PCI eXtensions for Intrumentation

# PXIMate

## A practical guide to using **PXI**


6th Edition

# PXImate

This book provides an overview of the PXI standard and its derivative versions. For those of you who have earlier editions of the PXImate, we have taken a different tack in terms of the chapter layouts and the goals of each section. It is written for the test engineer rather than a module design engineer. It provides an overview for those new to PXI systems and a useful source of reference material for the more experienced. This book will delve into hardware and software architectures, compare PXI-1 (original PXI) with PXI Express and look at how a PXI Hybrid chassis supports both form factors. For greater detail, we recommend that you go to the official website of the PXI Systems Alliance (www.pxisa.org).

This is a living document that Pickering Interfaces will continue to develop in support of the PXI standard and its future evolution. We welcome any feedback from users on subjects they would like to be included in future issues.

Please note that, while this document adheres technically to the PCI and PXI Specifications, statements on the status of various segments of the PXI market represent the opinion of Pickering Interfaces and not that of the PXI Systems Alliance, or PXISA.

Latest Specifications as of the date of publication:

| Title | Publication Date |
|---|---|
| PXI Specification – PCI eXtensions for Instrumentation. Rev 2.0 | 07-28-2000 |
| PXI Specification – PCI eXtensions for Instrumentation. Rev 1.0 | 08-20-1997 |
| PXI-1 Hardware Specification Rev 2.3 | 05-31-2018 |
| PXI-2 Software Specification Rev 2.6 | 03-20-2020 |
| PXI-4 Module Description File Specification Rev. 1.2 | 05-31-2018 |
| PXI-5 PXI Express Hardware Specification Rev. 1.1 | 05-31-2018 |
| PXI-6 PXI Express Software Specification Rev 1.4 | 03-20-2020 |
| PXI-9 PXI and PXI Express Trigger Management Specification Rev 1.1 | 05-31-2018 |



pickeringtest.com



We want to thank Emerson Test And Measurement (Formerly NI), Keysight, VTI, Concurrent Technologies, VPC and MAC Panel for giving us permission to use photos of their products in the PXImate.

## CONTENTS

# 1

# MODULAR TEST AND STANDARDS

## Introduction

First of all, welcome to the newest edition of our PXImate, a practical guide to using PXI. Before we delve into the technology of this 25+ year-old standard, we need to make a clear delineation on the definition of "Modular Test" - we have heard from relatively new test engineers who were not clear on what "Modular" means. This term is used by many test companies and in most cases today, this has the same meaning as "PXI". That is because PXI is a "Standard" that embraces the concept of Modular Test. But Modular is not always a standard.

Wikipedia says this about the term modular - "A design principle that subdivides a system into smaller parts called modules, which can be independently created, modified, replaced, or exchanged with other modules or between different systems". This term is applied to modular homes, modular furniture and so on. But we are focused on functional test. So, please read on.

First a little history of Test Instrumentation. The Test & Measurement (T&M) instruments used in verifying the functionality of an electronic device have evolved over the decades of electronics manufacturing. Early instruments – we will call them "Old School" – are single function, stand-alone devices, as shown in Fig. 1.1.

**Fig. 1.1 -** *"Old School" GPIB based Instruments*

They each have their own power source, front panel control and often some communication protocol like GPIB, which is a parallel bus communications standard introduced in the mid-1970s and is still in use today. In the past, programming commands tended to be proprietary formats. When you use a bunch of these instruments, it can be very expensive, complex to cable everything together and it can be rather large.

"Modular" as it applies to test systems was probably first applied in 1987 when the VXI (VMEbus Extensions for Instrumentation) Consortium was formed. The companies Colorado Data Systems, Hewlett-Packard, Racal Dana Instruments, Tektronix and Wavetek formed a consortium to take the existing VME Bus (created by Motorola in 1981 but focused on defense related systems) and created a standard for test and measurement. The VME Bus was then over six years old and supported by many companies, so it was stable. But the companies we mentioned earlier wanted to add features that all advanced functional test systems utilized – these included instrument synchronization and trigger buses. So, in 1995 VXI 1.0 was released.

VXI was the first standard to specify instruments with no front control panels. Software had advanced to the point where all operations could be performed by the host computer. Modular also meant that all instruments plugged into a common chassis, providing power, backplane control signals, synchronization, and trigger buses. This design helped to produce smaller test systems as you no longer needed so many rack-mounted instruments, AC power connections and control cabling. The military and the prime contractors worldwide seized on this form factor and it became the de-facto standard for functional test in manufacturing and repair depots in the 1990s and early 2000s. A VXI Chassis with Instruments is shown in Fig. 1.2.

**Fig. 1.2 -** *VXI Chassis and Instruments*

For those readers familiar with PXI, it all sounds similar, right? Well, PXI (PCI Extensions for Instrumentation) took the modular standard to a smaller size and different communications protocol but kept the same concept of modular for test. But more on the PXI Standard in Section 2. The bottom line for you as a test engineer is that when you purchase an instrument, chassis, switching or other products from companies with a PXI Logo they will all “play nice” together in your test system. A fully populated PXI chassis is shown in Fig. 1.3.

**Fig. 1.3 -** *Fully Populated Pickering PXI Chassis*

## Other Modular Standards

As mentioned earlier, modular test in our industry usually applies to PXI (there is also a modular standard called AXIe but the modules based on this standard are not compatible with PXI hardware). However, many T&M companies have developed other modular test platforms that are proprietary. For example, Pickering Interfaces created a modular chassis based on the LXI (LAN-based eXtensions for Instrumentation) standard. LXI is not truly modular, but the specifications allow for it. The lack of mechanical specifications in the LXI spec allowed Pickering to create a modular standard that accomodates larger modules for applications like RF, Microwave and High Voltage that are better addressed by a larger form factor than PXI. For more information on LXI, please visit the LXI Consortium website 'lxistandard.org'.

Other companies have also created their own test system formats. Companies like the VTI division of Ametek, Keysight and Emerson Test & Measurement (formerly NI) have created T&M platforms that are based on a proprietary modular architecture. Some examples are shown in Figs. 1.4 to 1.7.

**Fig. 1.4 -** *Pickering Model 65-231-9XX (9 kV Switching)*

**Fig. 1.5 -** *Keysight 34980A*
*Multifunction Switch/Measure Unit*

**Fig. 1.6 -** *VTI EX1200*
*Multifunction Switch/Measure Mainframe*

**Fig. 1.7 -** *NI CompactRIO*

If you are designing a test system that is focused on PXI, you can still consider proprietary modular products as most T&M companies have created software environments that work well with other platforms. In the end, consider the best solutions for your application whether the focus is specifications, adaptability, or price.

**Table 1.1 - Comparison of Modular PXI Test Systems Configuration to Conventional (Stand-alone instruments) Test Systems**

| | PXI | Conventional |
|---|---|---|
| **Size** | Compact, as up to 20 instruments fit in a 4U (6 in. or 15.24 cm.) chassis. | Much larger. In this example, the 20 instruments could occupy 36U (54 in. or 137 cm) or greater of rack space. |
| **AC Power** | One AC power cord per PXI chassis. | One AC power cord per instrument. In this case, 20 power cords. |
| **Synchronization of Instruments** | Integral to PXI chassis. | Multiple synchronization cables between selected instruments. |
| **Interface to Mass Interconnect (MI)** | Short cable, flex-circuit or PCB interconnects as the MI can be placed directly in front of the PXI chassis. This can provide better signal integrity. Also, switching can be installed in the same chassis, again making cabling short. | Lengthy cables from instruments to either Mass Interconnect and/or to the switching system and then to the Mass Interconnect. |
| **Cooling/noise** | Fans integral to the PXI Chassis. Potentially quieter than alternatives. | Most instruments have their own cooling fans. In the example here, that is 20 fans or more which could be much noisier than a PXI chassis. |
| **Cost** | Should be much lower as you have a single AC power source, no front panel controls, and no enclosure for each instrument. | Much more expensive for the reasons stated for the PXI chassis. |

# 2 PXI OVERVIEW

## PXI Background

PXI is based on the CompactPCI (cPCI) standard, which was introduced in 1995 with the aim of providing a flexible modular architecture for building embedded systems. It is an industry-standard board format specification owned by the PICMG (PCI Industrial Computer Manufacturers Group) standards body (picmg.org), which defines both mechanical and electrical characteristics. cPCI boards can have either 3U or 6U sizes, as shown in Fig. 2.1, with standard Eurocard form factors connected to a passive backplane via rugged IEC connectors with 2 mm pin spacing and standardized pin assignments, using either a 32-bit or 64-bit PCI bus for primary communication. cPCI platforms are designed for modularity, ruggedization and longevity, and provide dense computing performance, often supported by forced air cooling.

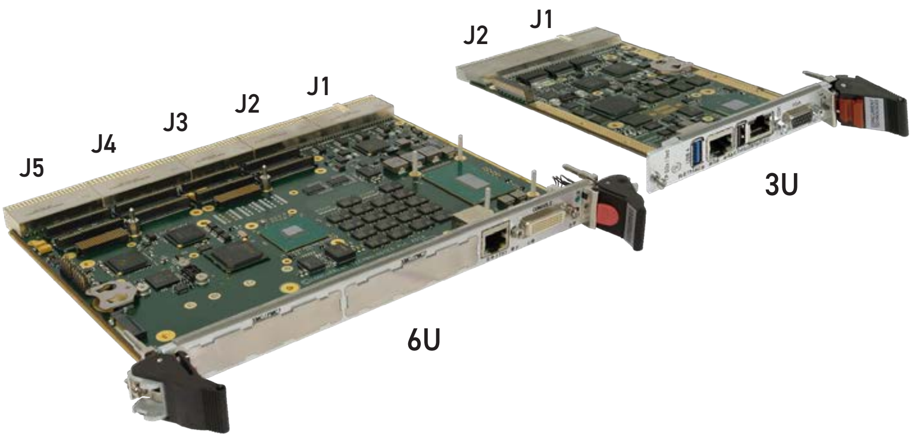



**Fig. 2.1 -** *Concurrent Technologies 3U and 6U cPCI boards*

3U boards have a J1 connector which carries the 32-bit PCI bus signals, and an optional J2 connector, which carries either user-defined I/O or the upper 32 bits of the optional 64-bit PCI bus. 6U cards have an identical J1, a J2 that is always used for 64-bit PCI, as well as J3, J4, and J5 connectors for a variety of user-defined applications.

PCI was the first universal, processor-independent computer bus to be adopted by all major microprocessor manufacturers, and the cPCI standard's success was primarily due to its adoption of this ubiquitous bus. By leveraging low-cost silicon and software developed for PCI, cPCI quickly became the most popular industry-standard modular architecture for embedded applications.

In 1997, National Instruments took the cPCI standard as a basis for a new modular Test & Measurement (T&M) architecture and, using the same model as the VXI T&M standard evolving from the VME computer bus, added several key features such as system clocks and triggers that are required for test, measurement, data acquisition and control applications. The additional PXI timing and synchronization signals were added into the J2 connectors of the base 3U and 6U cPCI modules and distributed via dedicated buses on the PXI chassis backplane. The PXI chassis J2 connectors also included access to a general-purpose analog bus, which was designed to share signals between adjacent modules in the chassis. More information on these features is provided later in this section. Fig 2.2 shows 3U and 6U PXI Modules with J2 Connectors.

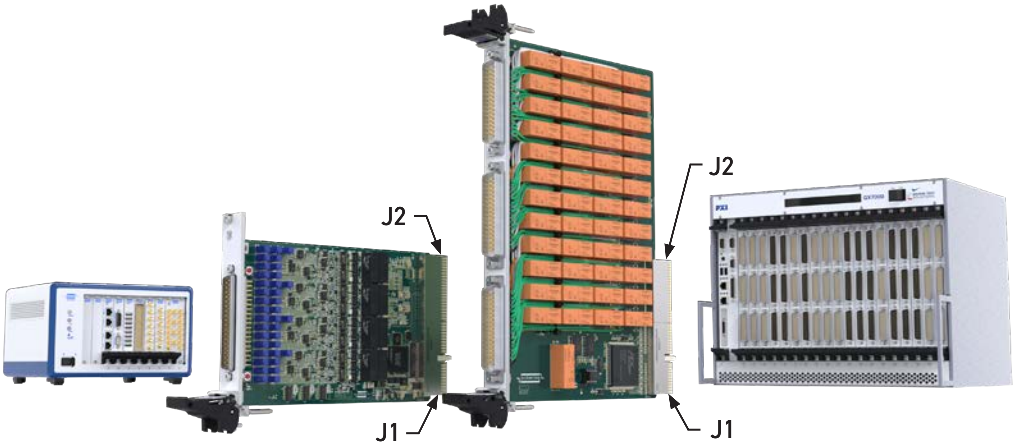



**Fig. 2.2 -** *3U PXI Module and Chassis, and 6U PXI Module and Chassis*

The software environment was also standardized, with a Windows operating system and high-level drivers.

The new standard was named PXI (PCI eXtensions for Instrumentation), and would bring mainstream PC technology to functional test systems, providing increased flexibility and higher performance while reducing capital size and cost. In order to develop the PXI concept to its full potential, National Instruments decided to make PXI an open standard and announced the creation of an industry consortium, the PXI Systems Alliance (PXISA), the other original member companies being GenRad and PXIT. Fig. 2.3 shows the signing of the consortium agreement.

**Fig. 2.3 -** *Ratification of the PXISA industry consortium by the original members in 1998. The gentleman on the left is Bob Stasonis, then at GenRad, who joined Pickering Interfaces in 2003 and is a co-author of this publication.*

With an open architecture based around the ubiquitous PCI specification, PXI adoption grew very rapidly and a within a year over 40 companies had become PXISA members, including Pickering Interfaces, who has been a Board-level Sponsor member ever since.

A notable evolution of the standard was the release in 2005 of the PXI Express (PXIe) specification, fuelled by the earlier development of PCI Express technology. PXIe significantly increased data bandwidth and improved synchronization capabilities. However, the PXISA were careful to implement the PXIe bus connectors in such a way that PXI and PXIe modules could co-exist in a special hybrid version of a PXIe chassis, thereby maintaining full backwards compatibility. This is essential as there is a huge installed base of PXI products, with most vendors supplying modules in both formats, and PXI & PXIe will without doubt co-exist for the foreseeable future.

## PXI Basics

PXI is a modular industry-standard platform for test, measurement, control and automation systems. It uses commercial PC-based PCI and PCI Express bus technology combined with rugged CompactPCI modular packaging, and integrates powerful timing and synchronization features. PXI is a high-performance, low-cost deployment platform and has become the dominant modular instrumentation standard for electronics Test & Measurement applications, with thousands of products that address markets such as aerospace & defense, semiconductor, automotive, rail, marine, communications, energy, medical and industrial.

Launched in 1997, PXI is an open industry standard governed by the PXI Systems Alliance (PXISA), presently with a membership of around 60 global companies chartered to promote the standard, ensure interoperability and maintain the PXI specification across its mechanical, electrical, and software architectures. This ensures that integration and software costs are minimized and allows trouble-free multi-vendor solutions to be implemented. Being an open standard architecture, any supplier who joins the Alliance can create compliant PXI modules safe in the knowledge that these will seamlessly operate in any PXISA member's chassis and with other PXISA members' modules.

In 2005 the standard expanded to cover two physical implementations of the PCI bus, namely PCI (later often referenced to as Classic PCI) and PCIe. These two versions of the bus are largely software compatible but not mechanically or electrically compatible. The two versions are referenced as PXI and PXIe, where PXI uses the multi-drop parallel bus structure of PCI and PXIe uses the point-to-point serial interface of PCIe. PXIe chassis, as shown in Fig. 2.4, are available in a hybrid configuration that supports both control methods to provide support for either style of module. The global acronym PXI(e) will be used throughout this publication to indicate that a statement applies to both PXI and PXIe.

**Fig. 2.4 -** *Pickering PXIe Chassis with Embedded Controller and Mix of Vendors' Modules*

# PXI Overview

The PXI and PXIe physical form factors are based on the cPCI and cPCI Express standards respectively, but with the addition of connections used to support timing/triggering functions and a local bus. cPCI and PXI modules are interchangeable - they can be used in either cPCI or PXI chassis - but installation of PXI modules into a cPCI chassis removes any ability that a PXI module has to support the dedicated hardware triggers, clock synchronization and local bus of PXI. The same applies to PXIe and cPCIe modules. The PXI(e) modules providing the required functionality for a user's application are plugged into a chassis which may include its own embedded controller or a PCIe to PXI or PXIe bridge that provides a high-speed link from a remote PC. In use, a PXI(e) system appears as an extension to the PCI or PCIe slots in the user's controller, regardless of whether the controller is embedded in the chassis or is a separate computer. PXI(e) controllers can have Windows, Linux or virtually any real-time operating systems.

Modules can be controlled in either of two ways in a PXI(e) system - message or register based. Register based communications is the most common method in basic PXI(e) modules. In register-based control, the controller accesses the registers within the modules which control their actions. Message-based communication, as the name implies, controls and reads data from the peripherals by exchanging text-based messages. Messages have a well-defined format and range of arguments. Message based control was more typical of standalone instruments in the past, but there are PXI(e) modules which use this method, especially in the PXIe form factor.

Instrument drivers simplify the communication and control of PXI(e) peripherals. An instrument driver is a type of computer software that allows interaction with hardware devices. It constitutes an interface for communications between the operating system and application specific software. It allows the transmission of commands and/or the reception of data. The main reason for using a driver is that it frees users from worrying about the low-level details of device control.

The PXI standard relies on VISA drivers. VISA is the Virtual Instrument Software Architecture, a standard managed by the IVI Foundation (ivifoundation.org). VISA defines a software layer that allows users to control instruments without needing to understand the workings of the interface bus being used. It provides a way of writing instrument drivers using a consistent approach. This means that, even when dealing with instruments from different manufacturers in the same chassis, interoperability becomes far easier. Kernel and IVI drivers may also optionally be used for module control. More information on Instrument Drivers can be found in Section 6 (Software).

