This paper was presented as a poster session at NSS90.It is not a discussion document

A modular approach to a VXI Resource Manager

P.J.Coleman-Smith
S.E.R.C. Daresbury Lab.,Warrington, England
and
Ch.Ender .
Physics Institute, University of Heidelberg,Germany

Abstract

The EUROGAM collaboration has decided to use 'D' size VXI modules for the detector electronics for the Ge and BGO detectors. This combined with a requirement for control information to be distributed via Ether-net, necessitated the design of a slot0 card for VXI.

The board is modular to accommodate any commercial VME processor and to allow for the specification of the VXI trigger lines.

1.INTRODUCTION

The basis of the detector electronics for the EUROGAM collaboration[1] is the VMEbus eXtensions for Instrumentation specification [1] (VXI). This uses the VMEbus [2] as a multi computer control bus, and adds features to enable it to be a multi-vendor open architecture for instruments.

VXI defines the hardware and software architecture. The features that are of particular interest are as follows :-

- Distributed low skew ECL star lines for high speed Triggers.
- Analog SUM bus for multiplicity.
- Bussed ECL, and TTL trigger lines for synchronous control of detector electronics modules.
- Private inter-module backplane bus for use as ECL in pipelined readout, or inter-module communications, or additional analog sum busses, and inspection lines.
- Larger module sizes, 'D' size : 14.4in x 13.4in, with Electromagnetic compatibility specification, and increased width of 1.2 in.
- Extra power supplies : +/- 24v, -5.2v, -2v
- Autoconfiguration allowing inter change of modules with no jumper changes.

2.FUNCTION.

The module is required to provide the minimum VXI slot0 functionality, a Resource Manager, VME arbitrator and system function source. Further the module must interface the trigger and analogue signals to the VXI backplane as required by the EUROGAM collaboration.

The definition of the connections through the module to the VXI backplane were not fully defined at the stage when the requirement for the module was first discussed, however there was a need for such a module at an early stage of the electronic development for the collaboration to allow the prototypes to be controlled and tested in a VXI environment.

The control communication path throughout the data acquisition system had been chosen to be Ether-net, so the module needed to provide an interface. The Processor chosen had to be capable of running the selected software, in this case VXWorks, and it would be an advantage to have a local disk, hence SCSI. The cost of developing such a board, from scratch, and the time delays involved were considered to be prohibitive.

The solution was to adopt a modular design philosophy and, where possible purchase ready made designs.

3.ARCHITECTURE.

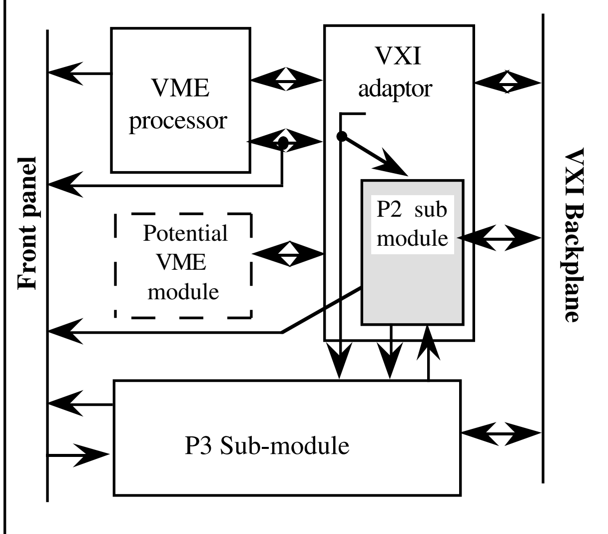


Fig 3.1 Module Architecture

The module is formed by a collection of interconnected sub-modules mounted in a VXI 'D' size module. The

[1] A brief overview and a principle introduction to the new Data Acquisition can be found in [IEEE NS-37 no2 page 326] [5]

individual sub-modules are bolted to the module shielding panels.

a.VXI adaptor

This sub-module extends and buffers the VME bus to a pair of DIN 41612 connectors suitable for a 'B' size commercial processor, and includes a VME Slave interface. The Slave provides the VXI configuration registers, controls for the MODID lines and software access to other sub-units.

There are a number of other connectors on this sub-module as follows:-

- Buffered VME bus .
- Processor P2 connector rows a & c.
- VXI bus P2 connector rows a & c .
- 16 bit data and 16 bit address external bus with controls, from the Slave.

The Slave control logic is implemented using an Altera MAX5128 Erasable Programmable Logic Device (EPLD)[3]. This gives the ability to change the use of the Slave configuration registers with ease.

Those configuration registers that are not currently defined, and some that are reserved, are implemented as RAM giving the ability of software testing of the Slave on power-up.

a.1.External bus operation.

