Proceedings of the International Conference on Accelerator and Large Experimental Physics Control Systems

November 11-15, 1991
KEK, Tsukuba, Japan
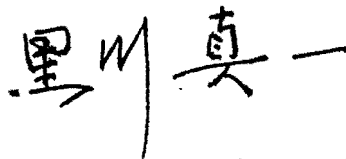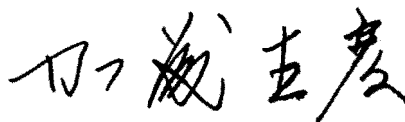
FOREWORD

International Conference on Accelerators and Large Experimental Physics Control Systems (ICALEPCS'91) was held on November 11 - 15, 1991 at KEK, Tsukuba, Japan. This was the first conference in this series held in Asia. It was a great pleasure for the organizers of the conference that more than 240 participated. Among them we were delighted to see large delegates from countires and institutions such as Russia, China, Korea, India, SRRC in Hsinchu, etc. from which very few had participated in the former conferences. This reflects the fact that the society has come to its maturity, to which the continuing effort by EPCS(Interdivisional Group on Experimental Physics Control Systems under European Physical Society) has contributed greatly.

The maturity was also demonstrated by a naming of the "standard model" of control systems which we heard during the conference. Also, in the conference, discussion was held on prospects of making a generic tool-kit of control system on the basis of world-wide collaboration. Of course, we are far from this goal; however, it is sure that we glimpsed our future during the conference.

Shin-ichi Kurokawa
Chairperson of ICALEPCS'91

Tadahiko Katoh
Chairperson of Local Organizing Committee

LIST OF INTERNATIONAL SCIENTIFIC ADVISORY COMMITTEE MEMBERS

BARTON, Donald S. *(BNL)*
BUSSE, Winffied *(HMI)*
DANEELS, Axel Jean *(CERN)*
DOHAN, Don A. *(SSCL)*
GURD, David P. *(SSCL)*
JAN, Gwo-Jen *(SRRC)*
KIMURA, Toyoaki *(JAERI-Naka)*
KUPER, Edward *(INP-Novosibirsk)*
LIU, Shi-Yao *(IHEP-Beijing)*
LUDGATE, George Arthur *(TRIUMF)*
MÜLLER, Klaus-Dieter *(KFA)*
NISHIMURA, Hiroshi *(LBL)*
PETERS, Franz *(DESY)*
SCHALLER, Stuart *(LANL)*
WADA, Takeshi *(RIKEN)*

BLUMER, Thomas *(PSI)*
CLOUT, Peter Norman *(VISTA)*
DASGUPTA, Subrata *(VECC)*
DUNAITSEV, Anatoly F. *(IHEP-Serpukhov)*
HUMPHREY, John (Rusty)*(SLAC)*
KATOH, Tadahiko *(KEK)*
KUIPER, Berend *(CERN)*
KUROKAWA, Shin-ichi *(KEK)* - Chairperson
LUCAS, Peter Wayne *(Fermilab)*
LUONG, Tam T. *(GANIL)*
NAVRATIL, Jiri *(CTU)*
OWEN, Edward Charles *(Daresbury)*
RAUSCH, Raymond *(CERN)*
STEINER, Rudolf *(GSI)*
WON, Sangchul *(PLS)*

LIST OF LOCAL ORGANIZING COMMITTEE MEMBERS

KATOH, Tadahiko *(KEK)* - Chairperson
KOHNO, Toshiyuki *(NIRS)*
MORITA, Koh-Ichiro *(NAO-NRO)*
OGATA, Hiroshi *(RCNP)*
PAK, Cheol On *(KEK)*
URAKAWA, Junji *(KEK)*
WATANABE, Shin-ichi *(INS)*

KIMURA, Toyoaki *(JAERI-Naka)*
KUROKAWA, Shin-ichi *(KEK)*
NAKAHARA, Kazuo *(KEK)*
OYAMADA, Masayuki *(LNS)*
TAKEDA, Shigeru *(KEK)*
WADA, Takeshi *(RIKEN)*

Organized by: KEK, National Laboratory for High Energy Physics

Supported by: European Physical Society
IEEE Nulcear and Plasma Sciences Society
Japan Physical Society

Sponsored by: Tsukuba EXPO '85 Memorial Foundation
Foundation for High Energy Accelerator Science

Contents

A USERS VIEW OF THE SPS AND LEP CONTROL SYSTEMS

R. Bailey
CERN
CH–1211 Geneva 23, Switzerland

Abstract

Every accelerator has a control system; at present the SPS has two, both of which are needed to run the machine. Consequently a user of the SPS / LEP complex has to be concurrently familiar with three control systems. While this situation brings problems it allows, even forces, comparison between the different systems, which in turn enriches the user viewpoint.