Since the introduction of the PXI Standard and the PXIe update, one additional option has been added - a chassis Trigger Bus Management system. More on this in the software section of this document.

Each section following this overview provides technical details on each key element of a PXI(e) Test System – chassis, modules, embedded/remote controllers and software. Finally, a new section for the PXImate, where our applications support people give you some of their "tricks" to help make PXI(e) system integration easier.

# 3 MODULES

The PXI standard is more than an electrical and software standard. The PXISA and the specifications developed also control the mechanical aspects – chassis, connectors and the physical module sizes and available PCB real estate. In this section we will elaborate on the modules' mechanical parameters and what they mean to your test system.

## Is PXI Limited?

Over the past 25 years, PXI has firmly established itself as globally the most popular modular platform for test, measurement, and automation applications, with over 2,500 modules now available from a large number of different instrument vendors. These modules cover a huge range of functionality from standard instrumentation such as digitizers, Source/Measure Units (SMUs) and digital multi-meters to complex RF instruments such as Vector Network Analyzers, together with waveform generation, current & voltage sources, digital I/O, data acquisition, all manner of switching from DC to light, sensor simulation for Hardware-in-the-Loop applications, serial communications, power supplies and much more.

A common misconception is that PXI's small form-factor means that its modules must have a lower performance than traditional boxed instruments. However, PXI's modular chassis-based architecture means that each module shares the chassis' power supplies, cooling and controller, and has access to comprehensive timing and triggering functions via the chassis backplane. This leaves practically all the module's footprint for its specific functional circuitry. In addition, PXI(e) modules may occupy multiple chassis slots if more space is required to expand their functionality. Contrast this to traditional boxed instruments which each require a controller, power supply, cooling and front panel controls and displays.

## PXI Module Architecture

The original version of the PXI standard, designated PXI-1, is based on the cPCI standard, and as such specifies both 3U and 6U modules which can support either 32-bit or 64-bit PCI operation at 33 MHz and 66 MHz bus speeds respectively. The 3U cards, shown in Fig. 3.1, are much more popular because they make for a more compact system.

**Fig. 3.1 -** *Pickering 3U PXI Modules*

CPCI and PXI modules are interchangeable - they can be used in either cPCI or PXI Chassis - but installation of PXI modules into a cPCI chassis removes any ability that a PXI module has to support the additional features of PXI.

A 3U module is 100 mm high (3.94 inches) x 160 mm deep (6.30 inches) and PXI specifies a 20.32 mm (0.8 inch) slot width - but modules may be built to occupy one or more PXI slots. There are two positions for connectors, designated J1 and J2. The J1 connector is mandatory, as it carries the 32-bit PCI bus interface.

# Modules

The J2 connector carries the 64-bit PCI bus and additional connections for the PXI clock, synchronization, triggers and analog bus. These resources are positioned on the connector as shown in the illustration below, and will be described in more detail in the PXI Chassis Backplane section. A module with J1 and J2 connectors, as shown in Fig. 3.2, fitted is designated as a PXI-1 version.

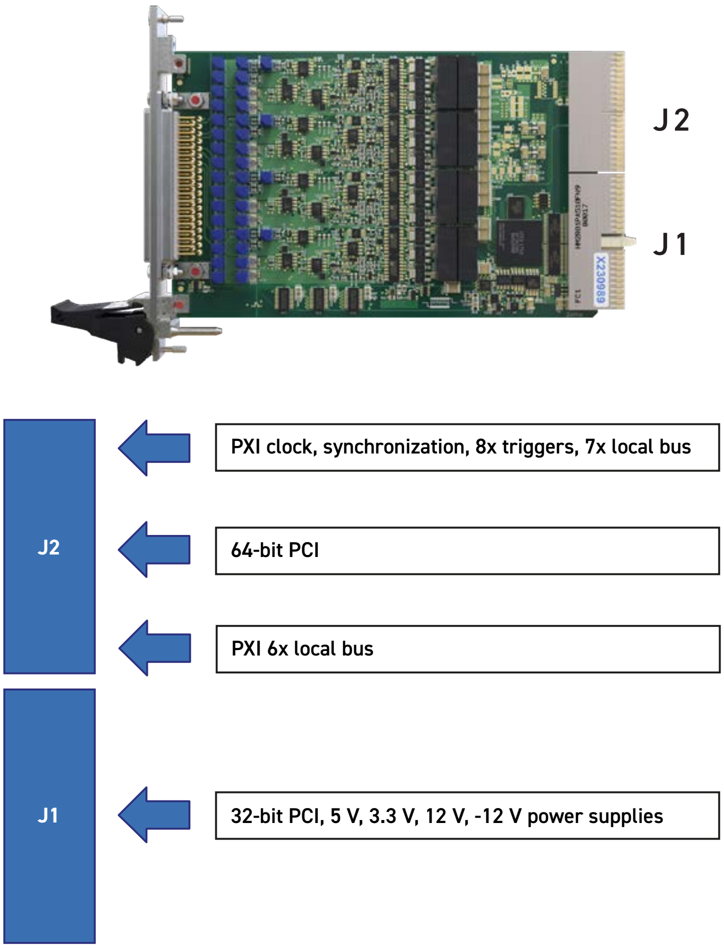



**Fig. 3.2 -** *PXI-1 Module and Connections*

In practice, all available PXI modules only support 32-bit, 33 MHz operation and the same is true of PXI chassis - the increased complexity of a 64-bit, 66 MHz backplane means the speed increase comes at a cost and conveys no great advantage to the user since speed of the backplane is rarely a limiting factor in test systems.

If the module has no functions requiring the use of the J2 connector, i.e. it is 32-bit and has no need of the PXI special features, J2 may be omitted from the module to save cost . As we will see when PXIe is discussed, PXI-1 modules that do not have J2 fitted are fully compliant with the Hybrid slots in a PXIe chassis.

Most 3U PXI modules have one ejector handle fitted at the bottom of the module, one being sufficient to enable reliable easy ejection. A screw fixing is provided at the top and the bottom, the bottom fixing being partly hidden by the ejector handle. Modules occupying more than one slot can have more than two screw fixings.

It is good practice that modules are operated with all fixings tightly secured, this is particularly important on modules that require a good ground connection. The front panel ground must be isolated from the PXI power supply ground to avoid ground loop currents. The module performance should be specified with the screws tightened.

In recent years as PXI has entered new application areas for which the restricted front panel space is an issue, there has been a tendency for some modules to drop the ejector handle.

### 6U PXI Modules

A 6U module is 233.35 mm high (9.18 inches) x 160 mm deep (6.31 Inches) and the same 20.32 mm (0.8 inch) slot width as the 3U module, with multi-slot modules allowed. While 3U is by far the most common module height, the larger 6U modules may be used for more complicated and sophisticated instruments and functions and the two may be integrated together in one chassis.

6U modules can have up to five connectors, labelled J1 to J5, as shown in Fig. 3.3. Like the 3U modules, the J1 connector carries the 32-bit PCI features and J2 the 64-bit PCI and PXI features. J3 is reserved for future expansion of the PXI hardware specification but can also be used for proprietary interconnections, as can the optional J4 and J5 connectors.

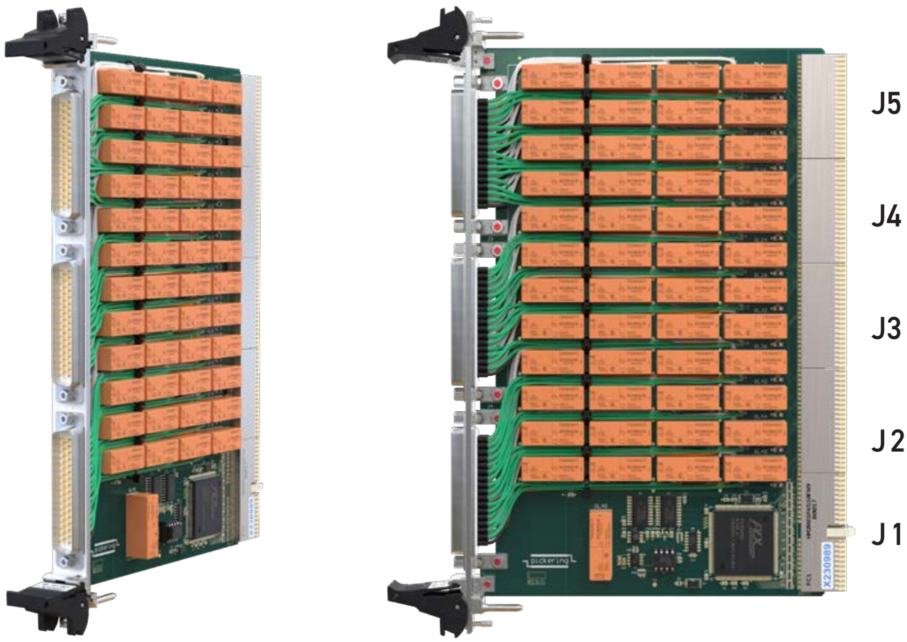



**Fig. 3.3 -** *6U PXI-1 Module*

6U modules require two ejector handles, and these should be used simultaneously to ensure the module is ejected out of the chassis connectors in a parallel fashion.

Like the 3U module, fixing screws are required at both the top and bottom of the module.

## PXIe Modules

The advent of the high-speed serial PCI Express (PCIe) bus in 2002 presented the PXISA with an opportunity to incorporate this technology into the PXI standard and take advantage of the much faster data bandwidth via one or more pairs of dedicated transmit and receive connections (called lanes) to each PXIe module. Each lane is capable of transferring data at 250 MB/s per direction in a PCIe Gen1 configuration, compared to the parallel 32-bit PCI bus speed of 132 MB/s per system. This would provide a step change improvement in performance for instruments which routinely transfer a large amount of data to and from the PXIe system controller.

The dilemma was however that PCIe is electrically and mechanically incompatible with PCI because of its completely different electrical interface and high-speed connectors. There was a large, well-established global installed base of modules based on the original PXI-1 standard which would need to be used alongside any new modules based on PCIe.

As well as the PCIe bus integration, there was also a desire in the PXI community to implement a superior timing and synchronization scheme. With speeds of circuitry ever increasing in the electronics industry, the need had arisen for higher performance clocks and triggers to enable tests requiring very tight synchronization or timing to be performed. The key to this would be to use spare connections in the high-speed PXIe connectors to implement differential clock and trigger systems that would provide higher speed, lower latency and greater noise immunity than the original PXI features. However the original capabilities would need to be retained for backwards compatibility.

The PXISA started the development of PXI Express in May 2005, in close collaboration with PICMG to ensure the PCI Express technology was correctly integrated into the PXIe backplane while still allowing compatibility with the existing large installed base of PXI-1 systems. The PXIe hardware standard (PXI-5) permitted the use of a chassis containing separate PXI-1 and PXIe compatible slots. This does however limit the flexibility of users from placing any module in any slot, and the number and locations of PXI-1 modules, shown in Fig. 3.4, and PXIe modules would be predetermined by the chassis selected. However, the cornerstone of the PXI-5 standard was an elegant and much more flexible solution: the hybrid slot. As noted previously, no PXI-1 modules use the 64-bit PCI bus, and very few make use of the local analog bus.

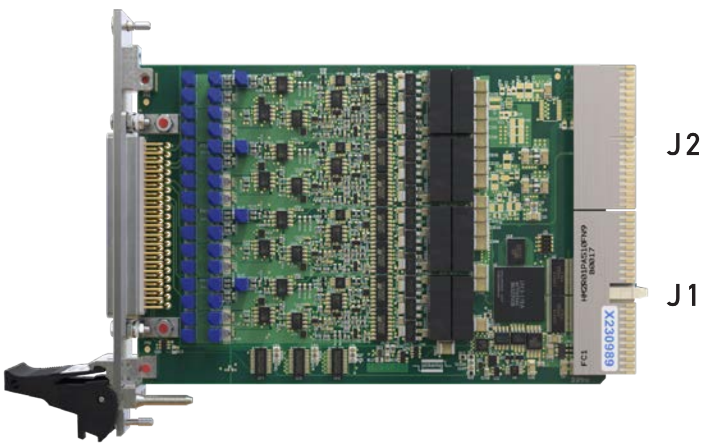



**Fig. 3.4 -** *PXI-1 Module*

This means that J2 could effectively be replaced by a smaller connector, designated XJ4, and shown in Fig 3.5, which is mechanically similar to J2 but consists of just the upper eight rows, which support the PXI 10 MHz clock, eight trigger lines, Star Trigger synchronization line and seven of the Local Bus lines.

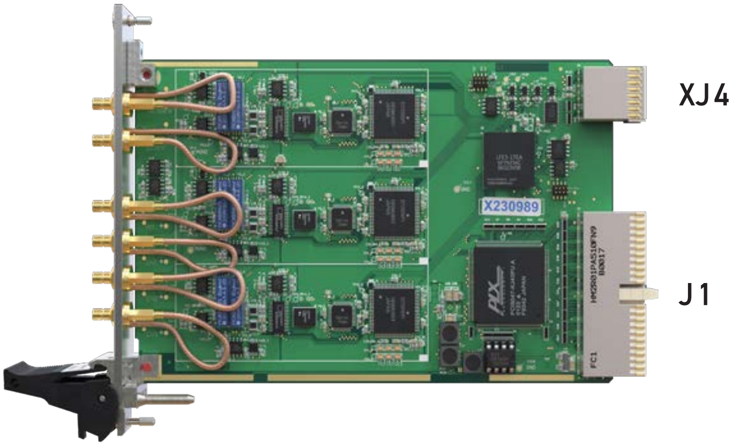



**Fig. 3.5 -** *Hybrid Peripheral Slot Compatible PXI-H Module*

Modules with this configuration of connectors are called Hybrid Peripheral Slot Compatible PXI modules, or PXI-H, and they are fully compatible with PXI-1 chassis. The same considerations apply equally to 3U and 6U modules. There is more information on PXI and PXIe compatibility in the Chassis section of this publication.

Additionally, a PXIe module, being based on the serial PCIe protocol, requires a lot less control pins than the parallel PCI bus. Therefore, a small high-speed connector, designated XJ3, and shown in Fig. 3.6, could handle all of the PCIe high-speed signaling and the new PXIe differential 100 MHz clock and Star Trigger synchronization additions, and this could be mounted on a PXIe module in such a way that it would not conflict with the positioning of the PXI J1 and XJ4 connectors.

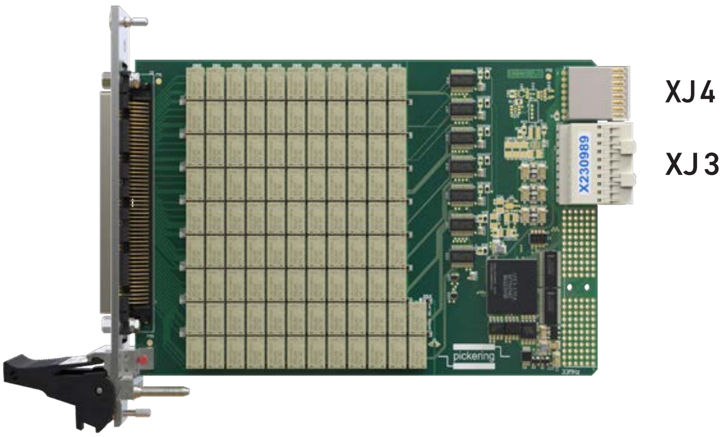



**Fig. 3.6 -** *3U PXIe Module*

# Modules

The PXIe upper XJ4 connector provides the required 3.3 V and 12 V power to the module (PXIe does not use a 5 V or −12 V supply), and backwards compatibility with the PXI-1 10 MHz clock, Star Trigger synchronization and eight PXI Triggers – however, because of the additional power supply connections for the PXIe version, only one Local Bus line can be made available .

This then means that a chassis slot, called a Hybrid slot, could be designed to accept both PXI-H modules and PXIe modules, providing a user with complete flexibility in the choice of modules for their application. A chassis can contain a large number of hybrid slots, allowing any module to be used in any slot. The next chapter will provide more detailed information on this topic. Because of the flexibility this approach offers, most manufacturers now only supply PXI-H as opposed to PXI-1 modules.

Like PXI, PXIe modules are available in 3U and 6U sizes. 6U PXIe modules, as shown in Fig. 3.7, also have XJ3 and XJ4 fitted, along with an optional XJ8 connector for additional power if required.

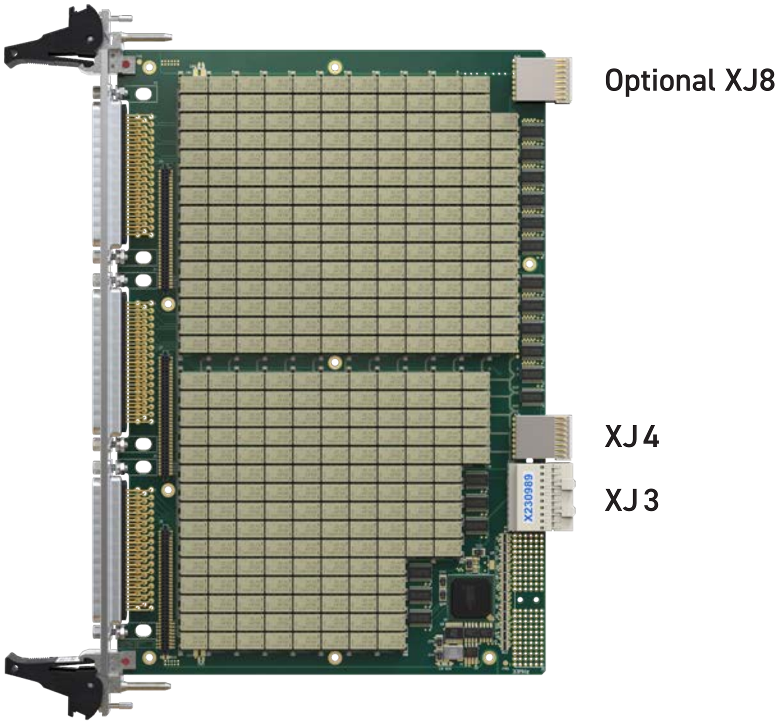



**Fig. 3.7 -** *6U PXIe Module*

# 4 PXI CHASSIS ARCHITECTURE

## Introduction

Let's first consider PXI systems – PXIe will be described later in this section.

