SLAC-PUB-6211
May 1993
(A/I)

Adding PCs to SLC Control System

T. Lahey, S. Levitt, R. MacKenzie, N. Spencer, K. Underwood

Stanford Linear Accelerator Center, Stanford University, Stanford CA 94305 *

Abstract

The SLAC Controls Department has interfaced IBM-Compatible PCs to the SLC Control System, for use by the Final Focus Test Beam (FFTB) experimenters, who are building new accelerator equipment and developing and testing it at their home institutions. They will bring the equipment to SLAC and integrate it into the control system using a new software package. The machine physicists and operators will use the existing SLC control system applications and database device types to control and monitor the equipment. The PCs support a limited control enviroment: they run DOS and exchange messages with the existing control system via TCP/IP over ethernet, using the new SLC Area Message Service. This mechanism will also allow SLC to implement other commercial device controllers that can communicate over ethernet and run the same software interface code.

Introduction

Final Focus Test Beam (FFTB) is an international collaboration that will run this year at Stanford Linear Accelerator Center (SLAC). The collaborating institutions are providing instrumentation and control computers that they are bringing from their home institutions and connecting to the existing Stanford Linear Collider (SLC) accelerator control system.

This paper describes the method used to connect this instrumentation to the SLC control system so that the operator interface from the control room matches the existing control system.

Justification

PCs were chosen for the control computers because they can be used in a standalone mode to develop instrumentation, they are commonly available, and they have good development tools. The international collaborators can develop the instrumentation at their home institutions, and bring their instruments and PC to SLAC.

*Work supported by the Department of Energy, contract DE-AC03-76SF00515

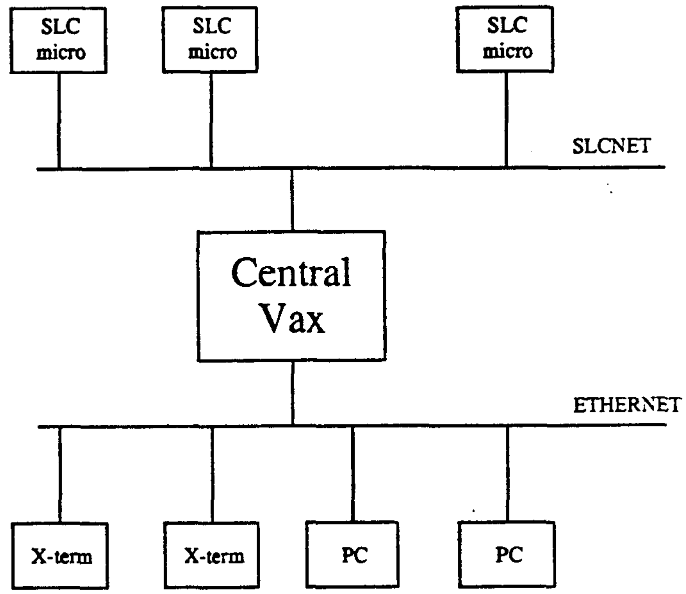

Figure 1: Adding PCs via ethernet to SLC Control System

Additions to SLC Control System

The existing SLC control system generally consists of:

- a SLC Control Program (SCP) that operators use from an X-windows compatible terminal.
- the database describing accelerator devices.
- a host VAX on which we run the SCP, database, and other applications.
- standard SLC 80386 micros that run the Intel RMX operating system, and are geographically distributed throughout the accelerator to control and monitor the accelerator devices.
- the SLCNET proprietary network and network software.

An FFTB PC is a IBM PC-compatible that runs MS-DOS, and communicates with the VAX over ethernet using

Presented at the Particle Accelerator Conference (PAC '93), Washington, DC, May 17-20, 1993

the TCP/IP protocol. Figure 1 shows the PC added to the SLC control system.

FFTB experimenters bring accelerator instrumentation that is controlled by a PC-compatible computer. The PC runs a program that receives and processes device commands. The program is built from experimenter software linked to a SLAC-developed library.

SLAC Controls has written a PC library that provides the basic micro services and device support, a VAX-resident Database Server (FFTB_DBS) that allows a PC to access the SLC database, and a VAX-resident Network Server (A_SERVER) that redirects messages to the PCs from existing SCP and applications.

The network software is SLAC's Area Message Service (AMS) [1] that resides on the VAX and PCs and sends messages using commercial TCP/IP software: TGV's Multinet on the VAX, and DEC's TCP/IP on the PCs.

To support FFTB, we have not modifed the SCP, other application programs, or the database device types.

Figure 2 shows the software components.

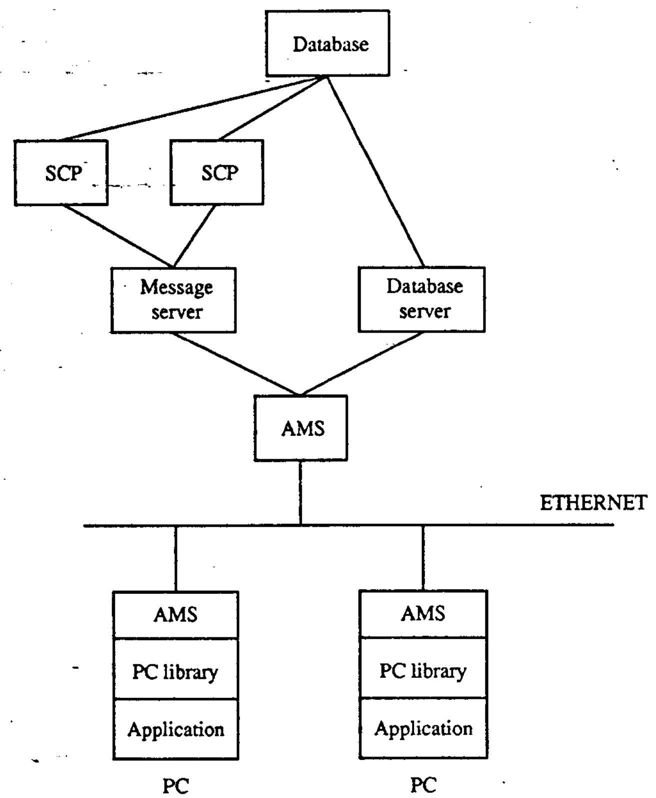

Figure 2: Software Components for FFTB PC

To add an FFTB PC to the control system, we identify the standard SLC device-types needed for the instrumentation, define the FFTB devices in the database, and define the PC as an FFTB PC.

