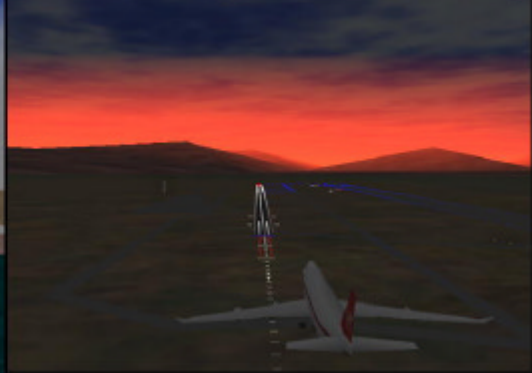




Primary Image Ltd.



Installation Manual



13 January 1998

© 1997-1998 Primary Image Limited. All Rights Reserved

The information contained herein is the property of Primary Image Limited and is not to be disclosed or used either in whole or in part without prior written permission of Primary Image Limited.

Primary Image Ltd.,
Millbank House,
171-185 Ewell Road,
Surbiton,
Surrey, KT6 6AP

Tel: +44 (0)181 339 9669
Fax: +44 (0)181 339 9091

Document Ref: 3344man002

PIRANHA™ and **TEMPEST™** are registered trademarks of Primary Image Ltd.

1	INTRODUCTION	2
2	CONFIGURATION OPTIONS.....	3
2.1	LOCAL TO EACH BOARD	4
2.2	MULTI-BOARD CONFIGURATIONS	5
2.3	IMPLEMENTING THE DIFFERENT CONFIGURATIONS	5
3	INSTALLATION.....	8
3.1	HARDWARE INSTALLATION	8
3.2	DRIVER INSTALLATION.....	10
3.2.1	<i>Windows 95 Driver Installation.....</i>	<i>10</i>
3.2.2	<i>Windows NT Driver Installation.....</i>	<i>11</i>
4	TROUBLE-SHOOTING.....	13
4.1	INSTALLATION PROBLEMS.....	13
4.1.1	<i>Windows 95 did not prompt me for new drivers when I first installed my Piranha card.....</i>	<i>13</i>
4.1.2	<i>I have installed the drivers, but I none of the Piranha demos will work.....</i>	<i>14</i>
4.1.3	<i>I have run DETECT and it has reported that the Piranha cards are not correctly set up.</i>	<i>14</i>
4.1.4	<i>I have run GALCONF but DETECT still does not report all of the Piranha cards that I have in my PC.15</i>	<i>15</i>
4.1.5	<i>I have run GALCONF, but now my Windows Desktop does not get refreshed correctly.</i>	<i>15</i>
4.1.6	<i>DETECT tells me that 'BIOS has configured Pixel Unit at illegal address'</i>	<i>15</i>
4.1.7	<i>When I run a Piranha application, I get a dialogue box telling me that a driver is not installed. 16</i>	<i>16</i>
4.1.8	<i>If all else fails.....</i>	<i>16</i>

1 Introduction

Congratulations on your decision to use Primary Image's **PIRANHA** hardware for your graphics system. We hope your projects using this hardware run smoothly. To this end, we recommend that you read this manual fully.

We will be glad to answer any technical queries you may have, preferably via e-mail at support@primary-image.com, or by phone on +44 (0)181 339 9669 (ask for support).

A brief outline of the contents of this manual follows:

Section 2 describes the various configuration options that are available with the **PIRANHA** hardware.

Section 3 deals with how to install the **PIRANHA** hardware and software on your PC.

Section 4 is a trouble-shooting section in the event that you are unable to get your **PIRANHA** hardware working.