PXI systems are composed of three main hardware components: chassis, controller, and peripheral modules.

Shown in Fig. 4.1, the PXI chassis is the backbone of a PXI system and provides the mechanical means of mounting the PXI modules and providing them with forced air cooling. It also provides DC power, the PCI bus and the PXI specific functions such as timing, triggering/synchronization; and analog bus. PXI chassis are designed to house either 3U or 6U PXI modules, but 3U modules can usually be fitted into a 6U chassis using an adaptor, and the PXI standard supports the design of chassis that allow both 3U and 6U modules to be used. The 3U size though is by far the dominant module size.

**Fig. 4.1** - *Pickering PXI Chassis with Embedded Controller*

The chassis allows cPCI modules to be supported, but they cannot take advantage of the additional PXI features.

The PXI specification supports the use of 32-bit and 64-bit PCI backplane connections at 33 MHz and 66 MHz bus speeds, ensuring theoretical bus speeds of 132 Mbytes/sec to 528 Mbytes/sec respectively, a speed far in excess of that available over GPIB or typical VXI interfaces. However, prior to this document going to publication, no PXISA member has implemented the 64-bit bus, instead opting to migrate to the faster PCIe bus.

## PCI to PCI Bridges

The specification allows up to eight chassis slots on each 33 MHz PCI segment. This does not limit the total number of slots available in the chassis however, since larger chassis include PCI to PCI Bridges to interconnect segments. Each PCI Bridge occupies one electrical slot on each of the segments it connects to and adds one clock cycle of delay per bridge.

### Slot Assignments and Glyphs

The location of a PCI Bridge is marked on the chassis and backplane by a vertical line shown between the slot numbers as shown in Fig. 4.2 and slot types (indicated by glyphs) are shown in Fig. 4.3.

**Fig. 4.2 -** *PCI Bridge Glyph*

| | |
|---|---|
|  | **Star Trigger Controller or PXI Peripheral Slot** |
|  | **PXI Peripheral Slot** |
|  | **PXI System Controller Slot** |

**Fig. 4.3 -** *Chart of PXI Glyphs*

### Slot1 – Controller

The left-hand slot of the chassis is reserved for the system controller. The slot has one set of connectors to the backplane and occupies one notional slot (refer to Fig 4.3). However, an embedded controller usually requires more than one slot and for this reason the standard allows the chassis to include controller mechanical expansion slots to the left-hand side (away from the rest of the Peripheral Modules). A typical chassis may have three Controller Expansion Slots, allowing up to the equivalent of four module widths to be occupied by the controller, but using just one set of backplane connectors in the designated Slot 1. The Controller Expansion Slots do not have backplane connectors.

Controller options include remote control modules that allow PXI system control from a desktop, workstation, server or laptop computer as well as high-performance embedded PXI controllers, all with either Microsoft Windows, Linux or a real-time OS.

## Modules

Apart from Slot 1, all other chassis slots may be used for any standard PXI modules to create the required application functionality. Each PXI slot has an associated slot number which is marked (in most cases) below the PXI slot, and these typically ascend from left to right. Slot 2 is a special case that is optionally able to accept a dedicated Star Trigger Controller – please refer to 'Star Trigger' later in this section.

## PXI Chassis Backplane Buses

The PXI-specific chassis backplane buses are shown in Fig. 4.4 and described below.

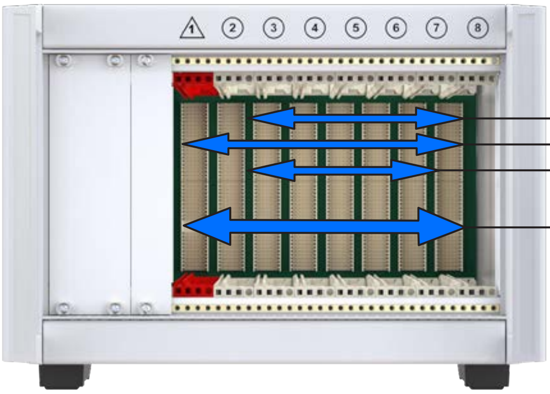



**Fig. 4.4 -** *8-Slot PXI Backplane Buses*

### Trigger Bus

There are eight PXI trigger lines available in the top eight rows of the J2 connector.

These triggers provide a method of synchronizing the operation of PXI modules. The bus is configurable so that any module can send triggers to other modules in the system and respond to events in other modules. The standard does not mandate the buffers or the support tools required. The Trigger Bus length is limited to 252 mm.

The eight PXI triggers can provide a low latency method of triggering events - it is possible for the triggers to be routed directly to the hardware. However, in most PXI implementations trigger operations do involve the driver (and in the case of IVI all involve the driver) so actual speed of operation is defined by software latency.

### Local Bus

The Local Bus is a daisy chain of 13 lines that interconnect adjacent PXI slots. Each line consists of a left and right pair – seven pairs are situated in the top eight rows of J2, and the remaining six are in the bottom three rows of J2.

Each Local Bus line on the right of the slot is connected to the neighboring Local Bus slot on its left. The bus is used to allow adjacent modules to exchange analogue signals (up to ±42 V) or digital signals directly. Software must check the compatibility of adjacent slots before the modules can make use of the facility. If modules that make use of the Local Bus are not placed in the appropriate positions, then the Local Bus functionality may be lost, since there is no requirement for modules to provide a bridge, and different modules will use the Local Bus in different ways (or more typically not at all).

The bandwidth and other characteristics of the local bus are not fixed in the PXI standard and are left to the chassis vendor's discretion. This has the effect of limiting how designers can use these connections.

### 10 MHz Synchronization Clock

The PXI backplane always provides a system reference clock operating at 10 MHz and having an accuracy of 100 ppm (parts per million) or better. The clock is specified to have a 50% ±5% duty cycle and each module is independently driven to avoid module interaction. The reference is distributed such that each slot receives the signal at the same time to within 1 ns.

For some applications the system reference clock is not accurate or stable enough, particularly for RF applications where it may define the frequency accuracy of an RF carrier. The PXI specification allows for an alternative system reference clock to be optionally provided from an external source and recommends that the backplane provides a facility to switch the clock lines to this alternative frequency source, usually supplied via a coaxial connector on the back or front of the chassis.

### Star Trigger

The Star Trigger is a high-performance synchronization system based on a special Star Trigger Controller module, shown in Fig. 4.5, together with dedicated high-speed trigger lines between chassis Slot 2 and each of the other Peripheral Slots in a PXI chassis. The position of the Star Trigger Controller slot is marked on the chassis by the Star Trigger Slot Glyph (Refer to Glyph chart). The modules in the Peripheral Slots must support the Star Trigger lines for the feature to work as defined here. The Star Trigger slot makes use of the lines on the left-hand side which would normally be part of the Local Bus. This slot is not an essential part of a chassis, but in practice most chassis provide it. It can be used as an ordinary PXI Peripheral Slot if a Star Trigger Controller is not fitted.


**Fig. 4.5 -** *NI PXI Star Trigger Controller*

When a Star Trigger Controller is installed in the chassis it can ensure that events in other peripherals are simultaneously triggered with very low levels of differential time delay between modules, due to line-length equalization techniques providing matched propagation times to each module slot. The trigger system is bidirectional so the Star Trigger Controller can allow an event in one Peripheral Module to trigger events in other modules.

The Star Trigger Controller provides thirteen output lines, each going to a different designated Peripheral Module. In a 14-slot chassis there are 12 Peripheral Slots (14 slots less Slot 1 and Slot 2) and typically one PCI Bridge (for a 33 MHz PCI bus). The Star Triggers are connected to the 12 Peripheral Slots. For a larger chassis the higher slot numbers (beyond physical slot 15) do not support the Star Trigger, and so they should be occupied by modules not requiring this facility.

For systems requiring more than one chassis, fitting a Star Trigger Controller into each chassis and connecting them together, either through cables or synchronizing them through the use of GPS timing, ensures that PXI instrumentation systems can perform more complex measurements in distributed systems.

As with the eight PXI triggers, Star Trigger operation may be executed through the driver, in which case synchronization may be dependent on software latency.

### Triggering across PCI Bridges

The presence of PCI Bridges should generally be transparent to the user of a PXI Chassis. However, if two modules require the exchange of trigger signals over the Trigger Bus, additional complications will arise because the Trigger Bus does not directly cross the PCI Bridge. The Star Trigger is wired to cross the first PCI Bridge but has a limited connection count.

If an instrument requires the use of two separate modules connected by the trigger bus, the instrument's operation can be complicated or disrupted if its modules are inserted either side of a PCI Bridge. It is therefore best to avoid dividing these modules with a bridge. The PCI Segment Divider Glyph on the front of the chassis clearly identifies the location of any PCI Bridge.

## How Bus and Device Numbers are Assigned to a Chassis Slot Number

Bus and device numbers are assigned at PC bootup by the PC BIOS. Assignments would be applied to devices internal and external to the PC. Attaching a chassis through a PC, either by way of an embedded PC or a controller installed in a PCIe slot of the PC, will present added bus and slot numbers available for PXI(e) installed modules. Identification of installed PXI(e) cards would be determined at bootup provided the chassis is powered at this time. The number of assigned bus numbers would be determined by the number of bus segments in the chassis.

Fig. 4.6 shows an example of a typical 8-slot PXI chassis:

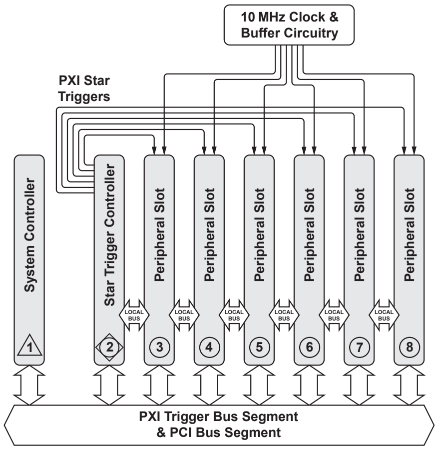



**Fig. 4.6 -** *A Typical 8-Slot PXI Chassis*

Physical slot 1 of the PXI chassis is occupied by either an embedded PC or remote control interface card. A bus segment would consist of slot 2 (Star Trigger or peripheral card) and peripheral slots 3, 4, 5, 6, 7 and 8. From the diagram above you can see that there are local bus and trigger capabilities associated with this bus segment and slot numbers.

As stated earlier, the PC BIOS will assign bus and device numbers to each installed PXI card at bootup. In the 19-slot chassis shown in Fig 4.7, the PC BIOS may assign bus 3 to cards installed in slots 2, 3, 4, 5 and 6 and a device number for each individual card.

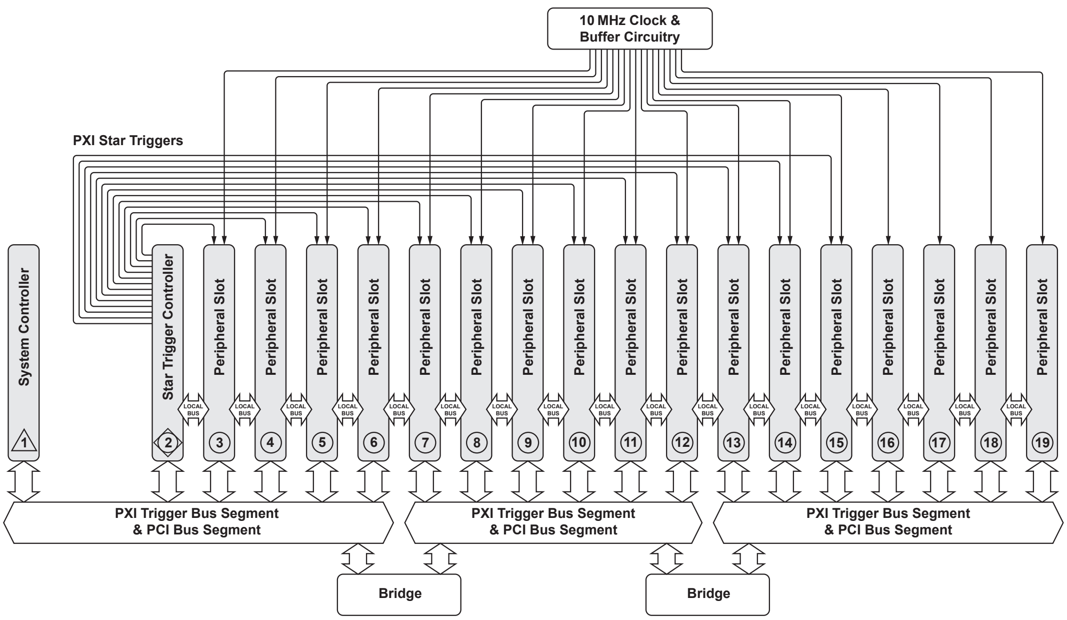



**Fig. 4.7 -** *A Typical 19-Slot PXI Chassis*

So, to access a card in physical slot 4 would be by referring to a bus 3 and a device number assigned also assigned by the BIOS. In this way each installed card will have a unique bus and device number. Slots 7, 8, 9, 10, 11 and 12 may be assigned to bus 4 and slots 13, 14, 15, 16, 17, 18 and 19 may be assigned to bus segment 5. If used in a Windows environment all the bus and device information can be determined with some low-level functions or by running Windows Device Manager.

One thing to keep in mind is that the PC BIOS determines the assigned bus and device numbers. Any change of a chassis, a PC or if a card is moved from one slot to another slot the assigned bus and device numbers will change. When moving a test program from one test configuration to another may require you to modify the test program to reflect the difference in assigned bus and device numbers.

## PXI Express (PXIe)

As the demands on PC speed grew, the speed of the PCI bus increasingly became an issue, the concept of a multi-drop system based on a parallel bus structure becoming ever harder to scale to escalating PC performance requirements.

A breakthrough was made with the introduction of a fast serial interface, PCI Express or PCIe, which carries data on wire pairs usually referred to as PCIe lanes. So far there have been four generations of PCIe, referred to as Gen 1 to Gen 4. A single lane is up to 2 GB/s bandwidth (Gen 4); however, lanes can be aggregated into higher data rate connections, with four lanes to sixteen lanes commonly available. Each serial connection is point to point, so a particular connection only carries traffic destined for the node connected at the end of it (and any connections downstream of that device) and has no un-terminated transmission line stubs to distort the high-speed waveforms. This allows the speed of each lane to be increased as technology improves. This serial interface system is inherently more scalable than a parallel bus system.

PCIe was introduced at Gen 1 with a raw bit rate of 2.5 Gb/s (2 Gb/s after decoding) in an 8-lane configuration. Subsequent generations have increased that data rate and implementations have increased the number of lanes to provide greater data bandwidths. Mechanisms are provided that transparently cause the data rate to drop if a lower rate (by Gen number or lane count) device is connected downstream of a higher speed connection.

The data connection rate is chassis, chassis slot and module dependent and as always, the greater the data rate, the more the cost of implementation. Other than this, users are unaware of the data management processes going on at the PCIe interface.

The structure is a tree style as shown in Fig. 4.8. A single PCIe connection expands into multiple connections below it which in turn can branch into further connections. For connections at the tree's trunk (Root Complex), a high BW is needed to maximize the data capacity since it must support all the end points on the downstream side.

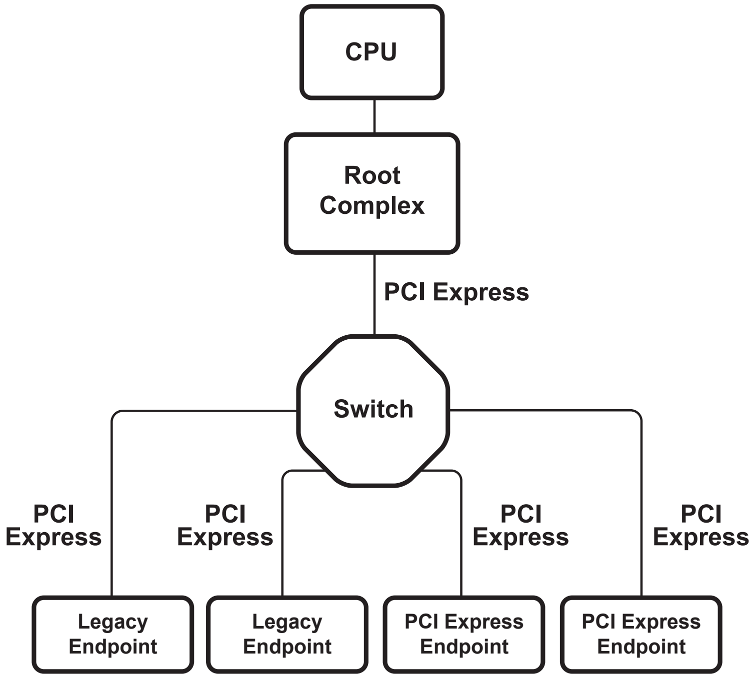



**Fig. 4.8 -** *PCI Express Architecture*

As with PCI, all traffic flows to and from the Root Complex, and actual speed is dependent both on the PCIe interface and the ability of the controller to handle all the data and drivers. PCIe was added to the PXI standards as PXIe. As with PXI and PCI there were extensions required to add test measurement features in creating a PXIe standard.