This paper assesses the SPS and LEP control systems from the point of view of the user, who may be an equipment specialist, operator, accelerator physicist or combinations thereof.

1. Introduction – what the accelerators do

Exploitation of the two large accelerators at CERN is a varied business. For the SPS in 1991 this amounts to running as a fixed target machine for over half the year, providing either protons (during 21 weeks) or sulphur ions (during 6 weeks) to the physics community. In conjunction with this the SPS acts as an injector to LEP, providing leptons in an interleaved repetitive supercycle. Furthermore about 15% of the fixed target running time is given over to machine development periods, when the SPS is required to run in some non–standard way, mostly as a testbed for the LHC. Finally, the SPS is also used in the other major mode of operation, as a proton–antiproton collider, for about 5 weeks.

In parallel with all of the 27 weeks of SPS fixed target running, LEP is taking beam either for Z^0 production or for a substantial machine development program, the latter amounting to about 30% of the total LEP running time.

For both machines, although mostly for LEP, installation and testing of new equipment is carried out throughout the year.

This diversity of operations and machine improvement is carried out from a common central control room, with the same teams being responsible for both the SPS and LEP. In particular, one group run the SPS in a variety of modes of operation throughout the year as well as running LEP. This means that these personnel have to be familiar with the different control systems used to interact with the accelerators. The same is true of the personnel responsible for equipment commissioning.

2. Overview of control systems available

From 1975 the SPS has been controlled, either exclusively or partially, via a system based on Norsk Data ND100 computers connected in a TITN star configuration [1]. The computers run SINTRON and the programmers are provided with the NODAL interpreter, libraries of graphics primitives and data modules and a means of calling FORTRAN executables [2].

From 1985 the major new requirement for SPS to provide beams to LEP meant a complete rewrite of the applications software. This was undertaken in a UNIX environment on an Apollo network, with C as the main programming language and Apollo–Dialog for the user interface. In the first instance access to the hardware was via a gateway into the existing TITN system. More recently the possibility exists to access some equipment completely independently from the TITN system, using the same overall Token Ring architecture as for LEP (see below).

Presently the SPS is run using a mixture of purely TITN (30%), Apollo via the gateway into TITN (50%), and purely Apollo Token Ring applications (20%) (see figure 1).