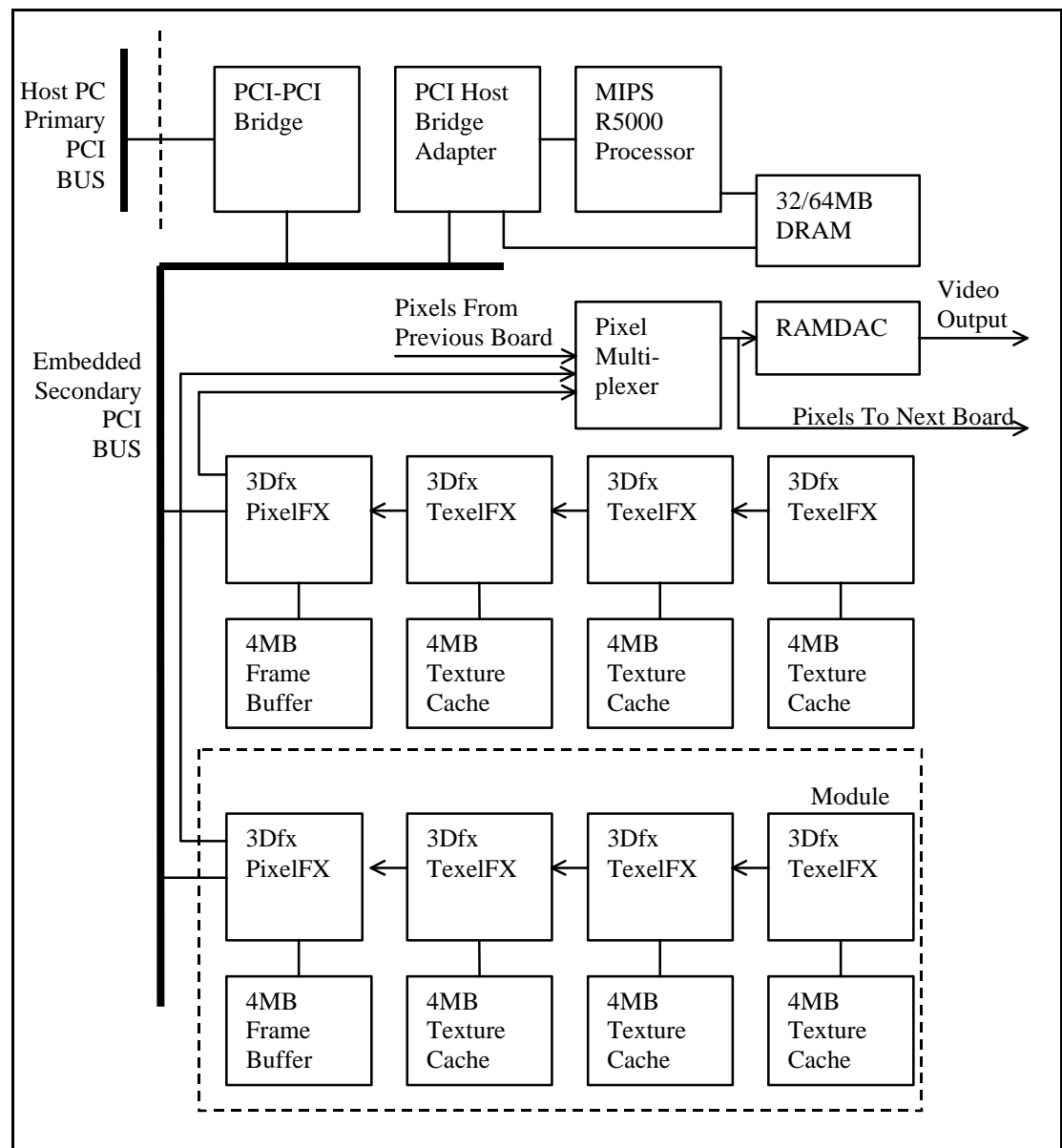
2 Configuration Options

The **PIRANHA** hardware offers many configuration options to satisfy different price/performance/functionality requirements. This section explains what these different options are and how they affect the system.

The options can be split into two broad categories:

- Options local to each **PIRANHA** card.
- Options available by connecting multiple **PIRANHA** cards together.

The block diagram below shows the main components of the **PIRANHA** hardware. Referencing this diagram may clarify some of the options.



2.1 Local to each Board

The **PIRANHA** card has two main functions to perform:

- Performing the geometry calculations: This includes database management, tree traversal, vertex transformation, lighting, and polygon setup.
- Performing the rendering: This is the process of actually rasterising the polygons into the frame buffer.

Each **PIRANHA** has an R5000 processor for performing the geometry calculations, and either one or two Pixel Chain units for performing the rendering. Each Pixel Chain consists of one 3dfx Voodoo PixelFX and from 1 to 3 3dfx Voodoo TexelFX's. Each Pixel Chain is capable of rendering at approximately 45Mpixels/second.