### PXIe Bus Enumeration

The bus enumeration on a PXIe Chassis is a little different to that of a PXI Chassis. In PXI, the location is defined by a Bus Segment and a Bus Device because each Bus Segment can support multiple Peripheral Modules. In PXIe, the End Points are simply one device, so essentially there is a Bus Segment for every device connected and additional Bus Segments for connecting buses. The consequence of this is that PXIe systems inherently have a much higher number of Bus Segments than PXI systems, and in some cases, this can lead to problems because older-version controllers designed for fast boot time may not fully enumerate deep PCIe bus systems. For this reason, PXI vendors often recommend a limited range of controllers for use on PXIe, the range being model specific and not just PC vendor specific. A PC vendor may have different models with differing enumeration capabilities.

### PXIe Chassis

A PXIe Chassis, shown in Fig. 4.9, uses a similar mechanical principle to a PXI Chassis, but the backplane and the connection to the PXIe modules is different both electrically and mechanically. A chassis may include support for both PXIe and PXI modules as discussed in a later section of this publication. For simplicity, the following describes a pure PXIe chassis and concentrates on the 3U rather than the 6U format.

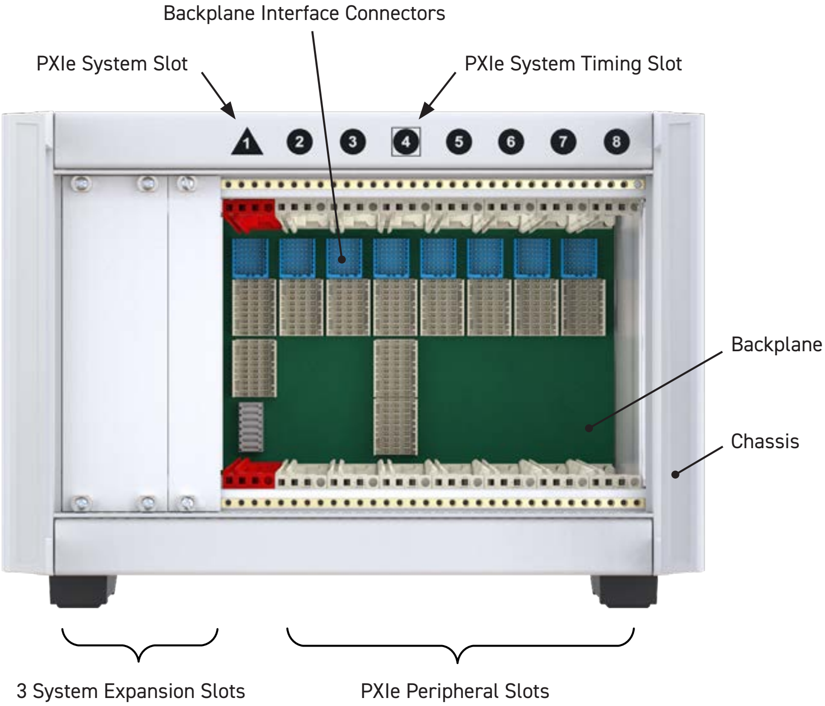



**Fig. 4.9 -** *A PXI Express Chassis Showing Slot Types*

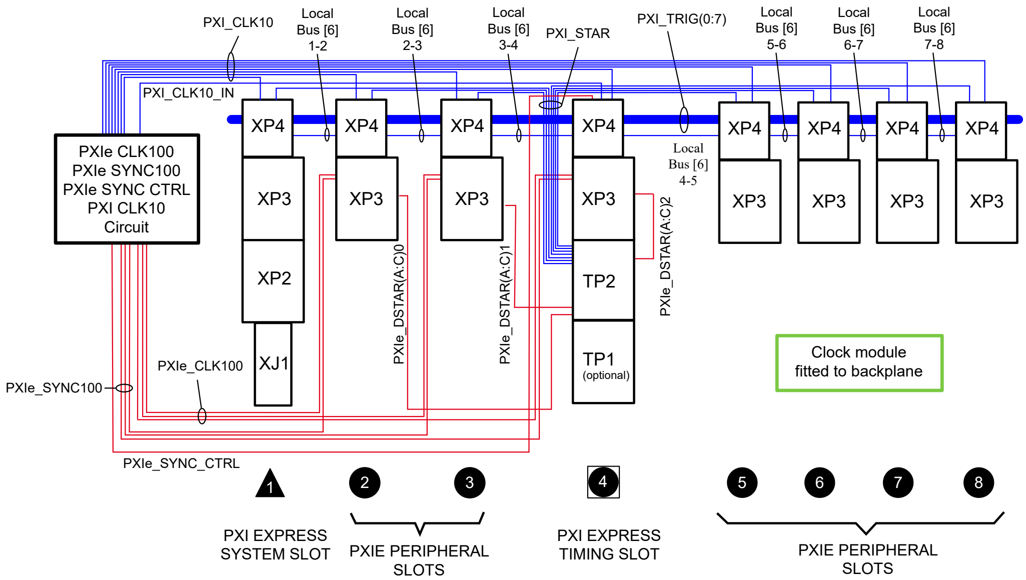
The glyphs underneath each slot are visually different to PXI so that there is a clear indication this is a PXIe slot. PXIe uses a dark background with white slot numbering whereas PXI uses a light background with dark slot numbering. A PXIe Glyph Chart is shown in Fig. 4.10.

| | |
|---|---|
| | **PXIe Peripheral Slot** |
| H | **Hybrid Peripheral Slot (accepts PXI Hybrid and PXIe cards)** |
| | **PXIe System Controller Slot** |
| | **PXIe System Timing Slot or PXIe Peripheral Slot** |

**Fig. 4.10 -** *Chart of PXIe Glyphs*

### Backplane

A PXIe backplane, as shown in Fig. 4.11, uses PCIe connections rather than PCI connections to provide a control interface. The PCIe connection can essentially be of any PCIe generation (currently Gen 1 to Gen 4) and can have a differing number of lanes connected to each slot. This is again an important difference from PXI, as slots may not be all the same. A user who has a module that demands high data connection speeds needs to choose the slot it is placed in to maximize the data bandwidth. If a “low” data rate module is placed in a high data rate slot the PCIe will adjust the data bandwidth to suit the module, if a “high” data rate module is placed in a “low ” data rate slot then the module will be serviced at the slot data rate. Your chassis supplier will be able to assist you here.




| | |
|---|---|
| XP2, XP3 | Connector carrying PXI Express Interface |
| XP4 | Connector equivalent to partially populated P2 |
| XJ1 | PXI Express System Slot connector |
| TP2 | PXI Express Timing Slot connector (TP1 is optional) |

**Fig. 4.11** - *PXIe Backplane*

Some chassis include a means of configuring the backplane lane connections to increase their flexibility. In particular, they allow lanes to be concentrated into a few interfaces so that data hungry modules can be serviced more rapidly and given greater bandwidth than other slots.

There are several other differences in the detailed implementation of the backplane on PXIe. A high-frequency System Reference Clock is added, which is a low jitter differential 100 MHz clock completely interoperable with the PXI 10 MHz clock via a new Clock Synchronization signal. All clocks/timing signals are generated by a PXI-5 compliant PXIe Clock Module fitted to the rear of the backplane. A new high-performance module synchronization system is also included, based on point- to-point differential signaling rather than the PXI-1 multi-drop single ended signaling. One limitation of PXIe is that the chassis only supports one Local Bus line to connect adjacent modules together as opposed to the 13 lines in a PXI-1 chassis, so vendors have by and large removed their dependency on this feature.

### System Slot

The System Slot can be used to host an embedded PC or a remote controller interface (including those based on the External Cable PCIe standard). Note that this controller slot is different to that used in PXI, the two are mechanically and electrically incompatible; a PXIe controller cannot be used in PXI, or vice versa.

### System Timing Slot

PXIe introduces new 3U and 6U System Timing modules with dedicated chassis slots. The location of the System Timing slot is not mandated by the PXIe specification, but if fitted it is usually located centrally in a chassis, i.e. in an 8-slot chassis, it will typically be slot 4 and in an 18-slot chassis, slot 10. The system timing slot can also accept a standard PXIe peripheral module if a System Timing Module is not fitted.

A System Timing Module shown in Fig. 4.12, provides three individual high-frequency differential star trigger/ clock signals to all peripheral module slots as well as distributing the original PXI-1 star triggers, and also manages the System Reference Clock.

**Fig. 4.12 -** *NI PXIe System Timing Module*

Some PXIe chassis manufacturers opt to exclude the System Timing slot and associated trigger lines from the chassis to save cost, and it is often replaced with a Hybrid peripheral slot for increased flexibility of module selection.

## PXI/PXIe Compatibility

As can been seen from the previous sections there are significant differences between PXI and PXIe, and that presents a backward compatibility problem. The number and variety of PXI modules available from different vendors means that user choice of modules is constrained if a PXIe only chassis is used, and that is less likely to be the case as PXISA members release Hybrid versions of their chassis. Many applications, including switching, do not need high data bandwidths and the differences in power supply specifications and trigger operations mean that many vendors may choose not to use PXIe. This was anticipated when the PXIe standard was created, so the standard allowed for the combining of PXI and PXIe in the same chassis. There are two ways of managing this compatibility, the inclusion of dedicated PXI-1 slots or Hybrid Peripheral Slots, as shown in Fig. 4.13.

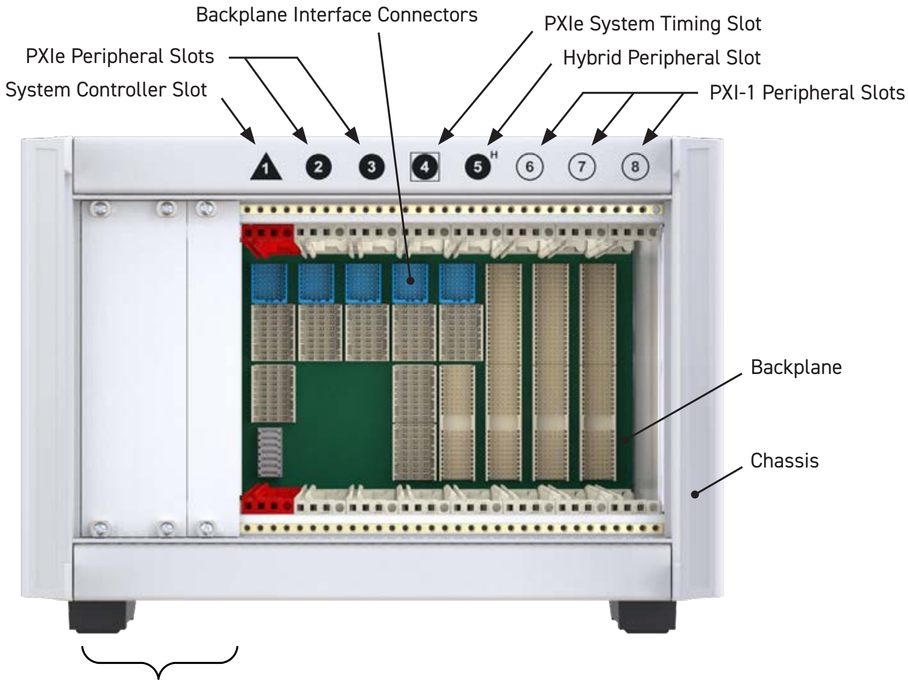



**Fig. 4.13 -** *A PXI Express Chassis with PXIe, Hybrid and PXI-1 Peripheral Slots*

### PXI-1 Slots

A PXIe chassis can also contain a number of PXI-1 slots, specifically designed to support PXI-1 and PXI-H modules. These have the conventional 32/64-bit PCI and PXI compliant power supplies. Only the PCI interface can be used to control the module, and the slot should accept any PXI-1 module. Some features may be restricted (for example Star Trigger, Local Bus) and only a few slots may be supported in the chassis. It is the lowest cost (to the manufacturer) route to increase the flexibility of the chassis but means that users must be careful about selecting modules to match the available slots.

### Hybrid Slots

In Hybrid Slots both the PCI and PCIe interfaces are present. The chassis delivers power compliant to both versions of the standard in every hybrid slot. Only one interface per slot is permitted to be used at a time, and that must use the corresponding set of power supplies and triggers.

From a user perspective, Hybrid Slots in a PXIe Chassis provide the flexibility to use either type of module. Hybrid-ready PXI modules, which are often not high data users, can be placed in any Hybrid Slot. This has resulted in chassis being offered that are entirely Hybrid with the exception of the System Controller and the Trigger Slot (optional). It tends to be a more expensive solution to manufacture because the chassis has to include both sets of power supplies and both PCI and PCIe interfaces on every slot. The PXI modules can only use one of the Local Bus connections, but it is unusual for PXI modules to use the Local Bus.

PXI and PXIe modules can be easily identified by their connectors as shown in Fig. 4.14. Here you see a standard PXI-1 module, a PXI-5 Hybrid PXI module, and a PXI-5 PXIe module. The PXI-5 Express module has two connectors labeled XJ3 and XJ4.

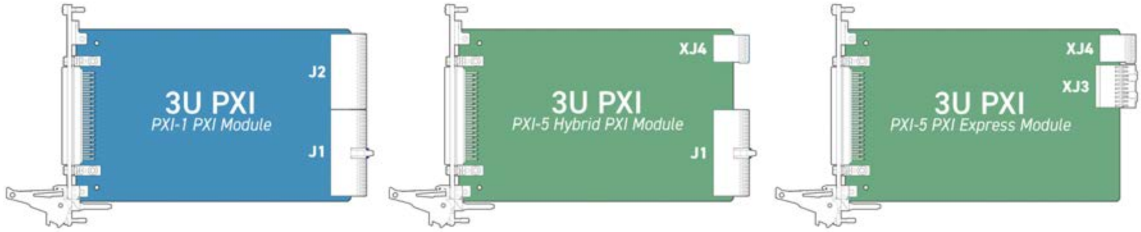



**Fig. 4.14 -** *Identifying PXI and PXIe Modules by their Connectors*

Fig 4.15 shows that a PXI-1 module will only plug into a full PXI-1 slot, whereas both PXI-5 Hybrid modules can plug into a PXIe "Hybrid" slot.

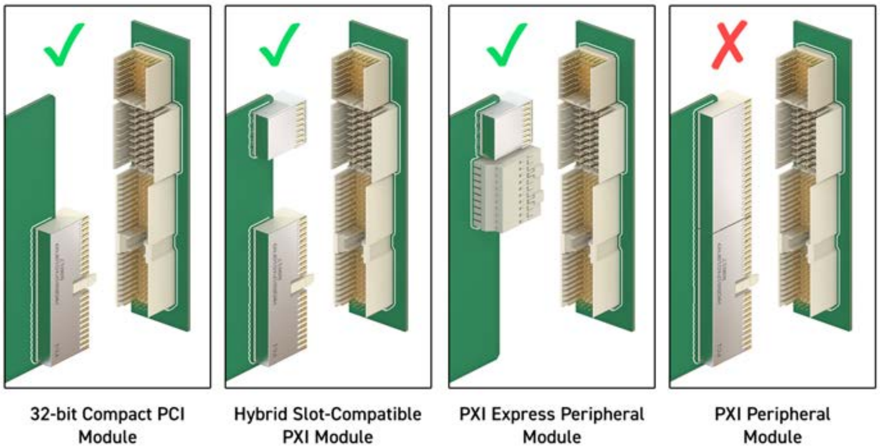



**Fig. 4.15 -** *Hybrid Slot PXI and PXIe Module Connections*

The controller for a Hybrid Chassis must be a PXIe controller, not a PXI controller. The PCIe interface from the controller supports the PXI modules by the inclusion of a PCIe to PCI Bridge to supply the PXI modules in Hybrid and PXI-1 Slots. This clearly creates a little more complexity in the backplane design and adds more PCI bus segments.

Typical PXIe Hybrid Chassis Backplane Architecture is shown in Fig. 4.16.

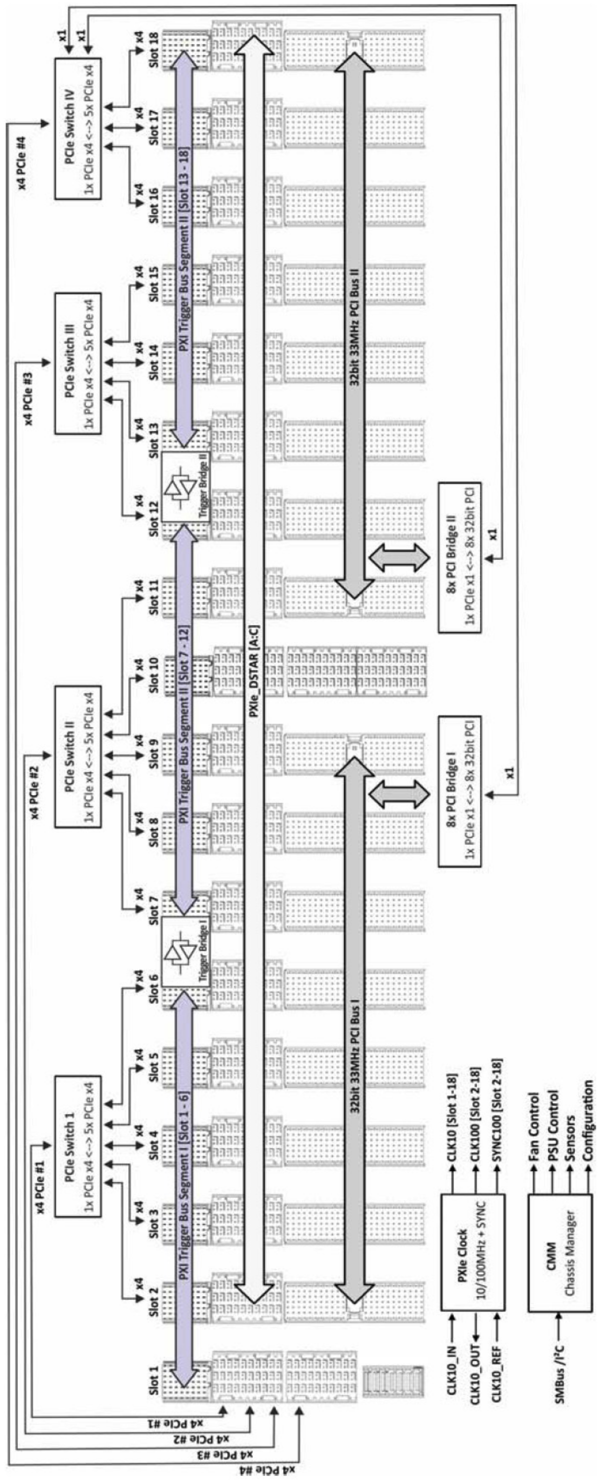



**Fig. 4.16 -** *Typical PXIe Hybrid Chassis Backplane Architecture (42-925)*

### Hybrid Chassis Power Requirements

A Hybrid Chassis must support the power needs for both PXI and PXIe modules, but it does not need to supply the power to both the PXIe and the PXI connections per slot simultaneously. The chassis needs to meet the power requirements set for both PXI and PXIe on a per slot basis (but not at the same time on any individual slot). This is of no great concern on 3.3 V and 12 V supplies where the PXIe Chassis places greater demands on the power supplies, but users need to check the 5 V and -12 V supply capability, the latter having no PXIe equivalent for peripheral modules.