LEP applications also run on an Apollo network, with C as the main programming language and Apollo–Dialog for the user interface. All the Apollos are connected on a control room Token Ring, with communications out into the field through a bridge to a machine Token Ring running around the accelerator [3] . At several points around the ring there is a further bridge or gateway into either a regional Token Ring or an Ethernet network. Connected to these local area networks is a variety of configurations, allowing access into the hardware via several different equipment control assemblies, mostly using the MIL–1553–B standard (see figure 1).
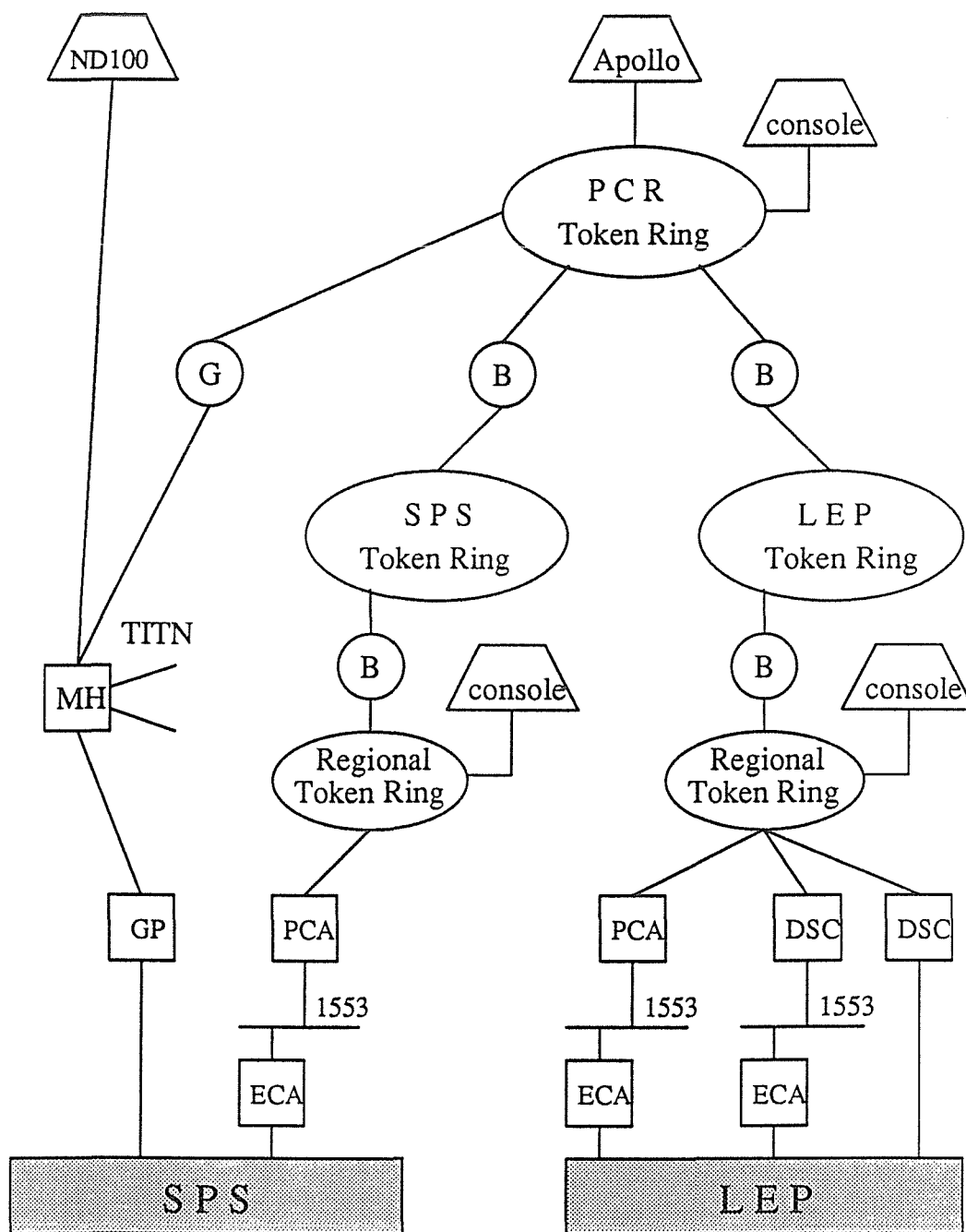
Figure 1
Logical schematic of the networks

3. Different types of user

The control systems of SPS and LEP are used at different times by a variety of different personnel. These largely divide into three categories; operators, accelerator physicists and equipment specialists, each of whom have somewhat different requirements for the control system. These requirements are not only for the underlying architecture (network, operating system etc), but also for the applications that run on top of it. In other words, the user here is seen as the person who runs the applications programs, rather than the person who writes them.

All types of user of course need reliable network communications, with good diagnostics when things go wrong. An adequate speed across the network from console to equipment is also generally required.

3.1 Equipment specialists

Equipment specialists need to access a diversity of accelerator hardware, setting and reading a multitude of parameters that are not of interest to other users of the control system. In many cases they also need to do this locally, in order to closely monitor the effects on their equipment. This means that they need to run specific programs both in the central control room and in the field, the latter requiring local console facilities. They may well want to run locally when the network is down. Most of these programs are written by the person who will run them, or at least by a close colleague, and as such the reliability of the application is not of great importance.

In many cases the amount of equipment accessed is far more than during normal operations, in order to thoroughly test a system, for example. For this reason the speed can be of prime importance to the equipment groups.