Options available affecting the geometry calculations are:

- 200MHz or 250MHz (available soon) processor.
- 32MBytes or 64MBytes of DRAM. About 4M Bytes of the DRAM is used by the **TEMPEST** code itself. The rest is available for 3D model databases, textures, etc.. So for very large databases you may require the 64MByte option.

Options that affect the rendering are:

- One or two Pixel Chain Units. Two Pixel Chain Units can be configured either in SLI mode or non-SLI mode by software. In SLI mode, both Pixel Chain Units work on the same polygons in parallel, each one doing alternate lines, this mode doubles the pixel fill rate to approximately 90Mpixels/second. In non-SLI mode each pixel chain unit works independently on different polygons, custom hardware merges the outputs of the two units together to achieve several effects. With two pixel chain units you can achieve **one** of the following:

double the pixel fill rate (SLI mode),

or generate 2 sub-pixel anti-aliasing (non-SLI mode),

or run in stereo glasses mode with a single board (non-SLI mode),

or run video modes more than 960 pixels wide (non-SLI mode).

In the non-SLI modes the achievable Geometry rate may drop by up to 30%.

- Each pixel chain may have either 1, 2 or 3 Texture Units. The following effects are available with these configurations:
 - 1 Texture Unit - Dithered tri-linear interpolation.
 - 2 Texture Units - True tri-linear interpolation or dithered tri-linear interpolation with detail texture.

3 Texture Units - True tri-linear interpolation with detail texture.

In addition, the maximum size of the texture cache is 4MBytes, 8MBytes and 12MBytes for 1, 2 and 3 texture units respectively. The size of the texture cache determines the total size of textures that can be used in any one view without needing to resort to paging from main DRAM memory. For optimum performance, all the textures for all databases in use should fit in the texture cache.

- Each Texture Unit may have either 2 or 4 MBytes of texture cache.

2.2 Multi-Board Configurations

Multiple **PIRANHA** cards may be connected together in a single graphics system.

A graphics system will consist of one or more channels, where a channel is defined as the hardware required to generate a single video output. Each channel may consist of up to 8 **PIRANHA** cards.

All the **PIRANHA** cards in a single channel **must** be connected using the intra-channel cables. If the channels are required to be frame locked, then the master boards in each channel should be connected via inter-channel cables. This is explained in more detail in section 3.

Multiple boards in a channel can be used to achieve any one of the following effects:

- 2, 4 or 8 boards each configured in SLI mode may be used to obtain 2, 4 or 8 sub-pixel anti-aliasing respectively. Geometry and rendering rates are the same as a single board in SLI mode.
- 1, 2 or 4 boards each configured in non-SLI mode may be used to obtain 2, 4 or 8 sub-pixel anti-aliasing respectively. Geometry and rendering rates are the same as a single board in non-SLI mode.
- Two SLI configured boards may be used to generate video modes wider than 960 pixels. This has the advantage over using a single board in non-SLI mode in that the Geometry performance is not compromised and the achievable rendering rate is doubled.
- Two SLI configured boards may be used to generate stereo glasses video output, again with double the geometry and rendering performance of using a single board in non-SLI mode.

2.3 Implementing the Different Configurations

A second Pixel Chain Unit may be added to a board as a plug in module.

All other local board options are manufacturing options, though some options may be added subsequently by returning the board for upgrade. Contact your sales representative for information on this.

2, 4 or 8 boards per channel and multiple channels should be cabled together as described in section 3.

The mode in which each channel operates is a function of the video mode, the number of boards connected in the channel, and the environment variable CGIMODEn where n is the channel number. n starts from 0 for the first channel in the system.

CGIMODEn should be set to one of the following:

ANTIALIAS	Causes modules to be configured in non-SLI so that each card generates 2 sub-pixels for anti-aliasing. Thus a single board channel will generate 2 sub-pixel anti-aliasing, a dual board channel will generate 4 sub-pixel anti-aliasing, and a quad board channel will generate 8 sub-pixel anti-aliasing. The animation rate may drop by up to 25% in this mode.
STEREO	A single board channel in non-SLI mode will be configured in stereo glasses mode. A dual board channel in SLI mode will be configured in stereo glasses mode.
TTL	This option may be used when video frame locking the entire graphics system to some other video source. Contact your sales representative if planning to do this. This option should be set if the external video source is generating TTL level signals, otherwise ECL level signals are expected.

These options may be used in conjunction, eg.

set CGIMODE0=ANTIALIAS STEREO

Would cause channel 0 (which must consist of two cards configured in non-SLI mode in this case) to generate 2 sub-pixel anti-aliased stereo glasses mode video.