## PXI Chassis Selection Considerations

As you have learned in an earlier section, a chassis is integral to the PXI Architecture. It provides DC Power, synchronizes the instrumentation and cools the modules. So, depending on the application, there are a myriad of choices in chassis from a number of manufacturers. For example, PXI/PXIe/Hybrid, cooling capacity, chassis noise, max power and so on are specifications that should be factored in your decisions. In this section, we will look at each of these considerations and give you questions to ask of your engineers and your PXI chassis vendor.

#### Slot Count

There are PXI(e) chassis available with 1, 2, 4, 5, 6, 8, 9, 10, 12, 14, 18, 19 and 21 slots! Do you purchase just enough slots for your applications or plan for expansion? Examples are shown in Fig. 4.17 and Fig. 4.18.

**Fig. 4.17 -** *PXI Chassis Choices Depending on your Test System Needs*

**Fig. 4.18 -** *Chassis as Small as 2 Slots with a Thunderbolt Interface are available*

In most cases, there are four slots allocated for the embedded system controller. There are also chassis that only support 1- or 2-slot controllers, which allows for more instruments and switching slots in one chassis. A possible downside is that the smaller controllers have fewer peripheral options like USB and Thunderbolt ports.

## PXI(e) System Controllers

The system controller controls the PXI(e) modules, executes tests and returns a pass/fail status. Controllers can be embedded, as shown in Fig. 4.19, or external to the PXI(e) chassis, as shown in Fig. 4.20. Embedded controllers are full-featured, compact PCs featuring CPU, memory and a variety of communications interfaces. A monitor, keyboard, and mouse are all connected to the front panel of the controller.

**Fig. 4.19 -** *Pickering Single Slot PXIe Embedded Controller*

**Fig. 4.20 -** *Pickering PXIe to PCIe Remote Controller*

The choice of which PXISA Member's controller you choose to purchase should not be dictated by the chassis manufacturer. All PXI(e) embedded controllers are designed to interoperate with all compatible chassis and modules. The choices should be made based on the test application – how much data will be generated and collected, RTOS (Real Time Operating System) test requirements, how many peripheral ports (USB, GPIB, Thunderbolt, etc.) are needed, Windows and/or LINUX support, cooling capacity of the chassis and other requirements. Your embedded controller manufacturer should be able to help you make the right choice. See the Controllers section for more detail.

## Getting Started

Before you select a chassis, what are you testing? Then based on the answers here, what types of instruments, switching and power supplies will you need? Some of the questions include:

- How many PXI(e) slots are needed? Many complex instruments and large switch matrices can occupy multiple slots. The modules can be 2, 4, 8 or even 12 slots wide. So, you may have to plan for multiple PXI(e) chassis.
- What instruments and switching are available for your application? Are they available in PXI, PXIe or both? These answers will determine whether your test system can use a PXI chassis, a PXIe chassis or a hybrid.

  - PXI(e) instruments and switching are available in 3U and 6U heights. The 3U is by far the most popular and has the broadest choices. But there may be a unique 6U module that fits your needs. Remember that 3U PXI(e) modules can fit in a 6U chassis provided you have an adapter that fills the space above the module for cooling purposes.
- What is the power budget for this test system? There are many choices in chassis and are dependent on their focus (there is a low power specification for portable test applications, and they have a limited power budget - more on power budgets later in this section) and may also be based on the sale price of the chassis. Picking the wrong chassis can cause intermittent false failures or even passes when an instrument does not have adequate power to make a stimulus or measurement, or the switch matrix fails to close a relay when there is inadequate power. There are minimum current requirements per slot in the specification but look at the datasheet of each module before looking at PXI(e) chassis. Keep in mind that in a multiple chassis' application, you may be able to allocate the high-power modules to a chassis with a more robust power supply and save money by putting the lower power modules in a chassis that is less power robust.
- Where will the chassis be used? On a test bench or in a cabinet? In the case of the former, you may want to select either a low noise option or a thermally controlled fan system that adjusts fan speed and subsequent noise based on the chassis temperature.
  - How is the cooling system oriented? If it takes air in from the bottom and out the back of the chassis, you will need to allow at least 1U of rack space below to allow for the cooler air intake. Some chassis take air in from the front and discharge it out the back. This design allows chassis and instruments to be stacked on top of each other saving valuable rack space.
  - How much heat will be dissipated by each module? The original PXI Spec mandated 25 W of cooling per slot. The PXIe specification is slightly more robust. In any case, consider putting modules that dissipate less heat between any high-power modules so they share the cooling capacity better. There are higher performance PXI and PXIe modules that exceed this 25 W recommendation, in some cases up to 80+ W. Be certain that the chassis you choose can support the cooling requirement of the modules required for your test application.

**Table 4.1 -** *PXI Power & Current per Module Type*
*(As PXI Hardware Specification 2.3 - 5/31/2018)*

| PXI-1 | System Controller | System Controller | Peripheral Slot | Peripheral Slot | All | All |
|---|---|---|---|---|---|---|
| **Power Supply Voltage** | +5 V | +3.3 V | +5 V | +3.3 V | +12 V | -12 V |
| **Current Requirement** | 6 A | 6 A | 2 A | 2 A | 0.5 A | 0.25 A |

Note: Minimum slot cooling is 25 W per Peripheral or Timing Slot

**Table 4.2 -** *PXIe Power & Current per Module Type*
*(As PXI Express Hardware Specification 1.1 - 5/31/2018)*

| PXIe | 5 V | 3.3 V | +12 V | -12 V | 5 V AUX | Total Power |
|---|---|---|---|---|---|---|
| System Controller Slot with 2 or More Expansion Slots | 9 A | 9 A | 11 A | N/A | 1 A | 140 W |
| System Controller Slot with 1 Expansion Slot | 2 A | 6 A | 4 A | N/A | 1 A | 60 W |
| System Controller Slot with no Expansion Slots | 1 A | 3 A | 2 A | N/A | 1 A | 30 W |
| PXI Express Peripheral Slot/ System Timing Slot | N/A | 3 A | 2 A | N/A | 0 A | 30 W |
| Hybrid Slot | 2 A | 3 A | 2 A | 0.25 A | 0 A | 30 W |
| PXI-1 Peripheral Slot | 2 A | 2 A | 0.5 A | 0.25 A | N/A | 26.5 W |

Note: Minimum slot cooling is 30 W per PXIe Peripheral or Timing Slot

**Table 4.3 -** *PXI Power & Current per Module Type*
*Low Power Configuration*

| | 5 V | | 3.3 V | | +12 V | -12 V |
|---|---|---|---|---|---|---|
| | System Slot 6 A | Each Peripheral Slot 2 A | System Slot 6 A | Each Peripheral Slot 2 A | Each Slot 0.5 A | Each Slot 0.25 A |
| Total Required Current | 6 A | 6 A | 2 A | 2 A | 0.5 A | 0.5 A |

**PXIe** – The Low Power specification for PXIe chassis is not contained in a table. Rather there are a set of rules and permissions laid out for the manufacturers to follow. Here is the text from section 4.11.2.2 of the PXI Express Hardware specification 1.1

> "A chassis designed for portable application or one with a DC power input may be constrained by the battery and operating hours. This may make meeting the minimum power requirements listed for a standard PXIe Chassis impractical, but minimum power requirements for this class of Chassis are still important for interoperability with PXIe Modules. The minimum power requirements for low-power Chassis are set to allow at least one PXI Express System Module that requires no expansion slots and two PXI Express Peripheral Modules to work in the Chassis, regardless of 3U/6U or the number of slots available".

# 5 CONTROLLERS

The PXI Hardware Specification requires all PXI(e) chassis to dedicate the leftmost slot of the backplane (slot 1) to a system controller. A PXI(e) system has sufficient flexibility to enable it to be configured for either an embedded controller in the chassis or remote control from an external computer. The choice of PXI(e) controller will very much depend upon the application for which the system is intended. This section will look at some of the choices available on the market as well as questions to ask.

## PXI(e) Embedded Controllers

PXI(e) embedded controllers, as shown in Fig. 5.1, are in effect complete miniature PCs in PXI(e) module format and provide compact, high-performance in-chassis control solutions for PXI(e) systems. They can be from 1 to 4 slots wide, with multi-slot modules accommodated by the mechanical space to the left of chassis slot 1, and include standard PC features such as integrated CPU, hard drive, memory, Ethernet, video, serial, USB, and other interfaces, with operating systems including Windows, Linux and Real-Time. Both PXI and PXIe versions are available, but there are now far fewer PXI models on the market. An alternative for a PXI-1 chassis is to use a cPCI embedded controller, which is compatible by virtue of PXI's cPCI heritage, but none of the PXI-specific features such as timing, triggering and synchronization will be available. However, this may not be an issue for more basic T&M applications such as switching and static digital or analog I/O.

All embedded controllers have extended temperature, shock, and vibration specifications and are optimized for the accurate synchronization and high-speed data transfers required by many test and measurement applications. They enable a complete PXI(e) system to be contained within a single chassis and are thus ideal for high performance or very compact/portable applications or operation where external computers are not permitted.

**Fig. 5.1 -** *PXI and PXIe Embedded Controllers*

## PXI(e) Remote Control

PXI(e) remote control solutions offer the performance and benefits of the PXI platform while enabling chassis control through a range of standard computers. This control provides a software transparent connection between a remote computer, such as a desktop/rack-mount PC or server, and the PXI(e) chassis and instruments. It requires a PCIe remote controller card in the host computer (there are now no longer any legacy PCI-based remote controller cards available) and a PXI or PXIe remote control module in slot 1 of the chassis, both connected by a suitable copper cable (fiber optic links are now no longer available). Figs. 5.2 and Fig. 5.3 show PCIe to PXI, and PCIe to PXIe remote controller kits, respectively. This then allows the controller to seamlessly establish a direct PCIe connection to the chassis, and the PXI(e) modules in the connected chassis can then be used as if they are PCI(e) devices directly installed in the host computer. The proviso here is that the chassis must be powered up first, and then when the controlling computer is booted up, it will enumerate all peripheral modules in the PXI(e) system as if they are its own internal PCI(e) boards, allowing straightforward interaction with these devices. Standard computers used to remotely control a PXI(e) test system are readily available and easy to upgrade, and this option is often the most flexible and cost-effective control solution.

**Fig. 5.2 -** *Pickering PCIe to PXI Remote Controller Kit*

**Fig. 5.3 -** *Pickering PCIe to PXIe Remote Controller Kit*

## Laptop Control of PXIe

For many applications a laptop style PC is a more convenient control option by virtue of having an integrated display, keyboard and other elements of the computer within a single compact package. A PXI Express system may be controlled directly from a laptop computer, as shown in Fig. 5.4, by using a PXIe remote control Thunderbolt interface module fitted in slot 1 of the system chassis, connected to the laptop's or a Thunderbolt-enabled mini desktop's Thunderbolt 3 port via a standard Thunderbolt 3 USB Type-C cable.

**Fig. 5.4 -** *Laptop with Thunderbolt Port*

## Controller Selection Considerations

PXI-1 embedded and remote controllers are relatively straightforward to select as 33 MHz 32-bit PXI-1 chassis have a fixed maximum data transfer rate of 132 MB/s, and all compatible controllers will support this. With PXIe chassis however, the data bandwidth depends on its PCIe technology, and can currently range from Gen 1, x4 lanes at 1 GB/s to Gen 3, x16 lanes at 16 GB/s, and Gen 4, x16 lanes at 32 GB/s. Therefore, it is important to select a PXIe controller/chassis combination with matching technologies.

A word of caution - PXI(e) controllers and chassis sourced from the same vendor will be fully compatible as the vendor will have thoroughly tested them together. However, when sourcing controllers and chassis from different vendors, particularly PXIe products, it is recommended to double-check their compatibility – vendors will usually be able to supply a list of other suppliers' products that they have tested for compatibility.

## Multi-chassis Configurations

Multi-chassis configurations allow two or more PXI(e) chassis to be managed by a single remote or embedded PXIe master controller.

The most common method of connecting multiple chassis together is daisy chaining, as shown in Fig. 5.5. This consists of one or more slave chassis connected in series to a master chassis via dedicated expansion modules, with a remote- control interface module in the system controller slot 1 of each slave and an expansion interface module in a peripheral slot of every chassis. Expansion modules are available that can daisy chain up to eight chassis to a single master controller, with each chassis controllable by the host machine.

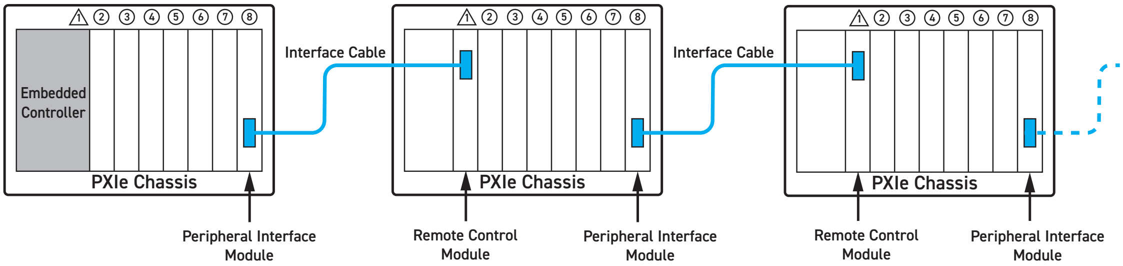



**Fig. 5.5 -** *Control of additional PXIe Chassis - Peripheral Interface Module Daisy-chained to Remote Control Interface Module in the Next Chassis*

While this daisy-chain solution requires expansion interface modules to be fitted into a peripheral slot of each slave chassis, some PXI Remote Control Modules contain built-in daisy-chaining capability through the inclusion of two ports—one for the master connection and one for a slave connection, as shown in Fig. 5.6.

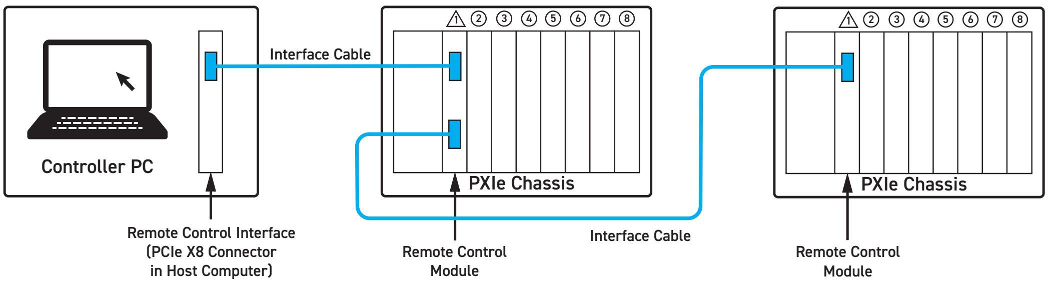



**Fig. 5.6 -** *Control of two PXIe Chassis - Control Interface in First Chassis Daisy-chained to Remote Control Interface Module in Second Chassis*

Alternatively, some PCIe remote controller cards contain two control ports, allowing two chassis to be connected in parallel, as shown in Fig. 5.7. Each chassis can then communicate directly to the host computer rather than through an intermediary chassis.

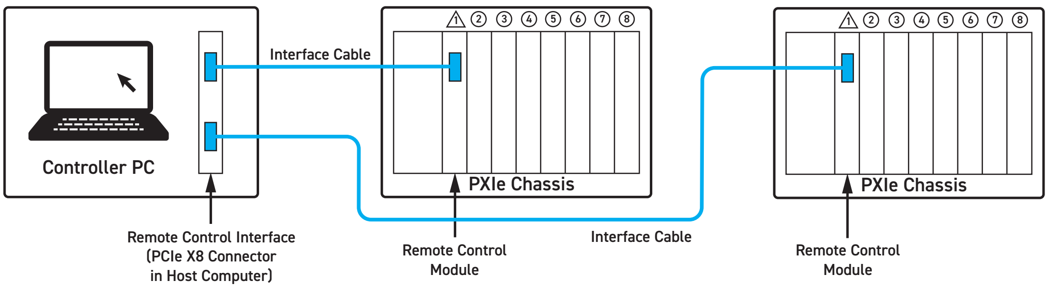



**Fig. 5.7 -** *Alternative Control of two PXIe Chassis - Remote Control Interface in Controller PC Linked to the Remote Control Interface Modules in the two Chassis*

A similar but potentially higher performance solution for the parallel control of two PXIe chassis is to fit two PCIe remote controller cards into the host PC, each controlling a single PXIe chassis.

# 6 SOFTWARE

## Introduction

The PXI standard is reliant on a standardized software and hardware environment. PXI modules have no front panel controls and rely entirely on software control via the PXI backplane.

PXI modules appear on the PCI bus of the controller and so installation of a PXI module is almost identical to that of a PCI card. In the case of PXIe, the situation is similar – installation of a PXIe module is almost identical to that of a PCIe card.

