Designing RF Control Subsystems using the VXIbus Standard*

Jeffrey D. Stepp, Frederick C. Vong, James F. Bridges
Argonne National Laboratory
9700 S. Cass Avenue, Argonne, IL 60439 USA

Abstract

Various components are being designed to control the RF system of the 7-GeV Advanced Photon Source (APS). The associated control electronics (phase shifters, amplitude modulators, phase detectors, automatic tuning control, and local feedback control) are designed as modular cards with multiple channels for ease of replacement as well as for compact design. Various specifications of the VXIbus are listed and the method used to simplify the design of the control subsystem is shown. A commercial VXI interface board was used to speed the design cycle. Required manpower and actual task times are included. A discussion of the computer architecture and software development of the device drivers which allowed computer control from a VME processor located in a remote crate operating under the Experimental Physics and Industrial Controls Software (EPICS) program is also presented.

1. INTRODUCTION

RF field parameters must be precisely regulated in order to confine and accelerate the APS positrons. Precision RF instruments control and monitor the RF field parameters. Sophisticated computer controls and diagnostics are necessary to support remote supervision and operation of the RF instrumentation. A modular RF control system has been implemented using the architecture of the VMEbus Extension for Instrumentation (VXIbus). Features include broadband backplane analog interconnections, precision timing signals, EMI/RFI compatibility, standardized configuration, and communication protocols. RF and control, signal conditioning, signal processing, and interface circuitry can be housed in the same module. Additional features include modular instrument-on-a-card applications and synchronization of many instrument channels. A VXIbus backplane interface links the modules to a controlling microprocessor for high-speed signal acquisition and data processing [1].

This paper describes the implementation of the APS low-level RF control system using the VXI architecture. It outlines how control system development was done with minimum staffing and with an off-the-shelf VXI interface module. Advantages of the VXI architecture are increased equipment density, eliminated cabling, and simplified system integration.

Using a register-based VXIbus interface, we can divide the function of a "complete instrument" into a number of VXIbus modules such as signal acquisition, signal processing, and control functions. Processing capabilities can thus be remote from the acquired data. This is then a shared-resource environment in which one or more processors do all the processing for the entire system [2].

The interface module uses most of the VXIbus features, including module self-test functions, module slot identification, failed-module inhibit capability, and the dynamic configuration on of a module's base address.

II. DESCRIPTION

The Bruel & Kjaer Type 3154 VXI User Module is a single-slot, C-size, Extended Register Based device, with which one can design and build a VXIbus module without designing a customized VXIbus interface. The following features of the B & K module were used to design the associated control electronics (phase shifters, amplitude modulators, phase detectors, automatic tuning control, and local feedback control) with multiple channels:

- an isolating, low-noise analog power supply
- digital power supply lines
- 3 TTL trigger lines
- a 10-MHz clock line
- 4 isolated input/output 16-bit ports from VXIbus interface registers for control purposes [3].

Approximately one-half of a C-size VXI PC board is available for user circuitry. This space was used to hold the RF modules, associated A/D and D/A converters, and timing and interface circuitry. Power and cooling are done by the VXI mainframe. See Figure 1.

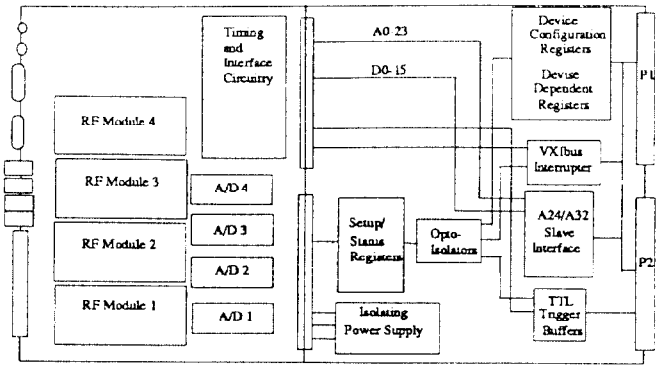

Figure 1. VXI Module block diagram.

The services of one full-time engineer, one full-time technician, one part-time draftsman, and one part-time software engineer were used over a period of six months to design and build one low-level RF system with four VXI modules.

VXIbus Interface

The register-based protocol defines the conventional method of VMEbus data transfers to address-mapped configuration and control registers. Each VXIbus module contains a set of 32 I/O registers through which the configuration, control, and sig-

*Work supported by the U.S. Department of Energy, Office of Basic Sciences, under contract W-31-109-ENG-38.