The following table shows all the possible configurations. Geometry and rendering speeds are specified relative to a single card with no module.

Hi-Resolution in the Special column refers to the use of video modes above 960 pixels wide or with a dot clock of greater than 70 MHz.

SPAA in the Special column refers to the number of sub-pixels of anti-aliasing provided by the configuration.

Table 2-1: **PIRANHA** Configuration Options

Number of Cards	Maximum Resolution	Max Dot Clock	Special	Rendering Speed	Geometry Speed	Module	CGIMODEn
1	960x720	66MHz	None.	1	1	None	N/A
1	960x702	66MHz	None.	2	1	SLI	N/A
1	960x720	56MHz	2 SPAA.	1	$\frac{3}{4}$	Non-SLI	ANTIALIAS
1	960x720	66MHz	Stereo glasses.	2	$\frac{3}{4}$	Non-SLI	STEREO
1	1152x900	84MHz	Hi-Resolution.	2	$\frac{3}{4}$	Non-SLI	N/A
2	960x720	56MHz	2 SPAA.	1	1	None	N/A
2	960x702	56MHz	2 SPAA.	2	1	SLI	N/A
2	960x720	56MHz	4 SPAA.	1	$\frac{3}{4}$	Non-SLI	ANTIALIAS
2	960x720	66MHz	Stereo glasses.	2	1	None	STEREO
2	960x702	66MHz	Stereo glasses.	4	1	SLI	STEREO
2	960x720	56MHz	2 SPAA, Stereo glasses.	2	$\frac{3}{4}$	Non-SLI	STEREO ANTIALIAS
2	1280x1020	110MHz	Hi-Resolution.	2	1	None	N/A
2	1280x1020	110MHz	Hi-Resolution.	4	1	SLI	N/A
2	960x720	56MHz	4 SPAA.	1	$\frac{3}{4}$	Non-SLI	ANTIALIAS
4	960x720	56MHz	4 SPAA.	1	1	None	N/A
4	960x702	56MHz	4 SPAA.	2	1	SLI	N/A
4	960x720	56MHz	8 SPAA.	1	$\frac{3}{4}$	Non-SLI	N/A
8	960x720	56MHz	8 SPAA.	1	1	None	N/A
8	960x702	56MHz	8 SPAA.	2	1	SLI	N/A

3 Installation

3.1 Hardware Installation

CAUTION: The *PIRANHA* hardware is sensitive to damage by electrostatic discharge. Precautions should be taken whenever handling the boards. Preferably an earthed wrist-strap should be worn during the installation procedure. If this is not possible, you should touch the chassis of the PC (which should be plugged in but not turned on) prior to removing the boards from their protective bag, then avoid moving your feet until after the board is inserted in the PC.

IMPORTANT: If your *PIRANHA* card has a module fitted, it is necessary to provide additional power to the board. You will need to connect the power from a spare disk drive power connector in your PC to the connector *P1* on each module in the system. This can be achieved using the power connectors provided. Up to two modules may be powered from a single connector from your PC's power supply by daisy chaining the supplied power connectors. If more than two modules are powered from a single connector then you may experience problems with the Piranha cards crashing due to voltage drops in the power connector.

IMPORTANT: *PIRANHA* cards require a certain degree of ventilation to prevent them from overheating. This means you should have some form of forced air cooling blowing over the *PIRANHA* cards.

When the 'right-most' and 'left-most' boards are referred to, this is for when the PC is viewed from the front. If your PCI back-plane is vertical, read 'upper-most' for 'right-most' and 'lower-most' for 'left-most'.

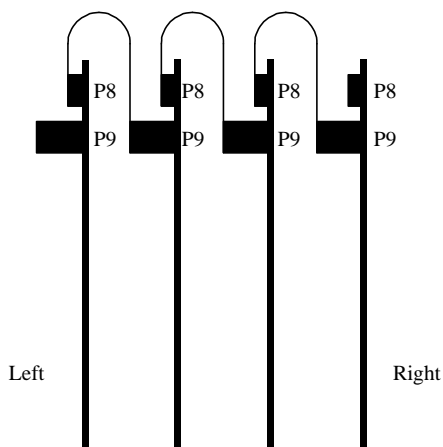
The following instructions refer to two types of cable. These are:

- Intra-channel cable. These are used to connect together all the boards that comprise a channel. They are recognisable as a flexi-PCB cable. You will notice that the connector on the one end protrudes slightly more than the connector at the other end.
- Inter-channel cable. These are used to connect together the clock-master card (ie. The left most card) of each channel in a multi-channel system. They are recognisable as a twisted pair cable. You will notice that the connector at one end is 14-way, whilst that at the other is 15-way.

