A NETWORKED REAL TIME CONTROL SYSTEM FOR THE STANFORD PHOTON RESEARCH LABORATORY

C. Cork(a), J. Fox(b), R. Melen(c)
Stanford Photon Research Laboratory
Stanford University
Stanford, CA 94035

ABSTRACT

The Stanford Photon Research Laboratory (SPRL) is constructing a 1 GeV electron storage ring with linac injector. This facility will sponsor basic research in accelerator physics, free electron lasers and applications of synchrotron radiation. This paper presents the design of the SPRL control system and highlights the functions and performance of its major constituents.

1. INTRODUCTION

The SPRL storage ring under construction at Stanford will be a low emittance, 1 GeV electron machine designed to operate with 1 ampere circulating current[1]. The ring resembles a racetrack with one or more straight sections on the East side being used for Free Electron Laser (FEL) undulators and a chicane on the West side containing four undulators for VUV Synchrotron Radiation (SR) research. The control system must provide the monitoring and control functions to operate the ring in both FEL and SR modes, as well as provide for future control of the linac injector and of scientific experiments which interact with the storage ring.

(a) Currently with Stanford Synchrotron Radiation Laboratory.
(b) Currently with IBM Almaden Research Center.
(c) Currently with Quinsys Technology.

*Work sponsored by Air Force Contract F49620-84-C-0012

The initial storage ring configuration will have 48 bend, 71 quadrupole, and 40 sextupole magnets with associated horizontal and vertical trims that are used as beam position correctors. Initially, 50 Beam Position Monitor modules will be monitored by the control system. There will also be numerous vacuum, coolant water, and thermal interlocks that are monitored by the control system. In all, there will be approximately 475 analog and 425 digital signals interfaced to the control system.

2. CONTROL SYSTEM ORGANIZATION

The SPRL control system is based upon a distributed and hierarchical architecture. The benefits of this approach have been described in numerous recent articles [2][3].

The control system architecture is outlined in figure 1. The system consists of an expandable set of loosely coupled computers which are organized into a functional hierarchy. This system consists of the following major components:

* A system supervisor computer (Digital Equipment Corporation [DEC] VAX 8300) near the main control area;

* A set of console computers (DEC VAXstation-II/GPX with VSV21 graphics boards) in the main control area;

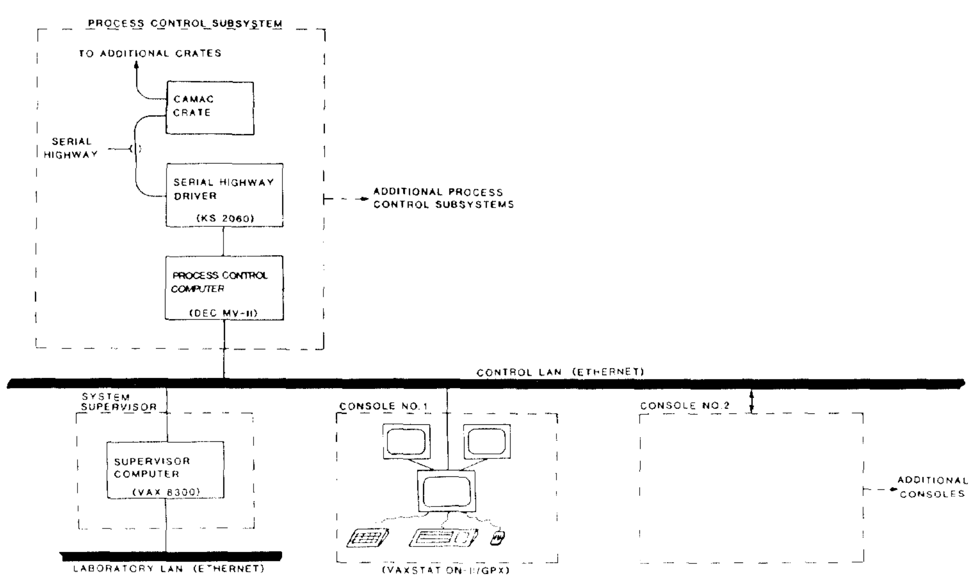

FIGURE 1: HARDWARE ARCHITECTURE

* A set of process control computers (DEC microVAX-II's) with individual CAMAC serial highway drivers (Kinetic Systems model 2060) in the main control area;