The external bus is implemented as an asynchronous handshake. There is a strobe signal from the Slave to the external bus, to indicate the address is valid, and an acknowledge signal to indicate data is accepted/available in much the same way as the VMEbus is organised.

A write to the data address @26 will enable data onto the external data bus, and cause a strobe to occur. The slave and hence the VMEbus then waits for an acknowledge signal from the external bus before completing the VME cycle. A Read will cause a strobe to occur, the slave will then wait for an acknowledge signal from the external bus before completing the VME cycle.

a.2.Special buffering requirements.

There are within the VMEbus a number of signals which can be considered as asynchronous. These signals are harder to buffer than those whose direction of drive can be predicted using the bus grant, bus busy, and R/W lines. The buffers must be capable of handling signals coming from both sides at the same time. So when the first isno longer driven from its source the second continues to drive the first, but when the second is no longer driven from its source, then both lines are no longer driven. The problem is further complicated by the need to use high current open-collector drivers for the VMEbus side. There are 14 such signals, so any solution must not use too many ICs.

The solution was found in using a programmable device with high current open collector outputs. The device is a PLX448 [4] from PLX technology. See Fig 3.a.2.1 for a circuit. Four of these circuits could be programmed into one device, so a total of 4 are used in the final design.

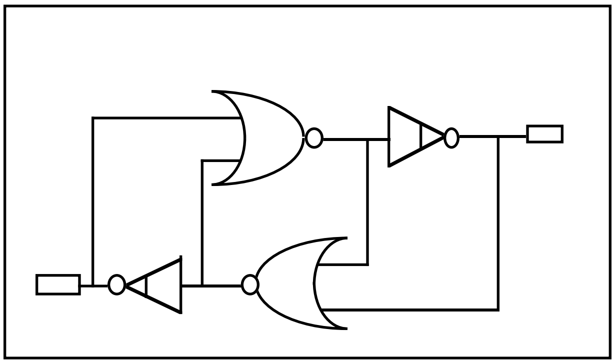


Fig 3.a.2.1 asynchronous buffer circuit

b.VXI bus P2 rows a & c buffers.

This sub-module is connected between the VME buffer sub-module and the VXI P3 sub-module. It is positioned over the VME buffer sub-module as a daughter board, and connects to the VXI P3 sub-module using a flexible IDC connection. It is a means of easily configuring the type of buffering to be used on the LBUSC lines . It is also used to control the TTLTRG and ECLTRG lines. The table 3.b.1 lists the buffering used in the collaborations version.

c.VXI P3.

This sub-module connects to the VXI P3 connector, the 16 bit data and address connector of the VXI adaptor sub-module, and the VXI P2 buffer sub-module. The sub-module provides buffers and drivers for the ECLTRG, TTLTRG, Clocks, and analogue signals from the front panel inputs to the backplane at the P3 connector and via the ribbon cable to the P2 sub-module.

A single signal is buffered to drive all the STARX lines from the front panel to the backplane, using low skew drivers. The STARY lines are monitored and the OR is driven to the front panel.

Hardware is provided to ensure that all drivers of VXI backplane signals will reset to the correct state.

There is a 16 bit ADC for monitoring of DC values from either the front panel or one of the P2 buffered analog lines. This is used in conjunction with multiplexers on the detector electronics modules for remote monitoring. Read out and control of the ADC is via a status/control register on this sub module that is accessed using the Slave external bus.

There is a development program under way at Heidelberg for a Fast ADC daughter board to provide logic and analogue monitoring of detector pulses in a one-shot mode . This will be a daughter board of the P3 sub-module, and will be read out via the Slave external bus by the VME processor. The inputs for this will come from the detector electronics via the backplane.

D.Front Panel.

This carries the connectors for the processor external connectors and external connections to the other sub-units for buffering to the VXI backplane. This means there has to be a specific front panel for each processor chosen. But since the collaboration only intend to use one type for this function it was not considered a problem.

e.Commercial Processor.

The commercial processor must be capable of providing the minimum VME slot 1 functions, and be VMEbus Master with A24/D16, A16/D16, and interrupt handler capability. It will in the case of EUROGAM also provide an Ether-net interface, and using a SCSI bus to interface discs, be a stand alone software environment. There must also be a target for VXWorks available.

The chosen processor in this case is the Motorola MVME147.

TTLTRG0-7	Buffered on and off the backplane. Open collector TTL signals. There are thus 14 signals which pass to the P3 sub-module The backplane drivers are enabled as a block by a control signal from the P3 sub-module. They have specific Eurogam signal names and functions.
ECLTRG0-1	Buffered on and off the backplane. Single ended ECL. The two signals are passed to the P3 sub module and controlled as for the TTLTRG lines.
LBUS0-11	6 are Analogue. These are terminated and buffered on this sub-module with outputs capable of driving 50ohm lines. The outputs connect to the front panel. 1 is TTL from the P3 control register. 4 are single ended ECL used for an event number.
SUMBUS	This is buffered with a 50ohm driver connected to the front panel.
CLK10	This is derived on the P3 sub-module.