### Operating Systems Supported

The PXI standard requires that PXI(e) modules must support 32-bit Windows or 64-bit Windows, commonly both are supported.

It can be assumed that versions of Windows supported by Microsoft will be supported, although there may be a lag between the release of a new Windows version and the availability of drivers.

At the time of writing most vendors will provide support for:

- Windows 8
- Windows 10
- Windows 11

Support for earlier versions of Windows may also be available, but since these are no longer fully maintained and supported by Microsoft, it cannot be assumed they will be provided. Note that support for Windows 7 ceased in January of 2020, and Windows 8 support ended in January of 2016.

As operating systems evolve, some compatibility problems may occur. For example, starting with Windows 8, any hardware installed requires signed drivers, whereas Windows XP did not, so a driver developed for Windows XP may not install on Windows 8 through 11. Always check with the hardware vendor that the operating system to be used is fully supported.

Also consider that most 32-bit drivers will work on a 64-bit system, the use of 64-bit Windows does not necessarily dictate the use of 64-bit drivers.

### Other Operating Systems

Requirements for 32-bit and 64-bit Linux are included in the PXI-6 PXI Express Software Specification, but support is optional. In addition to Linux variants, other operating systems may also be supported, but this is not a requirement of the PXI standard. If the user is planning to use any other system, checks must be made with the hardware vendors for availability of software support.

As there are many variants of Linux, there is not one driver package that will work in all cases. Fortunately, several PXISA members offer a ‘universal’ driver for control of PXI/PXIe/PCI cards that will work with the vast majority of Linux distributions without the need for compilation for a specific target. This is achieved by using udev and mapping the cards into user space, removing the need for a kernel object compiled for the particular kernel in use. Udev was made part of the standard Linux kernel somewhere about kernel version 2.6 and so this driver should work with any 2.6 or later kernel. Check with your PXI vendors to see if they support a universal driver.

# Software

To successfully operate a PXI platform the operating system must be able to connect to the PXI(e) bus and driver software must be available to support that operating system.

### Real Time OS

LabVIEW Real Time requires the use of a VISA driver and therefore most PXI modules should operate correctly using the Windows driver. However, check with your vendor to make sure.

Installation of Real-time OS like LabVIEW Real Time, Linux-RT or QNX (for embedded systems) may involve transferring a number of files to the target system using either the ftp tool provided or almost any ftp client application. Contact your software vendor to see if they are compliant.

### Development Environments Supported

The PXI specification also recommends, but does not endorse, that several Development Environments be supported. Here are some examples:

- LabVIEW (NI/Emerson Test & Measurement)
- LabWindows/CVI (NI/Emerson Test & Measurement)
- ATEasy (Geotest-Marvin Test Systems Inc.)
- Visual Basic (Microsoft)
- Visual C/C++ (Microsoft)
- Python (Python Software Foundation)
- MATLAB (MathWorks)

However, none of these are mandatory, so check with the vendor. Most vendors also provide support for Visual C# (Microsoft).

### Register Level Interface

Where no driver is available for the OS chosen for the test system, it may be possible to control many PXI cards using low-level register level control. This approach requires that the programmer has detailed information of the hardware and control techniques, therefore can only be considered if the PXI module vendor is willing to provide this level of detail.

This route to module control is not recommended except in exceptional circumstances, it is likely to require a great deal of assistance from the module vendor.

Check with the vendor before embarking on this approach.

## Driver Model

On most operating systems, including Windows, the user cannot interact directly with the hardware but must access through a driver designed for the purpose.

The kernel driver provides low-level hardware access in kernel space and exposes an interface in user space. The kernel driver provides only a very basic low-level interface and typically a further module Application Programming Interface (API) builds on the kernel to provide an interface better adapted to the control of each specific module.

More advanced APIs may build on the lower levels to provide increasingly useful interfaces to include further features and enhancements.

An application program may access the hardware module using any of the available APIs, choice will depend on a number of factors such as the programming environment, interchangeability requirements, and even personal choice.

The diagram in Fig. 6.1 shows a typical set of choices available, from low level programming using the kernel driver interface to increasing higher-level APIs that provide progressively better modelling of the functionality of the particular hardware module.

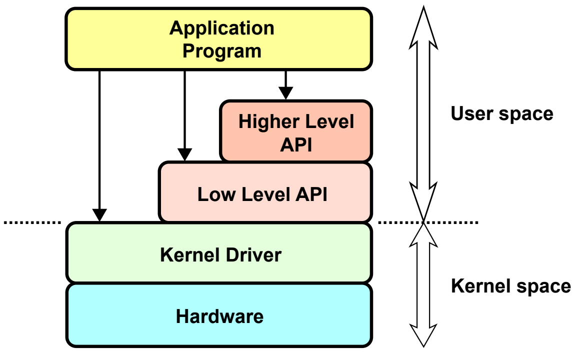



**Fig. 6.1 -** *Programming Layers*

PXI-2 (PXI Software Specification) indicates that "Additional software requirements include the support of standard operating system frameworks such as Microsoft Windows and Linux, and the support of the VISA instrumentation software standard. VISA is a kernel driver providing hardware control plus resource management. The interface is low level, providing only basic input/output functionality for module control. Module control at this level may be very complex and require detailed understanding of the card hardware. Almost all PXI manufacturers will provide a low-level API that encapsulates specialist knowledge of the hardware module to simplify the programming task."

Optionally, an IVI driver is provided by many manufacturers. This is a higher-level API which builds on the lower-level driver and may conform to industry standard functionality for the module type.

In many cases, a non-VISA set of drivers will be available. This is useful in cases where VISA is not available, either due to operating system or licensing limitations. For example, VISA is only available on a limited number of Linux distributions so the user will be forced to use an alternate kernel interface.

## Choice Of Driver

A VISA interface driver is required by the PXI standard, however many PXI(e) modules are provided with a selection of drivers. The user must select the driver most suited to their application and that may also involve an element of personal choice.

Increasingly IVI (Interchangeable Virtual Instrument) drivers are provided. This driver standard is aimed specifically at interchangeability which is discussed in a later section. It may also be required for particular software tools, notably Switch Executive from NI/Emerson Test & Measurement which will only handle modules with an IVI Switch class driver.

In some cases, a user may elect to construct a system without using VISA. In this case, it is essential to consult the hardware vendors to verify if a suitable driver is available.

## VISA

The VISA standard - Virtual Instrument Software Architecture – was originally created by the VXI plug & play system alliance and is now maintained by the IVI Foundation (ivifoundation.org).

The objective of the standard is to define a way of creating instrument drivers with a degree of interoperability between different manufacturers' modules.

The PXI standard encourages the use of the VISA standard. Key aspects of VISA are:

- It allows you to install different drivers from different manufacturers on the same PXI system without conflicts
- It uses a standardized VISA I/O layer for all I/O functions to ensure interoperability
- It defines a way of writing drivers
- A driver that follows the VISA specification uses defined data types and, in some cases, defined function names
- It reduces the process of learning new instruments and the time to develop a test system

## IVI

The IVI standard (Interchangeable Virtual Instrument) is supported by the IVI Foundation (ivifoundation.org). The aim of IVI is to give a degree of interchangeability, instrument simulation and in some cases, higher performance. IVI supports all major platforms including PXI, PXIe, LXI, AXIe and USB. Being a higher-level interface, which often uses a lower-level driver for the hardware interface, its use may result in slightly slower testing speeds compared to other drivers as an extra driver level has been inserted.

### Goals

The stated objectives of the IVI Foundation are to improve hardware interchangeability by:

- Simplifying the task of replacing an instrument with a similar instrument from another vendor
- Preserve application software if instruments become obsolete
- Simplify code re-use from design validation to production

**Improve quality by:**

- Establishing guidelines for driver testing and verification

**Improve interoperability by:**

- Providing an architectural framework that allows users to easily integrate software from multiple vendors
- Providing a standard access to driver capabilities such as range checking and state caching
- Simulating instruments to allow software development when hardware is not available
- Providing consistent instrument control in popular programming environments

As with VISA, IVI is a way to standardize driver development, but it goes much further.

The set of IVI specifications provides a number of instrument class definitions, each class has a standard interface for programming, including function names and data types. By appropriate use of IVI class drivers, a user can develop a system that is hardware independent, meaning instruments may be easily changed for similar instruments from different vendors without the need to re-code the user's application.

At the time of writing the following classes are defined:

- IVI-4.1: IviScope Class Specification
  This specification defines the IVI class for oscilloscopes.
- IVI-4.2: IviDmm Class Specification
  This specification defines the IVI class for digital multimeters.
- IVI-4.3: IviFgen Class Specification
  This specification defines the IVI class for function generators.
- IVI-4.4: IviDCPwr Class Specification
  This specification defines the IVI class for DC power supplies.
- IVI-4.5: IviACPwr Class Specification
  This specification defines the IVI class for AC power sources.
- IVI-4.6: IviSwtch Class Specification
  This specification defines the IVI class for switches.
- IVI-4.7: IviPwrMeter Class Specification
  This specification defines the IVI class for RF power meters.
- IVI-4.8: IviSpecAn Class Specification
  This specification defines the IVI class for spectrum analyzers.
- IVI-4.10: IviRFSigGen Class Specification
  This specification defines the IVI class for RF signal generators.
- IVI-4.12: IviCounter Class Specification
  This specification defines the IVI class for counter timers.
- IVI-4.13: IviDownconverter Class Specification
  This specification defines the IVI class for frequency downconverters.
- IVI-4.14: IviUpconverter Class Specification
  This specification defines the IVI class for frequency upconverters.
- IVI-4.15: IviDigitizer Class Specification
  This specification defines the IVI class for frequency digitizers.

It is important to remember that the class definition cannot include any vendor-specific features, it contains only the basic functionality of the instrument type. It also cannot consider differences in performance, such as accuracy or speed. In practice, it is essential that consideration be given to the consequences of changing from one manufacturer's module to another since those modules may not behave in the same way.

IVI drivers have built-in simulation capability. With this simulation feature it is possible to develop an application without the instrument being present which means software development may start before instruments are delivered, or while being used in another application.

### IVI Driver Architecture

An IVI Driver, shown in Fig. 6.2, is a driver that implements the inherent capabilities defined in the IVI-3.2 Inherent Capabilities Specification document, regardless of whether it complies with a class specification.

An IVI Class Driver is a generic abstract class defining the basic features of instruments of that class as agreed by the IVI Foundation members. An IVI Specific Driver contains features specific to a particular vendor which may not be applicable to modules from other vendors. IVI Specific Drivers may be further sub-defined as an IVI Class-Compliant Specific Driver or as an IVI Custom Specific Driver. An IVI Class-Compliant Specific Driver provides both the Class functionality and additional vendor-specific functionality.

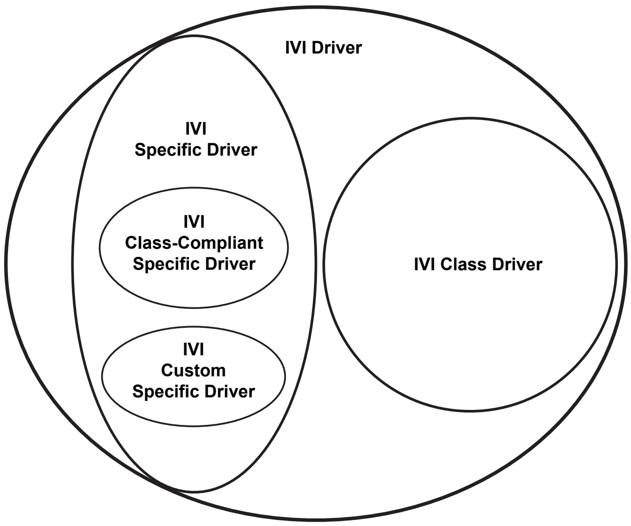



**Fig. 6.2 -** *The IVI Driver*
*(Reproduced from the IVI-3.1 Specification)*

Most specifications contain optional class extension capabilities, such as the Scanner function group in IviSwtch. Being optional, it cannot be assumed that all vendors will provide these capabilities.

Most IVI drivers fall into the IVI Class-Compliant Specific Driver group. This means that the driver is class compliant but adds further functionality beyond the class definition.

| IVI Class Driver | IVI Class Compliant Specific Driver | IVI Custom Specific Driver |
|---|---|---|
| *Inherent Capabilities* | *Inherent Capabilities* | *Inherent Capabilities* |
| *Base Class Capabilities* | *Base Class Capabilities* | |
| *Class Extension Capabilities* | *Class Extension Capabilities* | |
| | *Instrument Specific Capabilities*<br>Functions defined by the manufacturer | *Instrument Specific Capabilities*<br>Functions defined by the manufacturer |

Further to the above, drivers may be provided with a C interface, COM interface or .NET interface.

Most development environments are capable of interfacing to a C interface driver, and many to a COM interface, whereas the .NET interface has a more restricted range of environments.

As Pickering's IVI driver is a class specific (using IVI 4.6 IviSwtch) for switching, any examples presented here relate to signal switching. Other vendor's IVI class specific drivers work on a similar process.

## Example of Interchangeable Switch Modules

The IVI Switch Class driver is the key to interchangeability of switching in a test system, using this driver allows differences between software implementations from different vendors to be moved from the user application and dealt with by the IVI software system.

In the example shown in Fig. 6.3, a pair of changeover relays are used to connect one of two devices under test (DUT) to a signal generator and a spectrum analyzer. For this application a coaxial RF switch is required - NI/Emerson Test & Measurement's PXI-2599 and Pickering's 40-780-022 are both suitable for this application, however they use different drivers and have different naming conventions for the switch channel names as shown in the diagram.

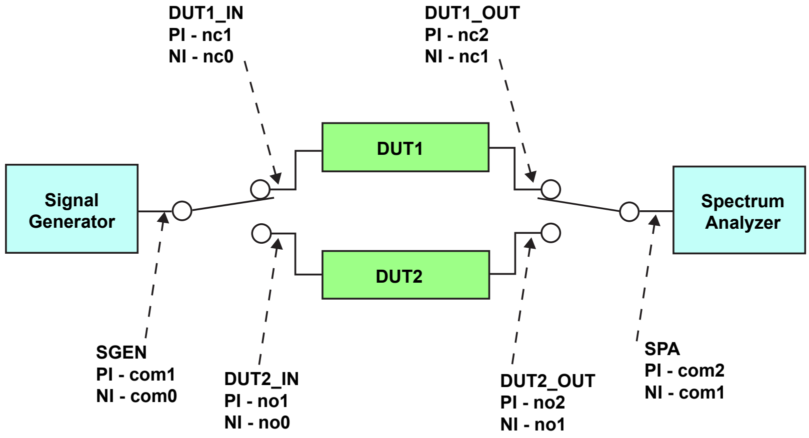



**Fig. 6.3 -** *Changeover relays connecting devices under test*

The first step toward interchangeability is to define virtual names for the channel names; these virtual names will be employed in the user application. Fig. 6.4 provides screen shots from NI MAX showing the Virtual Name tables for the NI-2599 and the Pickering 40-780-022 where the differing naming conventions of the two cards are mapped to a common set of Virtual Names.

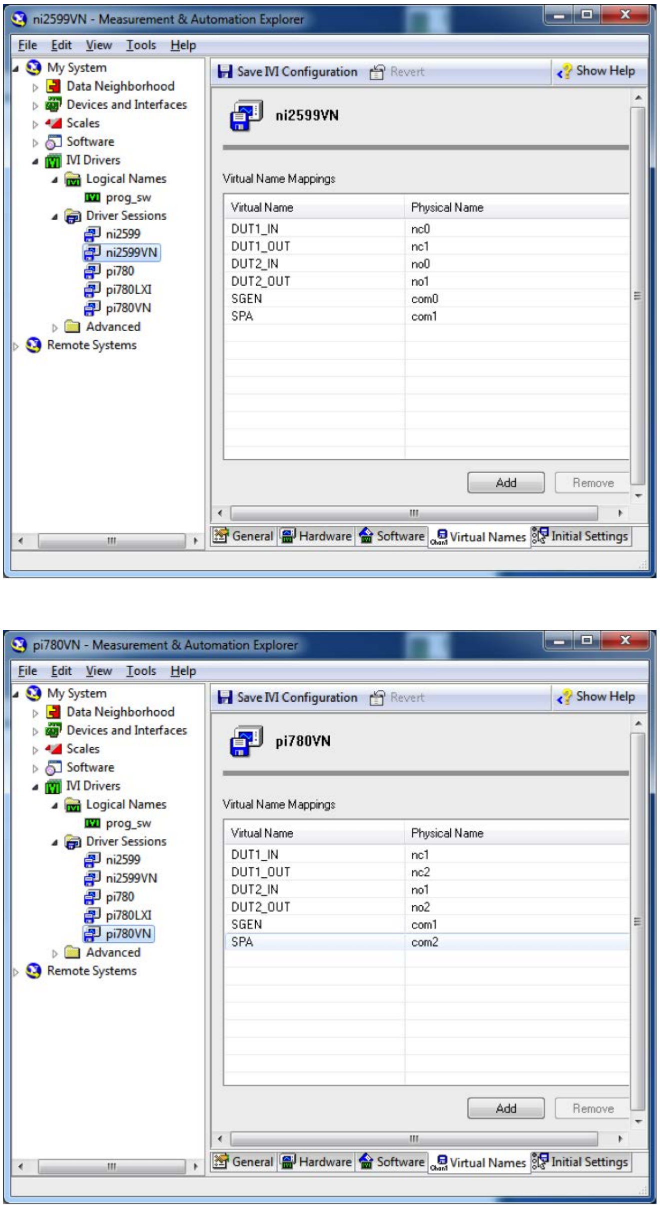



**Fig. 6.4 -** *Defining Virtual Names*

Next a level of indirection is employed to decouple the specific drivers from NI/Emerson Test & Measurement and Pickering Interfaces from the user application. The IVI Configuration Store (provided by the module manufacturer with the IVI Drivers) provides this indirection; it creates the concept of a Logical Name which 'points' to a Driver Session, this linkage may be changed within the store such that a Logical Name may be altered to refer to a different Driver Session. So, if all the differences between the NI and Pickering drivers can be encapsulated in a pair of Driver Sessions, then the Logical Name can be simply modified to refer to either Driver Session. The user then creates an application using the Logical Name. If at some time the alternate module is to be used, then the Logical Name may be changed to refer to the alternate Driver Session. So, the switch module may be replaced with one from a different vendor just by changing the linkage of the Logical Name.