* A set of CAMAC Instrumentation crates and modules which are directly interfaced to the process instrumentation;

* A Machine Control Ethernet LAN which interconnects the system supervisor, consoles, and process control computers;

* A Laboratory Ethernet LAN (SUnet) interface which connects the system supervisor computer to the terminal servers and to the outside world.

The system will actually exist in two or more stages. The initial stage is a minimal operational system where the system supervisor computer will be absent, its functions being performed by the process control computer. Subsequent stages will include the system supervisor computer, one or more process control computers to control the Linac and FEL experiments, and additional operator consoles as required.

A more detailed description of the system architecture and its main components are available elsewhere[4][5].

3. CONTROL SYSTEM SOFTWARE

The SPRL software control structure closely resembles the system architecture. A distributed, hierarchical structure is being used for control and data acquisition (figure 2).

The first level processing of signals is performed in the process control computers. Here device-dependent programs handle the interface between the machine instrumentation and the local dynamic database. The CAMAC serial highway driver, SHDRIVER, is a version of the ORNL driver for the Kinetic Systems model 2060 CAMAC serial highway controller[6]. The process control interface program, XCAMAC, was derived from the SLAC SPEAR/PEP program with the same name[7][8]. This program continuously polls the machine instrumentation and updates the local dynamic database. A collection of programs are used to initialize and access the dynamic database.

The next level is the process server level. This software level also runs on the process control computers. The server maintains closed-loop control of several key process variables (eg. ramping of the magnets), monitors variables for out of limit conditions, and handles alarm conditions. The process server also handles the protocol for communication on the machine control LAN and it transfers process information between the local dynamic database and the client systems on the network.

The third level software is the supervisor level. This software runs both on the system supervisor computer and on the console computers. At the operator consoles, the software provides an interface between the operator and the machine process variables. The operator console software will consist of a local multiwindow display server (X-Windows from MIT[9]), a customized window manager (SWM), and a number of application programs for machine modeling, control knob interfacing, process monitoring, and graphic trend analysis. There will also be a network subroutine library for communication with the system supervisor and process control computers.

At the system supervisor computer, tasks will run to maintain process history records and perform periodic data logging. Tasks will also run here to perform modeling services upon demand by the operator console computers.

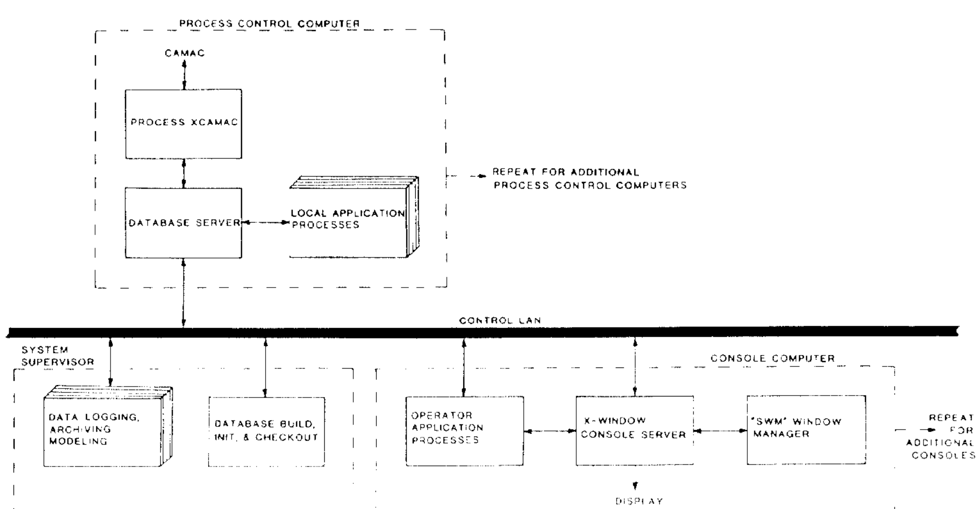

FIGURE 2: SOFTWARE ARCHITECTURE

MESSAGE LENGTH (bytes)	CASE 1		CASE 2	
	Message Rate (per second)	Throughput (Kbytes/sec)	Message Rate (per second)	Throughput (Kbytes/sec)
10	118	2.4	135	2.7
100	117	24	130	26
1K	70	140	75	150
10K	9	185	11	230
30K	3	192	4	232