The following procedure should be used to insert the boards in your system.

Step 1: With the power off, remove the computer cover. For each *PIRANHA* card you require in your system, you will need to locate a free PCI slot that is capable of taking a full length PCI card. All boards within a single channel must be located in adjacent PCI slots. Remove the blanking plate from each of the slots to be used.

- Step 2: If any of the channels in your configuration consist of multiple cards, then you must perform the following operation before continuing. For each board apart from the left most board of each channel, you should insert the long connector from an intra-channel connector into the connector **P9** on the **PIRANHA** card. This should be done in such a way that the connector naturally protrudes up above the card. Leave the other end free for now.
- Step 3: Insert all of the **PIRANHA** cards firmly into the PCI slots, ensuring that each one is fully located in the slot.
- Step 4: The free end of each of the intra-channel connectors should now be inserted into connector **P8** of the card to its left. As an example, the figure below shows a cross sectional view of what a 4-board channel would look like.
- Step 5: Connect your monitor(s) to **P7**, the SVGA connector, of the right-most **PIRANHA** card of each channel.

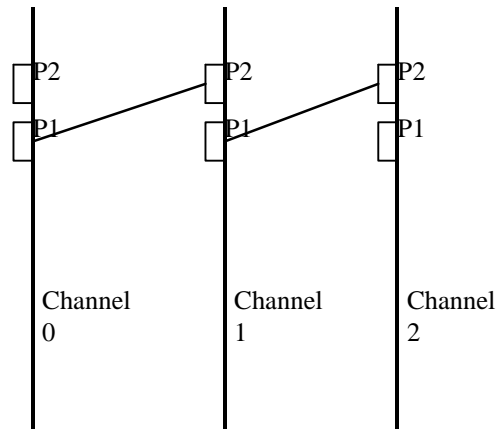


Vertical Cross-section Of a Single Channel

The following step is only required for multi-channel systems.

- Step 6: Plug the 15-way connector of one of the inter-channel connectors into **P1** on the left-most card of one of the channels. Connect the other end into **P2** of the left-most card of another channel. Repeat this until all the channels are connected together in a series.

The order in which the channels are connected together using the inter-channel connectors determines which channel is channel 0, which is channel 1, etc. Channel 0 will be the channel that contains the card that has connector P2 empty. Channel 1 will be the channel that contains the card that is connected to channel 0. This continues such that the last channel is the channel that contains the card that P1 empty. This is illustrated below.



Top View of 3 channels of 1 card each

If you ever need to remove the inter-channel connectors, you should be careful not to extract them by pulling on the wires. Doing this may break the cable.

3.2 Driver Installation

You should install the **PIRANHA** hardware as described in section 3.1 before installing the drivers.

Each **PIRANHA** card contains a PCI to PCI bus bridge. The specification for these was only introduced in PCI rev 2.0, and enhanced in PCI rev 2.1. The BIOS on your PC must be PCI 2.1 compliant in order for the **PIRANHA** devices to be detected and correctly configured. Most BIOS's dated 1996 or later will be PCI 2.1 compliant. If you have an older BIOS then you may experience problems during installation. Also, some BIOS's contain bugs in the code that sets up PCI to PCI bridges. Primary Image has produced a fix for these problems that should work in most cases. Refer to section 4 on trouble-shooting if you have this problem.

Driver installation is performed differently dependant upon whether you are using Windows 95 or Windows NT. These are covered in the next two sub-sections.

Generally you would perform the installation directly from the '**TEMPEST** Development Kit' CD, the instructions below assume this to be the case. If the target PC does not have a CD Drive then installation from floppy disk is also possible. On a PC with a CD Drive, copy the directories 'Win95' and 'WinNT' (ensuring that you preserve the directory structure) onto a blank floppy disk. Use this disk instead of the CD when performing the installation on the target PC.