Table 3.b.1 P2 module buffering

4.SOFTWARE INTERFACING.

The VME slave situated in the VXI adaptor sub-module is accessed in the short-i/o region of the VME address space. It is an A16/D16 slave with interrupter capability. Since it forms the configuration registers for the VXI slot0 function it has a fixed address.

The address is 0 in the logical address space of the VXI configuration registers space. This address space is part of the short i/o and VME addresses are calculated from V * 64 + 49152, where V is the logical address. This leads to an address of @C000 for this slave.

The address space occupies 64 bytes. The function of some of the words are defined in the VXI specification, others are user definable. The Resource Manager requires the implementation of a message based device in the crate, with location monitors on specific configuration register accesses. The necessary signalling to the processor is via a VME interrupt.
The configuration registers and their bit functions are shown in the table 4.a.1.

a.Configuration and Message registers.

(R) = read functionality
(W) = write functionality

Address	Function	Contents
0	ID (R)	Device class, address space, manufacturers ID number.
2	Device type (R)	Required memory, and model code.
4	Status(R) and Control(W)	4 mandatory status and 3 mandatory control bits, the rest are device dependent.
6	A24/A32 offset(R/W)	For allocated address offset, not used.
8	Protocol(R)	Indicates protocols, and capabilities of device.
	Signal(W)	For device to device signalling, sender writes a signal into this location.
A	Response(R)	Status of devices communication registers.
	Data Extended(W)	Optional data register.
C	Data High(R/W)	Data register with location monitor on writes.
E	Data low(R/W)	data register with location monitor on read and write.
10-1E	Reserved	Implemented as RAM all bits Read/Write.

Table 4.a.1 configuration registers

b.Device dependent registers:-

Address	Function
20	MODID (R/W). Used by the processor to control the MODID lines. bits 12 to 00 map as MODIDnn, bit 13 is high to enable MODID drivers.
22	Location Monitor status(R)/control(W) bits 7 to 00 interrupt vector number . bit 8 set if write to Signal register (@8)

bit 9 set if write to data extended register (@A).
bit 10 set if write to data high register(@C)
bit 11 set if write to data low register (@E)
bit 12 set if read from data low register (@E)
bits 12 to 8 may be set and cleared by a write to this location.
bit 13 is not currently used.
bit 14 is read/write, when set enables the interrupt .
bit 15 is read only, and is the OR of bits 12 to 8.

24 External bus address word .
Data written into this location forms the 16 bit address for the external bus used for accessing other sub-units.
bits 15 to 00 write only.

26 External bus data word.
bits 15 to 00 read/write. See section 3.a.2

28-2E Implemented as RAM all bits Read/Write.

30-36 not implemented as yet.

38 Signal(R)/Protocol(W)
For processor access to these registers.

3A Response(W)/extended data (R)
For processor access to these registers.

3C Data High (R/W)
For processor access to these registers.

3E Data Low (R/W)
For processor access to these registers.

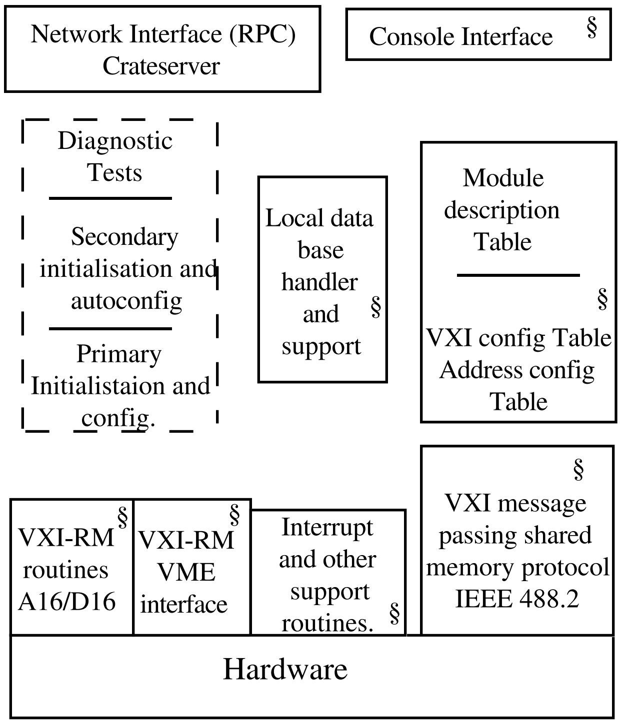
c.Resource Manager functions

For proper operation of the VXI-system the specification defines some software features. It is implemented in a modular fashion and makes use of a layer model (see Fig 4.c.1).