Key requirements;

> local console facilities
> execution speed

3.2 Operators

Operators rarely work on individual pieces of equipment, but rather on combinations of accelerator systems or even on the accelerator as a whole. In performing this work they prefer to see a high level of standardisation across the different applications and across the different accelerators. The applications also need to be easy to use, with the operator being presented with all the information that he needs but not swamped by auxiliary data that he rarely uses. Online documentation is a big help, particularly when the applications are new.

Since many tasks have to be performed at the right time in a sequence, the applications that perform them need to be highly reliable. Since operations is a long and repetitive process, it is essential that the speed of execution of programs is adequate, which generally means completion of the task in a matter of seconds. Good error reporting is also very important.

Key requirements;

> ease of use
> stability
> standardisation
> execution speed
> error reporting

3.3 Accelerator physicists

Accelerator physicists have essentially the same requirements as the operators, except for the important addition of flexibility to allow new, non-standard applications to be used. Indeed since machine development periods usually involve doing several unusual things, standardisation and error reporting are not so important.

Key requirements;

> flexibilty
>
> ease of use
> stability
> execution speed

4. Comparison of the different control systems

Table 1 summarises the results discussed in more detail here.

In all three cases the speed and reliability of the network is adequate. However when there is a problem, it is much easier to pinpoint on the TITN system than on the Token Rings, which have become extremely complex.

Local facilities are also better on the TITN, where much of the equipment data is stored locally rather than in a central database.

Table 1   Comparison of observations

	SPS old	SPS new	LEP
Network			
Speed	●●	●●	●●
Reliability	●●	●●	●●
Diagnostics	●●●	●	●
Local facilities	●●●	●●	●●
Applications			
Execution speed	●●	●●	●
Stability	●●	●●●	●●
Error reporting	●	●●●	●
Standardisation	●	●●●	●●
Flexibility	●●●	●	●●
Ease of use	●●	●●●	●●●

Key   The more blobs the better

●	poor
●●	adequate
●●●	good

4.1 SPS old

A key feature of the NODAL based control system is flexibility. It is extremely easy to produce a working application program, communicating with the machine and displaying data to the user. While this is an excellent feature, particularly for equipment testing or for one–off applications, as operations become more complicated it becomes more difficult to control the overall coherence of the system.

In the SPS the operational applications grew out of equipment commissioning programs, essentially on a system by system basis, and in an iterative way. As an example quadrupoles, sextupoles, octupoles etc were all controlled by different suites of programs all essentially doing the same thing. Adding a new system involved adding a new suite of programs to control it. Apart from the obvious problems of duplication of effort, this has also led to a certain diversity of the way similar functions had to be performed in different applications, which is very confusing to the user and makes it difficult to remember how to drive the different programs.

Because it is so easy to write or modify programs in this environment, in the absence of any real software management the stability of the applications is never fully achieved, and maintenance is consequently very difficult.

The very limited memory available in the control room consoles meant that most of the applications had to be kept small, and as a direct consequence of this error reporting had to be kept to a minimum, as did commenting the code.

Finally the speed at which the applications run has been found to be adequate. Since no online database exists the individual programs do their own data management, and though this brings its own problems it tends to be fast. Consequently the speed is determined by that of the NODAL interpretor and that of the TITN network. As a benchmark, sending a 100 point amplitude vs time function to the accelerator takes around 1 second per hardware address, which is considered acceptable.

4.2 SPS new

There were two significant differences between the way the new SPS applications were developed as compared to the old. Firstly the overall functionality of the software needed to operate the accelerator was analysed in detail before any design was considered, and secondly the underlying data structures were completely determined before any implementation was undertaken [4]. By its very nature this kind of software development leads to software that needs little change once implemented, and results in a very stable system. The highly modular way in which the applications were designed allowed an easy and standard way of handling errors, and the error reporting is excellent.

Knowing the detailed functionality led to a high uniformity, not just at the level of the operator interface [5] but more generally in the facilities the different applications shared. As examples there is only one function editor, one dataviewer and one application that is able to send to the equipment anything from a single function to the settings for the whole machine. This has contributed greatly to the ease of use of the software, and this is enhanced by a standard online help facility describing how to

drive the applications.

Having a sound definition of the data has allowed the applications to be largely data driven, giving coherence to the different accelerator systems and allowing new systems to be integrated without writing a single word of code.