In the screen-shot from MAX shown in Fig. 6.5, the logical name 'prog_sw' is linked to the driver session for the Pickering 40-780-022.

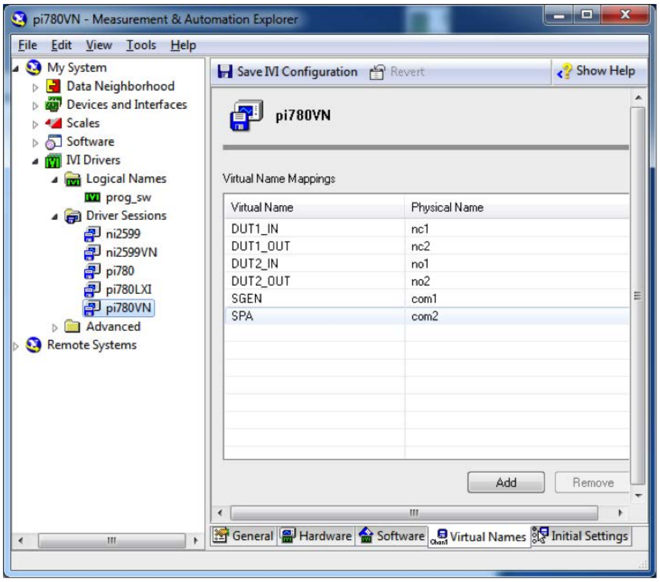



**Fig. 6.5 -** *A screen-shot from NI MAX*

It must be remembered at all times that only differences in software implementations can be interchanged, hardware and performance differences cannot be.

The user application should code using the IVI Swtch Class Driver thus:

```
err = IviSwtch_init("prog_sw", 0, 0, &vi);
err = IviSwtch_Connect(vi, "DUT1_IN", "SGEN");
err = IviSwtch_Connect(vi, "DUT1_OUT", "SPA");
```

This code uses a Logical Name to identify the hardware/software combination and uses Virtual Names to identify the switch terminal channels. It provides completely interchangeable code in that the Logical Name and the Virtual Names may be manipulated in the IVI Configuration Store at any time to permit this code segment to operate different switch modules from different manufacturers without the need to modify the code.

If at some time in the future a new switch product from a different vendor becomes available, all that is required is to create a new Driver Session that defines the driver to be used and the Virtual Name table to define the relationship to the channel names exported by that new driver. The Logical Name may then be modified to link to the new Driver Session and the user application will then use the new module without the need to modify or re-build the application.

For ease of use, instrument driver suppliers can provide an IVI configuration utility to assist users in modifying the IVI configuration store. The IVI Foundation does not define the IVI configuration utility. Therefore, it is likely that multiple IVI configuration utilities will be available.

## Trigger Management

In the original PXI Specifications, the trigger lines were solely controlled by the programmer. This meant that there was a possibility that the programmer might accidentally connect more than one trigger source or instrument on a single trigger line. This could not only cause false triggering but also potentially damage one or more instruments.

In 2012, PXISA released a software specification (PXI-9_Trigger Management – Updated 2018) creating a PXI Trigger Manager DLL. The PXI Trigger Manager is responsible for:

- Providing reservation functionality for each trigger bus line in the chassis for which it is registered
- Providing routing functionality between trigger bus lines, where such routing functionality is supported by chassis hardware
- Enforcing the specific routing capabilities of a chassis by returning well-defined error codes when unsupported or invalid requests are made
- Facilitating the guarantee of exclusive access to drive each individual trigger for a single client at any point in time

It is possible that your version of the PXI drivers does not have the DLL required for Trigger Management. If it is included, it will likely be found at C:\Program Files\PXI Vendor\TriggerManager1.dll. If you cannot find the DLL, contact your vendor. The PXI-9 specification can be downloaded at pxisa.org. Also, you can find applications on how to use IVI drivers at pickeringtest.com/kb/software-topics/ivi-driver-information-downloads.

### Example Trigger Management Code

Set and clear the reservation of line 2 on bus 3 of chassis 4.

```
tPXISA_Status status;  // Open a session to the desired chassis.

tPXISA_Session session;

status = PXISA_ChassisTrig_OpenChassis (

     4,

     “PXISA_CLIENT_2”,
```

```
&session
);

// Make the reservation
status = PXISA_ChassisTrig_SetReservation (
     session,
     3, // bus
     2, // line,
     kPXISA_Trig_Reserved
     );

     ////////////////////////////////////////////////
     // Client-specific code that uses the reserved
     // line goes here.
     ////////////////////////////////////////////////

// Unreserve the line
status = PXISA_ChassisTrig_SetReservation (
     session,
     3, // bus
     2, // line
     kPXISA_Trig_NotReserved \
     );

PXISA_ChassisTrig_CloseChassis (
session
);
```

# 7 BUILDING A PXI TEST SYSTEM

If you have gotten this far in the PXImate, you are probably getting ready to build your first PXI system. In this section, there are a few tips that may help. Our Customer Service team has decades of experience in deploying and sustaining customers' PXI test systems. Let their experiences help you make your integration tasks easier.

## PXI Grounding

Perhaps it is best to first talk about grounding in the PXI environment. Slots within a PXI(e) chassis are grounded through the front panel and return lines provided for power supply voltages. The only ground available at the front panel interface is chassis ground and is established by the front panel being correctly secured to the chassis. Fig. 7.1 is a diagram from the PXI Hardware Specification showing the power and grounding scheme used in a PXI chassis.

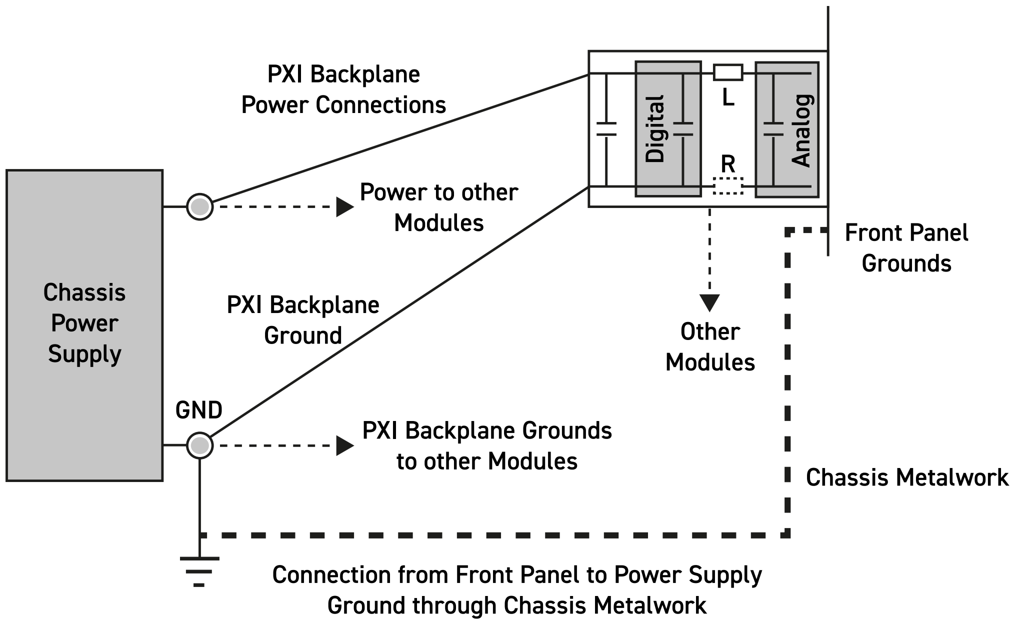



**Fig. 7.1 -** *Power and Ground Connections on a PXI Module*

A basic concept to keep in mind is that any applied source must be returned to this source. Care needs to be taken as to how this is returned.

In the case of Pickering PXI(e) switch modules, the switching is not connected to ground except for coaxial microwave and RF offerings. Microwave and RF cards have coaxial shields connected to the chassis, grounded through the front panel. Some cards have a ground connection available on the front panel connector. This is provided as a convenient chassis ground reference point. This signal name may be GND, FP_ GND or SIG_GND and isn't available on all switch cards. Note that other PXISA members switching modules may be different in their grounding schemes.

Another topic related to grounding is the use of screened (also called shielded) relays, optionally available on some of our switch offerings. Check out our Knowledgebase article "[Managing Screen Connections on Switch Modules]" to learn more.

## Power Sequencing

As you assemble your test system, you need to be aware of how and when to turn on each device in the system. There are several reasons for this. First, the initial inrush current of several large instruments and power supplies may be too great to power up simultaneously. Check the inrush specification of your devices to see if this will be an issue.

Secondly, and probably more important in a PXI-based Test system, the PXI(e) chassis needs to power up before the connected controller PC so that the PCI and/or PCIe bus gets enumerated properly. The result of an improper powerup sequence is that many PXI(e) devices will not be discovered and therefore will not be available to the test program. This issue is usually in the case of an external PC - an embedded PC is not an issue as it powers up with the modules. However, if the PXI(e) Chassis with the embedded PC controls other PXI(e) chassis in a star or daisy-chained configuration, they must all be powered up first.

While you could give a test system operator a power up procedure, it is highly recommended that you consider an AC Management Switch, also known as a Power Sequencer (an example is shown in Fig. 7.2). Typically, they will have six or more switched AC outlets of various current capabilities in the range of 110 VAC or 220 VAC. Manual control or software allows the test engineer to program the sequence of powering up the entire test system and any delay between each power connection.

**Fig. 7.2 -** *Pickering Power Sequencer*

## Mass Interconnect

In any discussion on cabling a DUT to a test system, “Mass Interconnect” needs to be considered. This is a mechanical interface that significantly simplifies connection from power supplies, instruments and switching to a DUT in a test environment and is particularly effective when a range of DUTs need to be tested on the same system. Fig. 7.3 and Fig. 7.4 show a test receiver connected to the front of modular instrumentation, in this case, a PXI chassis. These connections can be via short cables or wireless PCBs. An Interface Test Adapter (ITA) has a series of mating connectors that plug onto the Mass Interconnect receiver. These connectors are wired either to test probes, a series of connectors that mate to the DUT or to a connector where a DUT PCB is inserted for testing.

**Fig. 7.3 -** *VPC Cabled Mass Interconnect*

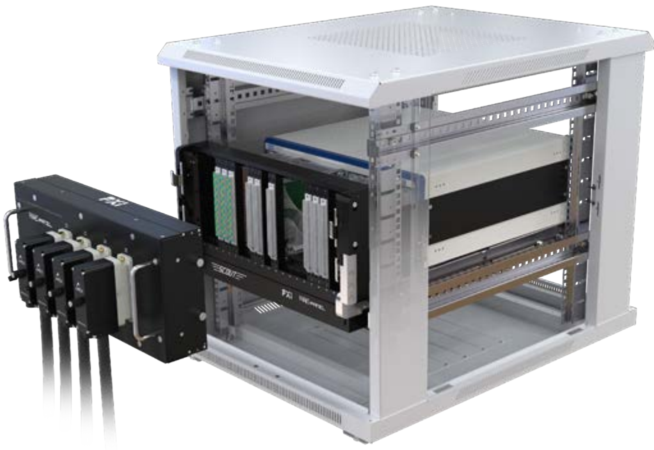


**Fig. 7.4 -** *MAC Panel Short Wire/Wireless Mass Interconnect*

The Mass Interconnect is clearly an additional piece of hardware to order, assemble, and integrate, so there will be added costs. To justify the expense, there are questions to be asked and considered, plus several design considerations:

1. **What is the overall DUT volume versus DUT type mix?** If your test system will be testing many different DUT types, which may require many ITA changes, a Mass Interconnect will make a lot of sense. If you are testing only one or two DUTs and ITA changes are usually not necessary, you probably don't need to use a Mass Interconnect, just cabling and connectors for manual insertion.
2. **How much down time can you justify for your test system in a day?** In a high-volume test strategy, ITA changeover times will be shorter with a Mass Interconnect approach. Also, the possibility of connecting the cables to the wrong mating connectors is eliminated or reduced.

3. **What are your accuracy requirements?** Mostly due to the robustness of the connections to the ITA, a Mass Interconnect will help ensure signal repeatability when changing over ITAs. At higher frequencies, signal fidelity will be improved compared to using loose cables to connect the ITA because the connection is shorter and more rigid in terms of its placement in the ITA and Mass Interconnect.

   If you plan to replicate the test system multiple times, a Mass Interconnect helps to guarantee repeatability across all systems.

4. **Is ruggedness a requirement?** Mass Interconnects are very robust and may make sense if your tester is in an environment where damage to the test system is possible.

5. **Will a self-test fixture be required for the system?** A Mass Interconnect will make the use of a self-test fixture easier and more repeatable.

6. **Are you migrating an older test system (e.g., VXI-based) to a new test platform such as USB or PXI?** If your present test system already has a Mass Interconnect, then the migration is made easier because you can reuse the ITAs. The Mass Interconnect just needs to be cabled in such a way that the test resources are connected to the same connector pinmap as the previous design.

### Choices

If you have concluded that you need Mass Interconnect, you need to select a vendor. Companies that manufacture Mass Interconnect systems include VPC and MacPanel, and both companies can support an extremely wide range of different vendors' PXI(e) modules with both wired and wireless receiver solutions. For ITA selection, the provider of your Mass Interconnect can help. In addition, many systems integrators and test fixture designers/ manufacturers are located in many cities around the world.

Connector types have very well-defined specifications, and they should be observed. Connectors are rated for max voltage, current, power, and frequency. A connector rated at 100 V and 10 A is not necessarily a 1,000 W connector – It maybe be up to 100 V or up to 10 A and not both. Be sure you understand your needs before specifying connectors.

With the above information, your vendor(s) can help select the size of the receiver and ITAs and configure the proper pinouts, receiver connectors, and the wiring required.

## Simulation

There will come a time when you are ready to write your test program and... potentially, a supply chain issue means that a key device will not be available for several weeks. You could just write your test program without having access to any hardware, but it is difficult to know if you are talking to the right instrument, setting logical parameters and what switch is closed. But fortunately, there are simulation options in many instances.

### IVI

If you go back to the Software section of this book, look closer at IVI (Interchangeable Virtual Instrumentation). IVI Drivers are primarily aimed to give a degree of interchangeability with other manufacturers of similar instruments. But IVI also supports Simulation during test program development so you can continue development and verification of your program while waiting for the actual device.

IVI Simulation provides the following:

- Instrument Calls – The test engineer can make instrument calls to the driver, which will not try to communicate with the instrument, but will return a valid instrument handle so you can pass it to other functions in the driver.
- Range Checking – Valid limits for the instrument simulated and the IVI driver returns valid response from the virtual instrument as well as errors when the command features parameters that are not supported by the real instrument.
- Simulated Data and errors – The drivers have built in simulated data generation algorithms. The test program in simulation mode returns to the program realistic data. You can also set up the IVI driver to return simulated errors so you can test the programs flow chart to see if reacts properly to bad data.

## Simulation Tools

For PXI(e) modules manufactured by Pickering Interfaces, we provide simulation software as part of our PXI driver installation that allows you to develop the test system software independently from your application hardware, thus minimizing the time required with the physical system. Our PXI modules may also be used in our range of LXI compliant, Ethernet controlled modular chassis, in which case a complete chassis and modules may be simulated by an LXI simulator tool (available in both hardware and software versions) in lieu of the actual hardware. An LXI Simulation Tool is shown in Fig. 7.5, and an LXI Debug Utility in Fig. 7.6.


**Fig. 7.5 -** *Pickering LXI Simulation Tool*

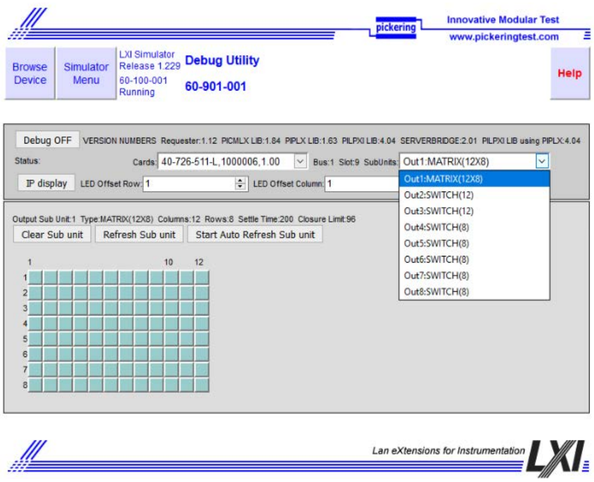



**Fig. 7.6 -** *Pickering LXI Simulator Debug Utility*

The advantage here is that our PXI(e) switch module Soft Front Panels are synchronized with the simulator. So, when you virtually close a relay, you don't just get a response in software, you can see what relay or relays would have been closed in your test program, creating further confidence in your software efforts.

Other PXISA Members may have similar tools for program generation. Check with your switching and simulation vendor to be sure.

## Cooling

If you are integrating your test system in, for example, one or more 2 m racks, the subject of cooling comes into play. All the components in the test system add to the amount of heat generated by the overall system. In a complex rack test system, you are likely to have one or more PXI chassis, several AC and/or DC sources and possibly some complex instrumentation too large to fit in a PXI chassis. So, consideration of the overall thermal footprint is needed.

- Individual Instruments and Power Supplies – Each stand-alone instrument and power supply exhausts heat into the rack cabinet. All this heat must be removed to keep the interior of the cabinet cool. The wattage rating should be available in each datasheet. Note where the heat is exhausted - if it is at the rear of the device, then the cabinet cooling must address this. Some instruments may exhaust out the front, eliminating most heat issues inside the cabinet. It should be noted that devices with either cooling scheme may be stacked directly on top of each other in the rack cabinet.