0-7803-1203-1/93$03.00 © 1993 IEEE

nal access are made for the module. Standard VXIbus configuration registers are included in these 64 bytes. These registers identify the device, its address space, and some status and control for the module. All binary, analog, and timing signals required for RF modules are represented as 16-bit data registers. The registers represent channels of analog control and monitor signals which access the RF electronics.

The register-based VXIbus interface is located in every VXIbus module. It controls the data cycle, interrupt cycle, and backplane handshaking with the VME processor card. This interface contains the backplane buffers, VXI configuration registers, and the timing signal generator. The module has additional VXIbus features that are not used in the VMEbus environment, including module self-test functions, module slot identification, failed-module inhibit capability, and the dynamic configuration of a module's base address.

Timing and Triggering

Each module contains a local timing circuit that synchronizes the operation of the RF control electronics. The VXIbus controller buffers external timing signals and drives the backplane ECL trigger, TTL trigger, and clock signals (ECLTRGn, TTLTRGn, CLK10). A local timing circuit on each VXI module uses the backplane signals to generate five variable-delay, variable-width pulses, providing all timing triggers and windows to control the RF electronics on the module. In this manner, the timing signals on all modules within an entire mainframe are synchronized to a common gating signal within 20 ns accuracy and jitter.

Electromagnetic Compatibility

Four areas of electromagnetic compatibility that are covered in the VXIbus specification are conducted noise, far-field radiation, near-field magnetic radiation, and electric-field radiation. Conducted noise is reduced by bypass capacitors located on all IC supply leads and also close to the backplane power pins. Far-field radiation is reduced by adequate ground planes on the PC boards, RF circuitry shielding, and the metallic VXI module enclosure. Near-field magnetic radiation is reduced by limiting the inductance paths of current loops. Additional shielding can be inserted between modules by the mainframe manufacturer. Electric-field radiation is limited by the grounded metallic module enclosure.

III. COMPUTER-BASED CONTROL SYSTEM

A computer-based control system has many advantages. A well-designed system can reduce staffing requirements. The system can be reconfigured easily when hardware and software modifications are made. Controls can be added in piecemeal fashion. Different computers located in separate areas or buildings can readily communicate through a local area network. Data can be easily archived and made instantly accessible to various workstation users simultaneously [4].

The APS control system is derived from the Experimental Physics and Industrial Control System (EPICS), which was developed by Los Alamos National Laboratory and Argonne National Laboratory. EPICS is a software system with which enables the user to readily develop control software in a simple and flexible environment.

The hardware for the test system, shown in Figure 2, consists of a workstation, a VME system, and a VXI system. The workstation runs the UNIX operating system with the X-windows graphical user interface (GUI). The operator communicates with the control system using a Sun workstation. The operator interface (OPI) running on the workstation communicates with the VME system through the Ethernet local area network using the TCP/IP protocols. The OPI, based on the X-windows GUI, configures the text and graphical representation of the data and interactive data entry for the test system. The VME system input/output controller (IOC) functions as a front-end computer doing the real-time control, data conditioning, and data acquisition. The IOC is running on top of the VxWorks real-time kernel. The VXI system is connected as a subsystem to the VME system through the MXIbus. This configuration is essentially based on the EPICS system architecture [5].

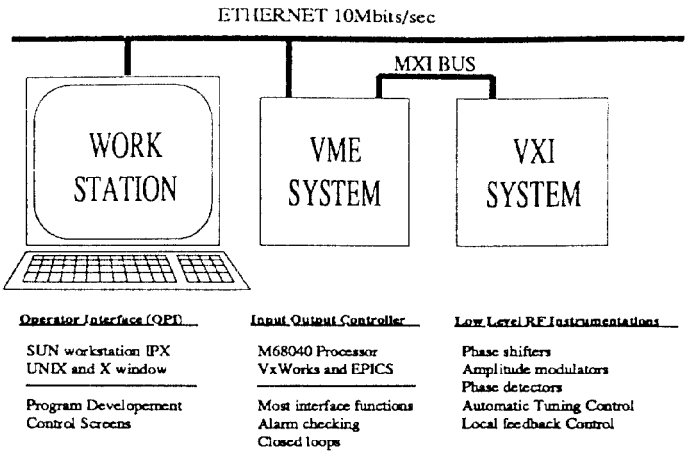

Figure 2. VXI control system hardware.

The heart of an IOC is a memory-resident database. Each signal is represented by a record. A record contains several fields that describe the operating range, conversion factor, scanning mechanism, and different alarm trigger levels. The database is made using the Database Configuration Tool (DCT) running on the workstation. The database is downloaded to the VME system during run time.

A primary EPICS concept is that each hardware-specific routine, such as the device driver, is isolated from the IOC core software by a record and device support layer (see Figure 2). The database of the IOC core communicates with the hardware via the record and device support layer and the hardware's device driver [6]. A device driver is needed for each new piece of hardware.