The major disadvantage of this approach is that the software has been produced specifically to operate the SPS in the various modes forseen over the next ten years. Any novel running of the accelerator during machine development sessions invariably requires new features which are very difficult, sometimes impossible, to accomodate. Up to now these problems have been overcome by exploiting the high flexibility of the old TITN system.

The speed of execution of tasks is similar to the old system, but in this case database access times and the TITN network are the determining factors. The reliability of the gateway into the TITN is not good but problems are easy to spot and rectify.

4.3 LEP

Before the construction of LEP was complete, an analysis of the software required to run the machine was made. Naturally the emphasis was put on the software needed to commission the accelerator, and for the startup of LEP the controls and equipment groups provided a suite of powerful utilities for sending settings to and acquiring data from the hardware. These utilities exist as commands on the control room consoles and provide a means of quickly making script programs to do standard or non–standard things to the accelerator. Much use of this facility has been made during the commissioning phase, and more recently by accelerator physicists during machine development sessions.

The applications used today in operations also make heavy use these utilities for accessing the hardware. While this may be convenient for the programmers it invariably introduces overheads in the execution speed. The speed is further reduced by the underlying online data organisation, since the structuring of the data does not reflect the way in which we now want to run the machine.

The development of the operational applications has not followed an integrated approach, which has brought low coherence and a very variable level of error reporting.

Uniformity across the user interface applications has been achieved to some extent. Following the standards of the SPS has ensured a look and feel of the individual applications that is liked by the operators, and most programs are now easy to use.

The operational software relies heavily on servers running at all levels of the network, from the control room Apollos down to the front end computers. While communication between these servers is normally transparent to the user it often involves passing through several bridges. If one of the bridges or servers dies it is sometimes difficult to diagnose which one, and in many cases a procedure of sequentially restarting one after the other is required.

Furthermore many applications are dependant on certain computers to be up in order to run. There are presently around 10 such critical nodes on the control room token ring, the failure of any one of which would affect operations to some extent, in many cases seriously.

These two implementation details directly affect the overall stability of the software needed to run the machine.

5. Alarms

Quite apart from the application software used to drive the accelerator, there is another area of the control system that is of great importance during routine operation of the machine, namely the surveillance system.

Ideally this should work on the simple principle that software, running without operator intervention, should check that all elements required to be ON are ON, that those that should be OFF are OFF, and that all settings stay within a tolerance acceptable for operations. This software, running frequently, should report any abnormal findings to an alarm system for processing prior to presentation to the operator as a new alarm on his screen [6].

In practice the viability of such a system depends very much on other parts of the control system. It is imperative for such a system to have available a definitive source of data reflecting the way the machine is actually supposed to run at the time. Furthermore because most machines run in several modes of operation, each requiring a different configuration, this image of the machine has to be dynamic.

It has already been mentioned that the LEP operational applications have not been developed in an integrated way, and one consequence of this is that there are several different ways of storing the actual machine settings. This makes it very difficult to provide standard surveillance programs; in reality each set of equipment has to have it's own program, a situation which is of course very difficult to administrate. So while for LEP the central alarm server works well, the amount of useful information reaching the operator is presently rather limited.

The same problems were encountered with the alarm system running on the SPS TITN network. Again there was no coherent image of the machine, and it took several years before the alarm system was providing information of sufficient credibility for the operators to use with confidence.

It is ironic that the new SPS operational software, which is driven from a central online database, does not yet have any kind of alarm system. Indeed we are experiencing problems due to this as more and more systems are migrated from the TITN to the Apollo–based software, since there is presently no means of surveying them. The aim is eventually to use the same system that is presently in use for LEP, but with simple surveillance programs comparing measurements with settings in the online database.

6. Conclusions and remarks

In the case of both the SPS and LEP, the network and control room utilities proved adequate during the running in of the machine. As testimony to this, beam was circulating in LEP one or two days after first injection, and the first Z^0 was reported within a month.

However, remember also that machine commissioning is done by specialists and over a limited period of time. When it comes to building the complex, integrated software packages that are required in routine operations, it has proved difficult to do so from the utilities provided. What is needed is a review of the operational requirements and a corresponding rewrite of the application software. Furthermore it is very difficult to determine these operational requirements in advance of getting hands–on experience of the accelerator.