Using PCs, KEK provides FFTB quadrupoles and a quadrupole-support table, and Beam Size Monitor instrumentation. A SLAC/DESY group provides FFTB beamline alignment instrumentation.

Alternate Platform

The main use of this package is on a PC-compatible. The FFTB package can also run as a process on the VAX, emulating the functionality of a remote micro. An initial implementation of this VAX process, called a VAX-based micro, is presently used to control FFTB magnet mover devices. The movers are physically connected to standard SLC micros, and were initially developed and tested with a standalone VAX program. To integrate these movers into the control system, the FFTB package runs as a VAX-based micro, controls the mover devices and supports device commands issued from the SCP. The VAX-based micro sends messages to standard SLC micros to control the physical hardware, performs special transformations that are specific to the FFTB movers, and replies to the SCP commands.

The VAX-based micro is also used to test PC software.

Functionality of a PC

To use a PC to control devices, the PC must support a subset of the standard SLC micro functionality. This includes:

- devices in the database: the PC supports a subset of standard SLC devices. The PC emulates all the necessary characteristics of these device types, updating all values in the database for the given device type. A PC owns database units to describe any instruments that are connected to the PC.
- communication with SCPs, supporting existing application messages.
- periodic events to trigger updates of PC device values in the database.

The PC program acts both as a server and a client. It is a server to the SCP, processing SCP messages that contain device control or monitor command. To support the SCP message, the PC must get and put device values from/to the SLC database that resides on the VAX. The PC is then a client to the database server, requesting get/put operations of values in the database.

The SLAC-written PC library hides many of the details of the SLC micro and its integration within the control system. The interface with experimenter-written code is well-defined [2] and simpler than similar interfaces in standard SLC micros.

The PC library contains device-specific functions for the supported device types. The experimenter initializes a device buffer, and then gets/puts values from/to the database for devices via the device buffers. For each device buffer, the experimenter provides the address of their routine which will do the physical device I/O and command processing. The PC library and database server support four device buffer types: actuator, analog, digital, and general.

A PC uses: actuator buffers to get/put database values for magnets and power supplies, analog buffers for floating point values, digital buffers for digital states (bits), and general buffers to get device values for any other type of device in the database.

The PC library contains a main function that initializes the network, and then enters a loop that dispatches SCP messages to the experimenter-written code for processing, dispatches commands to experimenter-written code requesting a periodic update of devices values to the database, and calls the user each loop execution.

The experimenter also writes three pre-defined functions: USER_INIT to intialize their hardware and actuator, analog, digital, and general device buffers; USER_CODE to execute periodic functions needed each program loop; and USER_STOP to shut-down their application.

The PC can get values from the database for devices in any part of the accelerator. The PC must maintain current values for devices that it owns in the database.

PCs do not receive SLC timing interrupts. Timing signals are generated by standard SLC micros, and delivered directly to hardware that is connected to the PC.

Database Server

The VAX-resident database server, FFTB_DBS, provides access to the database via device buffers. It performs device buffer initialization, get and put of values via device buffers, cancellation of device buffers, protection of database access to allow a PC to write to only its own database devices, and other important support functions. The database server contains critical information to access the database. PCs do not directly access database values, thus protecting the database. The server allows easy control of database access.

Message Server

The VAX-resident message server, A_SERVER, dispatches all commuications between SCPs and PCs. When the SCP or other application program calls the existing message service, and wants to communicate with a node connected to ethernet, the message is routed to this server. A_SERVER sends/receives messages via AMS to the ethernet nodes. We updated the existing SLC message functions to recognize a node that is connected to ethernet, and to redirect the message to A_SERVER.

Both FFTB_DBS and A_SERVER use a list of legal nodes, providing security for which nodes can connect to the control system via AMS.

The number of available DOS TCP/IP connections is a critical resource. We have used a single message server to reduce the number of connections between a PC and the SCPs and other VAX applications.

Project Status and Futures

We are close to completion of this project. The FFTB experiment will use PCs and VAX-based micros this summer and autumn to measure the alignment of movers, control and monitor quadrupole-support table, measure beam size, and control the movers.

For future projects, this package can be used to connect smart instruments to the control system. We have started to migrate this package to an HP Unix system for control and monitoring of smart instrumentation. The AMS network is already implemented on the HP platform. We plan to compile and link the PC library on the HP platform, define database devices and control the instruments from the existing SCP and other applications. We can add additional device commands to control/monitor the smart devices.

Using this package for VAX-based micros, we support unique devices that exist in small numbers. When device support is added to a standard SLC micro, it is generally added to all the micros in the control system. When the support is complicated and requires prototyping, or it is needed in only one part of the accelerator, we can now implement the micro that controls the devices, in a VAX-based micro or in a PC.

References

[1] M. Crane *et al*, AMS: Area Message Service for SLC. *These proceedings, SLAC-PUB-6166 (1993).*

[2] T. Lahey *et al*, SLC PC-VAX Link for FFTB: PC Users Manual. *SLAC Controls Department Software Manual* (November 1992).