3.2.1 Windows 95 Driver Installation

The following instructions apply to Windows 95, OSR2 (OEM Systems Release 2 of Windows 95, some PC's are shipped with this version pre-installed), Memphis and Windows 98.

Windows 95 should detect the presence of the **PIRANHA** the first time you turn on the machine after fitting the boards. If you have any problems during the installation, refer to section 4 on trouble-shooting. The following steps should be performed.

Step 1: Windows will display a message box saying that it has detected new hardware. It will say that it is a PCI to PCI bridge and it will automatically install the drivers for it. You should not need to interact in this.

Step 2: Windows will display another message box saying that it has detected new hardware. It will say that it is a PCI Multi Media Video Device. It will prompt you with a message saying that you must install a driver for it. Click the 'Have Disk...' button. A box will appear asking where the drivers are located. Insert the '**TEMPEST** Development Kit' CD into your CD Drive. Enter the directory

[DRIVE LETTER]:Win95

Where *[DRIVE LETTER]* is the letter of your CD Drive.

Press return. Windows will then list the drivers on the CD that match the device it has found. There should be only one of these, so select it.

Windows will copy the relevant drivers and update its registry. You will then be told that you must shut down and restart in order to use the driver. Click cancel on this so that the PC will not restart.

Step 3: Windows will display yet another message box saying that it has detected new hardware. Again, it will say that it is a PCI Multi Media Video Device. Follow the instructions in step 2 again. This time you will notice that the driver that it displays as matching the device is different than the first time. This is because the **PIRANHA** card contains multiple PCI devices.

Step 4: If the card has a module, Windows will install another driver for the module, you will not need to interact with this since Windows knows what to do this time.

Step 5: If you have installed more than one **PIRANHA** card, Windows will install all the other cards automatically.

Step 6: Shut down and restart.

The set-up procedure installs the file *PIMEMMAP.vxd* into the directory *%WINROOT%\system*. The registry will also be updated to cause the driver to be started when the system boots.

3.2.2 Windows NT Driver Installation

The following instructions apply to both Windows NT version 3.51 and version 4.0.

Insert the '**TEMPEST** Development Kit' CD into the PC's CD Drive.

Either from a DOS prompt, the File Manager, or the Explorer, run the following:

```
[DRIVE LETTER]:WinNT\setup.exe
```

Where *[DRIVE LETTER]* is the letter of your CD Drive.

This will copy the drivers *PIMAPMEM.sys* and *PIGENPRT.sys* into the directory *%WINROOT%\system32\drivers* and update the registry to cause these drivers to be started at boot up.

You should then shut down and restart your PC.

4 Trouble-Shooting

4.1 Installation Problems

The Piranha card contains an embedded PCI to PCI bridge which is fully compliant with the PCI specification version 2.1. Unfortunately, not all PC BIOS's seem to be able to set these up correctly. This is either due to the BIOS being pre PCI V2.1, or the BIOS containing bugs.

This can lead to the card not being installed correctly by the BIOS's Plug 'n Play boot sequence.

This can manifest itself in one of the following ways:

- Windows 95 does not report that it has found new hardware when you first plug the Piranha cards into your PC.
- Windows appears to have installed the cards correctly, but when you try to run an application using the Piranha cards it does not run up correctly.

The following Q&A sections will try to help you overcome any installation problems.

4.1.1 Windows 95 did not prompt me for new drivers when I first installed my Piranha card.

If Windows 95 did not detect the Piranha card at all, and thus did not prompt you to install the driver, then you must install the driver manually. To do this:

Click on **Start->Settings->Control Panel**.

Double click on **Add New Hardware**.

Click on **Next**.

Click on **No** when prompted if you want Windows search for new hardware.

Click on **Next**.

Double click on **Sound, video and Game Controllers**.

Click on **Have Disk**

Insert the '**TEMPEST** Development Kit' CD into your CD Drive. Enter the directory

[DRIVE LETTER]:\Win95

Where *[DRIVE LETTER]* is the letter of your CD Drive.

Click on **OK**.

Double click on **Piranha Processor PCI Bridge**.

Click on **Next**.

Click on **Finish**.

This will copy the Piranha drivers to the *windows\system* directory and update the registry so that the drivers are started on power up.

You should now restart the PC.