The VXI instrument prototypes can be classified into two categories: analog input record and analog output record. The block diagram of the device support and device driver routines is shown in Figure 3. Two device support entry tables (DSET) are shown in the block diagram, one for the analog input record and one for the analog output record. The main difference between them is the DSET of the analog input record has the

'read_ai' entry but the analog output record has the 'write_ao' entry. The items in the entry tables are predefined in EPICS.

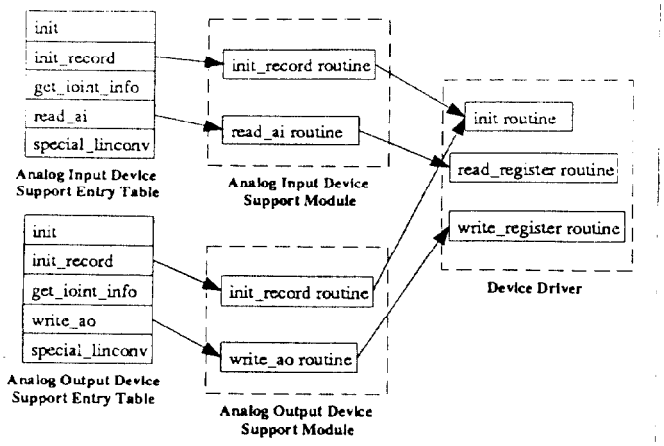

Figure 3. Device Support Layer.

The 'read_ai' routine for the analog record first calls the device driver 'read_register' routine to get the raw value, then converts the raw value into proper form according to the corresponding record conversion setting defined in the database. The value is returned in the proper form and raises an alarm if an error is reported by the 'read_register' routine of the device driver. The 'write_ao' routine reverses the process of the 'read_ai' routine. It converts the value passed by the record into the hardware data format according to the corresponding record conversion setting, calls the write_register routine, and passes along the data, and raises an alarm if an error occurs.

Device Driver Implementation

Our prototype VXI instrumentations are based on the Bruel & Kjaer Type 3154 User Module as the VXIbus interface. Each module has the same set of isolated input/output 16-bit ports; therefore, only a single device driver is needed. The device driver has three routines—for hardware initialization, read operations, and write operations.

During system power-up, the EPICS VXI resource manager initializes all the standard VXI tests, such as module self-test, module slot identification, dynamic configuration of a module's base address, and inhibiting failed modules. The initialization routines of the device driver first calls the resource manager to obtain the base address of the module specified by the signal assignment in the database and then calculates the address of the corresponding register. If this is an input operation, the initialization routine sets up the associated ports for a strobe operation by setting the corresponding bit of the Strobed/Transparent Configuration Register in the User Module. Finally, the routine reports any errors to the calling routine.

For read and write operations the routines are fairly straightforward. The read routine peeks the associated register and returns the value to the calling routine. Similarly, the write routine pokes the associated register with the value passed by the calling routine.

ACKNOWLEDGMENTS

The authors wish to thank Terry Smith and Christine Verdico for their part in integrating the system. We also wish to thank the ASD/APS Controls Group for providing the application code, writing the EPICS database, and designing the interface control panels.

REFERENCES

[1] C. Ziomek, "Interfacing RF Control Electronics to the VXIbus for the Ground Test Accelerator," Proc. Neutral Particle Beam Tech. Symp., San Diego, CA, May 21-24, 1990.

[2] C. Ziomek, "Accelerator RF Instrumentation Implemented with the VXIbus," ATE & Instrumentation Conference, Anaheim, CA, Jan. 15-17, 1991.

[3] "User Module Type 3154 Hardware Users Guide," Bruel & Kjaer A/S, Naerum, Denmark, 1992, 1-2.

[4] A. J. Kozubal, L. R. Dalesio, J. O. Hill, and D. M. Kerstiens, "Run time environment and applications tools for the Ground Test Accelerator Control System," in *Accelerator and Large Experimental Physics Control Systems*, D. P. Gurd and M. Crowley-Milling, eds. 288-291, (ICALEPCS, Vancouver, BC, Canada, 1989).

[5] M. Kraimer, "Experimental Physics and Industrial Control System (EPICS) Input / Output Controller (IOC) Application Developer's Guide," EPICS Manual, pp. 41-57, March 1992.

[6] R. T. Daly, F. C. Vong, M. R. Kraimer, "The APS Radio Frequency Test Stand Control System," Real Time '91 , Computer Applications in Nuclear Particle and Plasma Physics, Julich, Germany, 25-28 June, 1991.