The new SPS software is a good example of what can be done. It was based on 10 years experience of running the SPS in a variety of modes, and the software produced satisfies most of the operational requirements.

The same thing now has to be done for LEP, this time after 3 years experience but drawing on the lessons learned in the SPS.

7. References

[1] M.C. Crowley–Milling, "The Design of the Control System for the SPS", CERN 75–20

[2] M.C. Crowley–Milling, "Experience with the Control System for the SPS", CERN 78–09

[3[R. Rausch, "Real Time Control Networks for the LEP and SPS Accelerators", in *Control Systems for Experimental Physics*, Villars, Switzerland, September 1987, pp. 244–247.

P.G.Innocenti, "The LEP Control System", in *Accelerator and Large Experimental Physics Control Systems*, Vancouver, Canada, 1989, pp 1–5

P. Lienard, "The SPS and LEP Control Network Architaecture", in *Accelerator and Large Experimental Physics Control Systems*, Vancouver, Canada, 1989, pp 215–220.

[4] R. Bailey, J. Ulander, I.T. Wilkie, "The New Applications Software for the CERN SPS", in *Control Systems for Experimental Physics*, Villars, Switzerland, September 1987, pp. 158–163

[5] A. Ogle, J. Ulander, I.T. Wilkie, "Experience with Workstations for Accelerator Control at the CERN SPS", in *Accelerator and Large Experimental Physics Control Systems*, Vancouver, Canada, 1989, pp 437–441.

[6] M. Tyrell, "The LEP Alarm System", these proceedings.

Experience Controlling the LAMPF-PSR Accelerator Complex*

S. Schaller, R. Stuewe, E. Bjorklund, M. Burns, T. Callaway, G. Carr,
S. Cohen, M. Harrington, D. Kubicek, R. Poore, and D. Schultz
Los Alamos National Laboratory
Los Alamos, New Mexico, 87545, USA

Abstract

In recent years, control system efforts at LAMPF have emphasized the provision of uniform control for the LAMPF linear accelerator and associated beam lines and the Proton Storage Ring and its associated beam lines. The situation is complicated by the presence of several control philosophies in the operator interfaces, data base mechanisms, and front end data acquisition and control interfaces. This paper describes the current system configuration, including the distributed operator interfaces, the data and control sharing between systems, and the use of common accelerator diagnostic software tools. Successes as well as deficiencies of the present system will be discussed with an eye toward future developments. *

I. BACKGROUND

The Clinton P. Anderson Meson Physics Facility -- also known as LAMPF, the Los Alamos Meson Physics Facility -- is composed of an 800 MeV proton linac plus associated beam lines and targets, and an 800 MeV Proton Storage Ring (PSR) plus its beam lines and a neutron spallation target that serves the Los Alamos Neutron Scattering Center (LANSCE). The linac accelerates beams of H+, H-, and polarized H- (referred to as P-) ions up to 120 times per second in pulses of up to 1000 microseconds width. The average H+ beam current can be as much as 1 mA. The proton storage ring serves as a beam compressor, taking a full H- macro-pulse from the linac and ejecting it in several hundred nanoseconds.

When LAMPF was built in the 1960's it was one of the first accelerators to be designed for computerized control. Since the IEEE CAMAC standard did not exist at that time, a significant amount of effort went into the design and construction of LAMPF-specific data acquisition hardware. The system that resulted was called RICE (Remote Instrumentation and Control Equipment). RICE hardware is still used for more that 60% of the linac equipment. Over the years, the control system was expanded to include CAMAC hardware accessed on demand through remote computers. With the addition of remote operator consoles, the initial star architecture has evolved into a much more distributed configuration.

The Proton Storage Ring was designed in the early 1980's to be independent of LAMPF with a separate control room and beam lines. As a consequence, the PSR Control System was designed and implemented with only minimal consideration of LAMPF requirements. The PSR system did provide recognition of its effect on linac timing requirements and it used a device naming scheme that was similar to LAMPF's. The PSR system emphasized continuous update of a centralized database.