  There are also instruments and power supplies that use the top and bottom surfaces for air intake and exhaust. This design requires a specified number of rack Us (Unit of measurement in rack mounts - 1U equals 1.75 Inches or 44.45 mm) space between the adjacent rack mount devices to facilitate proper cooling.

- PXI(e) Chassis – Calculating the heat generated by a PXI(e) chassis is tricky. That is because it depends on the number of slots populated with Modules and their individual power requirements. You can use the maximum power rating of the chassis power supply as a worst-case scenario. The cooling scheme comes to play here much like the instruments and power sources above.

  In terms of rack placement, a chassis that takes in air from the front and out the back may be stacked in a similar fashion to the scenario I mentioned above. However, PXI(e) chassis that intake at the bottom need at least 1U of empty rack space between any device stacked underneath.

- Cables – In most cases, engineers do not expect that the heat of a cable would be a major factor. But suppose the test system has a cable rated at 5 A and with 50 wires. Its maximum surface temperature is 100 °C and the ambient is 25 °C, so the cable can have a maximum rise of 75 °C. The data sheet indicates that the temperature rise under load increases by 0.08 °C per square of current (Amps). Simple arithmetic then suggests the maximum square load current should be 937 A. A wire carrying 5 A has a square load current of 25, so 37 wires could be carrying 5 A and the temperature will rise to a little less than 100 °C. If the cable had an operating temperature which is higher, then the maximum load would be more restricted. If all wires were carrying the same current (an unlikely scenario) then the square of the current in each of the 50 wires should be 18.7, or 4.3 A.

  This is probably a worst-case scenario, but it is possible in very high-power test applications... and a cable dissipating heat at near 100 °C is a danger to the operator if any of the cable is external to the rack and you still have the issue of dissipating that heat if it is inside the rack. So, look at your test program and ask these questions;

  1. What is the max current per channel?
  2. How many wires will be carrying current at any one time?

These two questions and some simple maths will alert you to a possible problem at test system deployment.

- "In with the cold air, out with the hot." – Moving heat out of a test system rack can be serious business. The heat can possibly damage or at least affect the accuracy of sensitive instruments. As mentioned above, if temperatures get very high, you could have a danger to people maintaining the system.

  In this case, first you need to know what is expected from Product Engineering in terms of cooling. Is an intake filter and exhaust port adequate for cooling? How big do the filter and exhaust ports need to be for adequate cooling? What is the maintenance schedule for filter cleaning/replacement? Where should the rack exhaust the heat - not aimed at the operator unless you are testing in Alaska! Does the environment where the test system is operated indicate that a sealed test rack is required? An example could be a shipboard test system where salt air is corrosive. If so, is an AC unit required?

## Microwave Cabling

Standard AC/DC cable selection is driven by Ohm's law, cable lengths (think capacitance) and the temperature of the environment. In the case of RF and Microwave cabling, there are far more parameters to consider.

The coaxial (also called "coax" for short) cable selected can affect insertion loss, distort the signal because of the added capacitance, and limit the power to be handled.

If you choose the correct cable for your application, there should hopefully be minimal surprises. However, you should not make a bend smaller than the recommended radius, expose the cable to excess temperatures, vibration, mechanical stress, or chemicals. Be certain to attach the coaxial cable into properly designed connectors, paying careful attention to insulation and dielectric strip lengths, solder temperatures and dwell times and shielding preparation.

Questions to ask when designing an RF Cable "channel" (a signal path from the instrument to the DUT and a return path) for a given stimulus and/or measurement:s

- Insertion Loss – The switching and the entire path of the channel must be considered. The coax wire type, cable length and the number of connectors involved, as well as the insertion loss of the switching module all contribute to the quality of the channel. When using a Mass Interconnect, you automatically add three more insertion losses – the connector from the instrument cable, the Mass Interconnect's interface, and the connector from the DUT cable to the Mass Interconnect.
- Performance – Will coax with a simple braid shield be adequate or do you need to invest in more than two shield layers, such as double shielding or even "quad-shield", which uses four alternating layers of foil and braid? The best performance will be a "semi-rigid" coax. As the name implies, the cable is not as flexible as most coax types. With a solid copper outer shield, semi-rigid cable must be carefully bent so as not to kink the shield, resulting in signal losses.
- Connector Types – The instruments you choose will have a particular connector type to interface to the DUT. These connectors will usually fit to a limited number of coax types. So, selecting the coax must be tempered with the required connector.
- Impedance – To get a clean, repeatable signal, the signal path must match the impedance of the DUT as well as the instruments used.

# 8 THE PICKERING GROUP AND PXI PRODUCT OVERVIEW

## Pickering

The Pickering Group of companies has been in switching technology since 1968 when our reed relay division, Pickering Electronics, introduced its first instrumentation grade reed relays. The Group also comprises Pickering Interfaces—the switching and simulation division and Pickering Connect—the connection division, with manufacturing in quality accredited factories in the UK and the Czech Republic (EU). We have direct sales and support offices, representation and system integration partners throughout Asia, Europe and North America. Together we deliver high-quality switching and simulation products and a full range of standard and custom cable and connector solutions. Our employees share a customer-centric approach and are dedicated to quickly getting our products functioning at their peak and into our customers' hands.

Pickering Electronics is well known for leading the high-end reed relay market through innovative product design, high-performance components and exceptional quality control. Today, Pickering Electronics' Single-in-Line (SIL/SIP) range is by far the most developed in the relay industry with devices 25% the size of many competitors. These small SIL/SIP reed relays are sold in high volumes for demanding Aerospace, Infrastructure, Semiconductor Testing, Test & Measurement, and ATE applications worldwide.

Pickering Interfaces was formed in 1988 and provides modular signal switching, simulation, software and services to streamline the design, deployment and sustainment of high-performance electronic test and verification systems. We provide the most extensive range of switching and simulation solutions in the industry for PXI, LXI, USB and PCI applications. To support our switching and simulation solutions, we also offer application software and software drivers along with a full range of supporting connectivity and cabling solutions supplied via our Pickering Connect division.

Please see the following pages for further information on our extensive range of PXI products..


**Fig. 8.1** - *The Pickering Facility in Bystřice, Czech Republic*

# Product Overview

## **PXI** & **PXIe** Switching

- General Purpose up to 9kV & 40 A
- Matrices
- Multiplexers
- RF & Microwave up to 110GHz
- Fault Insertion
- Optical
- Differential & Telecoms
- Scalable Switch Systems
- 6U Switching

Our range of PXI "BRIC" Scalable Switch Systems provide large matrix and multiplexer solutions that can be easily reconfigured by the addition of sub-cards.

## **PXI** & **PXIe** Simulation

- Programmable Resistors for Simulation of Resistive Sensors including:
    - Temperature
    - Strain
    - Pressure
    - Load/Force
    - Moisture
    - Light
- Battery Simulators
- Thermocouple Simulators
- LVDT, RVDT, Resolver Simulator
- 4-20 mA Current Loop Simulator
- Power Supplies
- Digital I/O & Prototyping
- Waveform Generation
- RF and DC Attenuators
- Amplifiers

All of our PXI modules will plug into any PXI chassis or a Hybrid Slot in a PXIe chassis. They can also be hosted in our LXI/USB modular chassis, allowing remote control over a Gigabit Ethernet or USB3 connection.

Our PXIe modules will plug into any compliant PXIe slot or Hybrid Slot in a PXIe chassis.

## **PXI** & **PXIe** Modular Chassis

Our PXI & PXIe chassis will accept any PXI vendor's modules, and include:

- 8, 14, 18 , 19 and 21-slot Versions
- Gen 2 & Gen 3 PXIe Chassis
- Gen 4 1-Slot PXI Embedded Controller
- Remote Control Kits for External PCs

## **LXI/USB** Modular Chassis

These chassis accept our 1000+ PXI Switching and Simulation modules and include:

- 2, 4, 6, 7 and 18-slot Versions
- Scan-list Sequencing/Triggering Provided on Larger Chassis
- Ethernet or USB Control Enables Remote Operation
- Low-cost Control from Practically Any Controller
- LXI Provides Manual Control via Web browsers
- Driverless Software Support
- Power Sequencing Immunity
- Ethernet Provides Chassis/ Controller Voltage Isolation
- Independence from Windows Operating System

## Supporting Cable & Connector Solutions

- Cable Assemblies
- Connectors
- Breakouts
- High Density
- High Voltage
- Power
- RF

## Custom Cable Assemblies

There are times when a standard cable will not do the job adequately. You may need custom cable lengths, specialized connectors & wire types, or complete specialised looms. We are always willing to quote for your specific designs.

## Try our Online Cable Design Tool

With this free online tool, you can design your custom cable assembly by using either our built-in library of standard cable sets or create an assembly from scratch.

Give it a try at
pickeringtest.com/cdt

## Comprehensive Software Solutions

Our commitment to delivering you high-quality products does not just extend to our switching solutions. Our in-house software team has created a range of applications to accelerate test system development and execution. These include Switch Path Manager (SPM) signal routing, Sequence Manager scan-list/triggering, PXI & LXI Simulators and a Microwave Switch Design Tool, along with a comprehensive range of Windows and Linux drivers (available to download free of charge from our website). We have a strong focus on ease-of-use and compatibility. More details at pickeringtest.com/products/software.

## Diagnostic Test Tools for Switching Systems

Verification and diagnosis of complex switching operation in a test system has always been an issue. Our BIRST and eBIRST diagnostic test tools provide you with a quick and straightforward way of finding relay failures within our PXI, PCI and LXI switch systems.

## Long-Term Support

Most test engineers reasonably expect their test systems to last at least as long as the products being tested. We understand this need and pride ourselves on the fact that the design and manufacture of all our switching, simulation, software and connectivity products are done in-house. These capabilities enable us to provide guaranteed long-term support, typically 15 to 20 years, and low obsolescence.

## Resources

To help you select the appropriate product from our extensive catalog, we offer our online Product Selector; take a look. Our products are also supported by detailed data sheets, operating manuals, and access to our in-house team of experienced application engineers. Don't miss out on the available Support Knowledgebase, success stories, white papers, webinars, videos and more at pickeringtest.com/resources.

# 9 USEFUL INFORMATION AND GLOSSARY

This section provides useful information sources for PXI products, organizations and standards.

## PXISA

The PXI Systems Alliance is a not-for-profit organization run by companies involved in PXI products.

The organization is responsible for the production of the technical specifications for PXI and the promotion of the PXI concept in the market.

Membership of the PXISA is open to any company that is willing to promote the PXI standard and manufacture products that support PXI.

### MEMBERSHIP CATEGORIES

#### PXISA

There are three levels of Membership in the PXI Systems Alliance.

Membership of the PXISA allows participation in the PXISA technical and marketing programs and the use of the PXI logo.

Only PXISA members are permitted to use the PXI logo on their products and marketing material. Executive and Sponsor members must also manufacture PXI products and/or provide services.

#### Sponsor Member

Sponsor Members, such as Pickering Interfaces, view PXI as a vital part of their ongoing business and want to be involved in all decisions. To be a Sponsor Member the company must be an Executive Member for at least one year prior to applying for Sponsor Membership and be producing PXI products. Sponsor members have a seat on the Board of Directors and voting rights.

#### Executive Member

Executive Members view the PXI architecture as important to their business. They have a voice in technical specification and approval and Marketing direction of the PXISA. An executive member must also be producing PXI-related products. Executive members have voting rights.

#### Associate Member

Associate Members want to be a member of the Alliance supporting PXI and to be informed of progress. They do not have voting rights and this membership level is for information purposes only.

A membership list, PXI products and membership applications are available on the PXISA website pxisa.org

## Other T&M Organizations

### AXIe Consortium

The AXIe Consortium is responsible for the development and promotion of the open AdvancedTCA Extensions for Instrumentation and Test (AXIe) standard. axiestandard.org/

### IVI Foundation

The IVI Foundation promotes specifications for programming test instruments that simplify interchangeability, provide better performance, and reduce the cost of program development and maintenance. The Foundation has been responsible for managing the VXIplug&play specifications since 2002. ivifoundation.org/

### LXI Consortium

The LXI Consortium is responsible for the creation and promotion of the LXI standard. It has a similar structure to the PXISA since it has different membership levels and working groups to create and maintain the specification. lxistandard.org/

### PICMG

PICMG (PCI Industrial Computer Manufacturers Group) is a consortium of over 600 companies who collaboratively develop open specifications for high performance telecommunications and industrial computing applications, including the CompactPCI (cPCI) and CompactPCI express (cPCIe) standards. CompactPCI products can be used in PXI Chassis and PXI products can be used in cPCI chassis with some limitations in each case. picmg.org/

### PCISIG

The purpose of the PCI Special Interest Group (PCI-SIG) is to deliver a stable, straightforward and compatible standard for PCI devices.

The organization is important to PXI since the control interface is built on the PCI and PCIe standard and uses common electrical devices to PCI cards, including the PCI Bridges incorporated into the PXI backplane. pcisig.com/

### USB

The USB Implementors Forum (USB-IF) promotes the Universal Serial Bus and maintains the specifications and a compliance program. USB 1.1, 2.0 (High Speed), and 3.0 (Super Speed) devices are available. USBTMC is a software interface standard created to support test and measurement applications and is now controlled by the IVI foundation. usb.org/

### VXIbus

The VXI standard is controlled by the VXIbus Consortium. While there have been no new products released in over five years, there is still a large installed base of VXI systems around the globe. vxibus.org/

## Useful Websites

| | |
|---|---|
| IEC Specifications | iec.org |
| IEEE Specifications | ieee.org |
| Pickering Interfaces | pickeringtest.com |
| VME Specifications | vita.com |

## Terminology

| | |
|---|---|
| 3U, 6U | Refers to the height of the module, the 6U modules being approximately twice the height of 3U modules. 1U is 44.45 mm (1.75 inches). |
| API | Application Programming Interface |
| AXIe | AdvancedTCA Extensions for Instrumentation and Test |
| CompactPCI (cPCI) | A ruggedized version of a PCI card conforming to PICMG 2.0 specification, providing superior mechanical performance and easier insertion & removal of cards. |
| GPIB | General Purpose Interface Bus, a standard used for interconnecting bench instruments using an 8 bit wide data system. Standard is defined by IEEE 488. |
| IVI | Interchangeable Virtual Instrument |
| J1, J2, J3 | Connectors on PXI(e) modules that mate to the PXI(e) chassis backplane connectors P1, P2 and P3. On 3U modules connectors J1 & J2 may mate to P1 & P2. |
| Local Bus | Bus that can be used to connect adjacent PXI modules in a chassis without the use of PXI features. The Bus can be used for analog or digital signals and is module defined. A reduced capability version is available for PXIe slots. |
| PCI | Peripheral Component Interconnect, bus system commonly used in computers to provide additional functionality. |
| PCISIG | PCI Special Interest Group |
| PICMG | PCI Industrial Computer Manufacturers Group |
| PXI | PCI eXtensions for Instrumentation |
| PXIe | Peripheral Component Interconnect express, bus system used in computers to provide additional functionality. Replaced parallel bus of PCI devices and provides higher speed multi-lane serial ports. |
| PXI-1 | Original PXI configuration. Includes P1 and P2 on both 3U and 6U PXI |
| PXI-5 | PXI Express Configuration |
| PXI-5 Hybrid | PXI-1 modules with most of J2 removed to allow insertion into a PXI Hybrid slot |
| Star Trigger | Fast trigger system driven from Slot 2 that provides a low latency synchronized trigger to the Peripheral Modules. |
| Trigger Bus | A backplane bus defined in the PXI standard that can be used to trigger events. The triggers can be conditioned by software. |
| USB | Universal Serial Bus |
| VISA | Virtual Instrument Software Architecture |
| VXI | VME Extensions for Instrumentation |
| P1 | Backplane connector on a chassis carrying the 32-bit PCI bus. |
| P2 | Backplane connector on the chassis carrying the 64-bit PCI bus and PXI specific features. |

| | |
|---|---|
| P3 | Reserved connector whose use is undefined by the PXI standard and can be fitted to 6U Chassis. |
| P4 | Connector used on 6U Chassis that allows 3U cards to be fitted in 6U Slots. It provides the same functions as P1. |
| P5 | Connector used on 6U Chassis that allows 3U cards to be fitted in 6U slots. It provides the same functions as P2. |
| System Slot or Slot 1 | Slots on the left-hand side of a PXI(e) Chassis reserved for use by the system controller (embedded or remote). |
| XJ3 | PXI Express connector on both 3U & 6U modules |
| XJ4 | Portion of J2 for compatibility of PXI Hybrid slots and allow the module access to the PXI trigger busses. |

## pickering Direct Sales & Support Offices


**Pickering Interfaces Inc., USA**
Tel: +1 781-897-1710 | e-mail: ussales@pickeringtest.com

**Pickering Interfaces Ltd., UK**
Tel: +44 (0)1255-687900 | e-mail: sales@pickeringtest.com

**Pickering Interfaces Sarl, France**
Tel: +33 9 72 58 77 00 | e-mail: frsales@pickeringtest.com

**Pickering Interfaces GmbH, Germany**
Tel: +49 89 125 953 160 | e-mail: desales@pickeringtest.com

**Pickering Interfaces AB, Sweden**
Tel: +46 340-69 06 69 | e-mail: ndsales@pickeringtest.com

**Pickering Interfaces s.r.o., Czech Republic**
Tel: +420 558 987 613 | e-mail: desales@pickeringtest.com

**Pickering Interfaces, China**
Tel: +86 4008-799-765 | e-mail: chinasales@pickeringtest.com


Local Sales Agents in **Australia, Belgium, Canada, China, India, Indonesia, Israel, Italy, Japan, Malaysia, Netherlands, New Zealand, Philippines, Singapore, South Africa, South Korea, Spain, Taiwan, Thailand, Turkey and Vietnam.**