4.1.2 I have installed the drivers, but I none of the Piranha demos will work.

You should run the utility *DETECT.exe* from the directory *\tempest\utils*. This program will scan the PCI bus and check each Piranha card found for valid configuration. If any Piranha card is set up incorrectly by the BIOS then this will be reported. If no errors are reported, and the number of cards detected matches the number of Piranha cards in your PC, then the cards are set up correctly.

4.1.3 I have run *DETECT* and it has reported that the Piranha cards are not correctly set up.

If *DETECT* reports an invalid set up then you may be able to use the *GALCONF* utility to fix the problem.

GALCONF.exe is located in the *\tempest\utils* directory on the '**TEMPEST** Development Kit' CD. If you run it with no options then you will get a warning message to inform you that the utility may generate side effects, and thus all other applications should be run before running it. You can use the *nowarn* command line parameter to suppress this warning message.

GALCONF will scan the PCI bus to determine what PCI memory resources are currently being used by other devices, such as Graphics Cards, etc. It will then determine the largest contiguous range of memory that is currently unused by other devices. Finally, it configures all the Piranha's in the PC to use that memory range.

By default, *GALCONF* assumes that the BIOS has at least configured the bus numbers of all the busses in the system. If, when you run *GALCONF*, you get the message:

!!! The BIOS has not correctly configured all the busses in your system.

!!! You should re-run this utility with the 'bus' option

Then it means that your BIOS has made no attempt to configure any PCI to PCI bridge chips. In this case, you should use the command line option *bus* when invoking *GALCONF*.

After running *GALCONF* you should run *DETECT* again to check that all the Piranha cards are now configured correctly.

Once you have determined the correct settings for *GALCONF*, you will need to run it up each time you restart your computer. You could place a shortcut to *GALCONF* in your start up folder so that *GALCONF* gets invoked automatically each time you start up windows. If you wish to start up your application automatically when the PC boots

you should take special care to ensure that *GALCONF* is run before the application starts. The easiest way to do this is to put a shortcut to a batch file in the start up folder. The batch file should run *GALCONF* followed by your application. If you simply put both *GALCONF* and your application in your start up folder, then they could get executed in any order.

- 4.1.4 I have run *GALCONF* but *DETECT* still does not report all of the Piranha cards that I have in my PC.

See section 4.1.8.

- 4.1.5 I have run *GALCONF*, but now my Windows Desktop does not get refreshed correctly.

GALCONF attempts to determine the physical PCI memory usage of all currently installed PCI devices.

The memory usage of a PCI device is supposed to be hidden from user applications since it is normally only the device driver that needs to know. *GALCONF* uses a backdoor approach by examining the PCI configuration registers. This can confuse some device drivers. We have only found this problem with certain Graphics Cards.

If you get this problem, then it means that *GALCONF* cannot safely determine current PCI memory usage. In this case you will need to manually tell *GALCONF* the start address where it should begin allocating Piranha cards. This will be a trial and error task. However, once you have determined a valid address, you should be able to use the same address each time you restart your PC (though you may need to change it if you add any new hardware).

To specify the address to use you should use the command line option *addr <address>* where address is in hex.

eg. *galconf nowarn addr 20000000*

Would begin allocating Piranha cards at address 0x20000000. The address specified should not be less than 0x20000000. You should try addresses in increments of 0x10000000 until you find one that allows a Piranha application to run, and does not affect any other hardware on your PC.

- 4.1.6 *DETECT* tells me that 'BIOS has configured Pixel Unit at illegal address'

Due to the internal addressing architecture of the R5000 host bridge on the Piranha card, there are a few addresses that the Pixel Units may not be configured at. Since the BIOS does not know what these addresses are, there is a small possibility that a Pixel Unit will get configured at one of these addresses. If you get this error, then you will need to run the *GALCONF* utility. See section 4.1.3.

4.1.7 When I run a Piranha application, I get a dialogue box telling me that a driver is not installed.

You have not installed the Piranha device drivers correctly. For Windows 95 see section 4.1.1. For Windows NT, see section 3.2.2.

4.1.8 If all else fails.

If you have carefully followed the advice given in the previous sections, but you still cannot get any Piranha application to run, then you should contact Primary Image's Technical Support, preferably via e-mail (support@primary-image.com). Please include a full description of what you have tried. Also attach the output of *GALCONF* and the verbose output of *DETECT* (/v option). This can be done most easily by dumping the output to a file,

eg. *galconf nowarn [bus] [addr XXXXXXXXX] > galconf.out*

and *detect /v > detect.out*