All modules are part of the VXII run time library (VXIRTL). All tasks which require access to any VXI specific operation has to use these routines. They take care of all aspects of accessing the VXI system during data acquisition.

Furthermore they take care of the requested VXI operations and store the data written into and read out of from the VXI specific part of the hardware - the configuration registers - in a small data base which is used for test and debugging of the system. During the system Initialisation this data base is used to allow a proper initialialistaion of the VXI hardware. In addition to the storage of the VXI specific Information, some module specific data and action requestsare stored inside this data base

Fig 4.c.1 Software Overview



The parts in the dashed box are the VXI Resource Manager specific programs which are executed during the startup of the VXI crate, all others are part of support routines. The part with the double box contains database tables. Boxes marked with § are part of the VXI-RM software.

The initialisation task is specified by the VXI specification:-

1) Identify the logical addresses of all devices in the system.using the MODID lines for relating physical postions to logical addresses.
2) Check that all devices with self-test capability pass their tests, and put those that do not into the Safe state.
3) Identify the address space requirements of A24 and A32 devices, and allocate them base addresses.
4) Set up the Commander/Servant heirarchy among devices.
5) Allocate the IRQ lines to the interrupters and handlers.
6) Initiate system operation. (No other master device may begin using the bus until sent the 'Begin Normal Operation' command.)
7)Implement a message based device.

All of these functions are carried out at power-up, and rarely when the system is re-initialised.

Currently we support static and dynamic configurable VXI devices. The full support of the VXI message based protocols will be implemented in a second developmemnt step, we currently use use only register based VXI devices.

After the initialisation, further VXI module testing and and diagnostics can be executed by a set of module specific programs.

After the test are passed successfully the VXI resource manager software enables access by starting a so called

"Crateserver" which will take over the further control of the VXI crate.

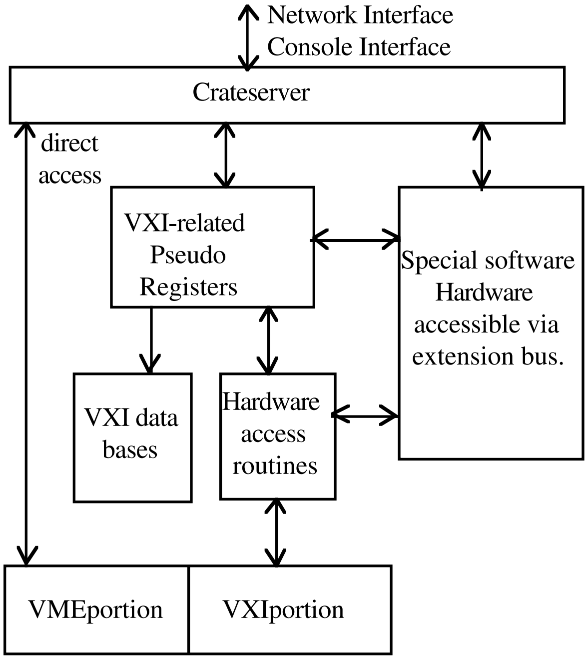


Fig 4.c.2 Crateserver: The VXI crate system is controlled be a so called Crateserver. This task controls on a rather high level the access to the VXI system.

The Crateserver has an entire network interface, which sits on top of the RPC (SUN Microsystems Remote Procedure Call) interface, and a Console interface which can be used for debugging and frontend diagnostics of one VXI crate.

To simplify the software, the Crateserver is able to access directly the hardware of the system (register access). Unfortunately this is not feasible in several cases, its just too slow. To overcome this a so called "pseudo register access" is implemented, which hides the complexity of the access by special routines, and is also part of the run time library

Most parts of the code are implemented in an operating system and processor type independant way. In the future we will implement on top of this software an X-window based user interface to allow a mucheasier way of operating the VXI system, especially during test, debugging, and failure analysis.

5.REFERENCES

1. VXI bus specification revision 1.3. VITA 10229 N. Scottsdale Road, Suitre E, Scottsdale, AZ 85253 USA.

2. VMEbus specification IEEE STD1014-1987 from VITA 10229 N. Scottsdale Road, Suitre E, Scottsdale, AZ 85253 USA.

3. Altera MAXPLUS data book 1990, Altera Corporation, 2610 Orchard Parkway, San Jose, CA 95134-2020

4. PLX technology Inc, 625 Clyde Avenue, Mountain View, CA 94043.

5. A new concept in data acquisition electronics, control, and monitoring for the next generation of nuclear gamma ray spectrometer. IEEE NS-37 no2 pages 326-330.