TABLE 1. Results of performance tests for two MVII's communicating using DECNET. CASE 1: One Master and One Slave Process; CASE 2: Two Masters and Two Slaves. The values presented for CASE 2 are the aggregate rates.

4. NETWORK PERFORMANCE MEASUREMENTS

In order to help validate the design of the computer system architecture, we performed a series of tests on the proposed network. These tests measured the message rate and total data throughput for various length messages passed between two DEC MVII computers using DECNET.

The tests were performed using the "transparent" DECNET system service routines called from FORTRAN running under the VMS operating system. A master program in one computer established a link with a slave program in a second computer and then sent groups of various length messages that were "echoed" by the slave. There was virtually no error checking performed at the user level. A second set of tests were performed using two sets of programs operating simultaneously.

Table 1 presents the results of the tests. All of the tests were performed on dedicated computers with no contention from other user programs. About 90% of the MVII processing cycles were consumed for all message lengths when one set of programs was running and 100% was consumed when two sets of programs were running.

These tests indicate that the performance of VMS/DECNET is easily sufficient to support a real-time system meeting the SPRL requirements. It is estimated that <15 messages per second at an average length of 1K bytes will be required. Therefore it appears that less than 20% of the network's peak capability will be routinely used.

5. CONCLUSIONS

The SPRL project is well underway and project completion with stored beam is anticipated in June, 1988. The control computers, CAMAC highway, Ethernet LAN's and their associated software have been integrated and the database acquisition software is running. Work is progressing on the operator consoles and graphics software. Future papers will report on the operational experience with the SPRL control system.

6. ACKNOWLEDGEMENTS

Most of the hardware and software designs grew out of work performed by the SLAC I&C department. In particular, our basic system design was based upon the PEP and SPEAR storage ring control systems. We have relied heavily upon recommendations from SLAC engineers, operators, machine physicists, and users who have had experience with these storage rings. In particular we would like to thank H. Wiedemann for his perspectives as a machine physicist and, T. Taylor for his experiences as head of operations at SPEAR and PEP.

REFERENCES

[1] L. Emery, et al, "Stanford Photon Research Laboratory's VUV-FEL Ring Lattice," This conference proceedings.

[2] J. Altaber, P.G. Innocenti, R, Rausch; 6th Annual Workshop on Distributed Computer Control Systems, International Federation of Automatic Control, Monterey, 20-22 May, 1985.

[3] W. Klotz, "The ESRF Control System," ESRF Internal Report, ESRF-SR/COMP-86-01, December 1986.

[4] R. Melen, "The SXRC Storage Ring Instrumentation and Control Design Report," SPRL Design Report, SRN-09, December 1986.

[5] J.Fox, "Specifications for the SPRL Timing System," SPRL Design Report, SRN-07, June 1986.

[6] E.T. Blair, "System Manager's Reference Manual for the ORNL VAX CAMAC System," Internal document, Oak Ridge National Laboratory, August 1985.

[7] S. Howry, T. Gromme, A. King, M. Sullenberger, "A Portable Database Driven Contol System for SPEAR," Presented at 1985 Particle Accelerator Conference, Vancouver, B.C., Canada, May 1985.

[8] R. Melen, "Specifications for the SPRL Real Time Data Acquisition and Database Software Package," SPRL Design Report, SRN-08, February 1986.

[9] Digital Equipment Corporation, "ULTRIX-32W Technical Summary," DEC Document No. AA-GT87A-JN, 1986.