In 1988, responsibility for the Proton Storage Ring was transferred to LAMPF. This paper describes the present configuration of the two control systems and the attempts that have been made to integrate them in a useful manner. We conclude with a brief description of our plans for the future. More information about our plans can be found in a companion paper at this conference [1].

II. CURRENT CONFIGURATION

A. LAMPF Control System (LCS)

The evolution of the LAMPF Control System (LCS) has been described in detail elsewhere [2-4]. The LCS is currently composed of a network of VAX computers connected via an Ethernet using DECnet for communications. Computer systems in the LCS network are of two types, (Figure 1). A typical LCS operator console computer runs VMS and drives one or more LCS operator consoles. Such a computer may also have a CAMAC-based data acquisition and control capacity. A typical LCS data acquisition front-end computer runs the VAXELN real-time kernel and handles hard real-time data acquisition through CAMAC. The VAXELN nodes do not have local disks.
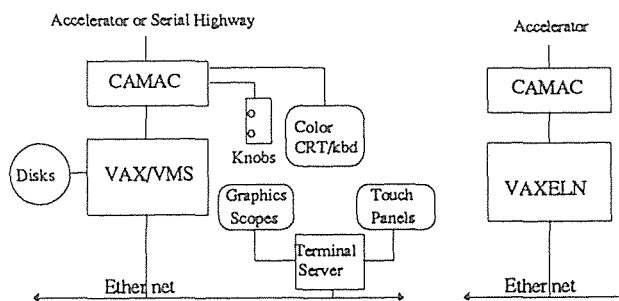

Figure 1. LAMPF Operator Interface and Front End Computers

Each LCS operator console is composed of one or more color character-cell CRTs which are shared between a number of application programs, several graphics scopes, trackball-based touch panels, and a set of analog control knobs. The graphics scopes and touch panels are attached to the computer

through terminal servers. The color CRT and knobs are attached through CAMAC.

The color CRT in the LCS operator interface allows any LCS program to be called up at the operator's demand. This interface also gives the operator access to a number of supervisory tools which allow the state of any devices to be displayed and controlled. The touch panels also allow access to a fixed number of application programs.

LCS operator consoles are now supported in the LAMPF Central Control Room (CCR), the Injector Control Room (ICR), and the LANSCE Control Room (LCR). (See Figure 4 for a geographic representation of this distributed functionality.) The main CCR control computer was the center of the original star configuration. It still maintains its central position as it drives four of the five LCS operator consoles in CCR and serves as the central repository for LCS software and databases.

Since the RICE hardware was and still is a primary feature of the LAMPF Control System, a VAXELN front-end computer is dedicated as its interface with the rest of the control system (Figure 2). The RICE system is composed of 73 hardware data acquisition modules arrayed along the linac and in the injector and experimental areas. A distinct advantage of the RICE system is that it supports simultaneous timed data takes on each RICE modules. This provides a very powerful method for acquiring longitudinal snapshots of the entire linac at a particular time on a particular beam pulse. For untimed data takes, data caching facility is provided. In addition, the RICE system interfaces with the accelerator "fast protect" system. If a hardware monitor determines that too much beam is being spilled, the fast protect hardware sends a signal that simultaneously inhibits the injector and notifies the RIU computer that a fast protect has occurred. The RIU computer immediately reads the state of all hardware monitors to determine where the fast protect occurred.

73 RICE Modules

RICE Interface Unit — Timing
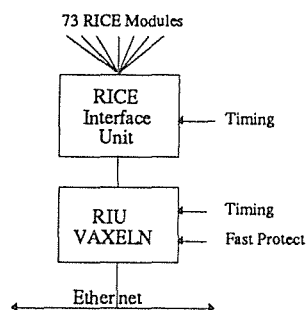
RIU VAXELN — Timing — Fast Protect

Ethernet

Figure 2. LAMPF RICE Interface Front End Computer

LCS data acquisition is demand driven. Each node in the LCS network contains a static database derived from a master database on the main CCR computer. Application programs request data and issue commands through a standard data access interface which uses the local database to resolve device addresses at run-time. Application programs may be split up among several nodes. A locally designed Remote Procedure Call (RPC) interface allows the pieces to communicate without dealing with the complications of DECnet. Programs

that know they need large amounts of data can improve their system throughput by forming "aggregate devices." The LCS data access interface makes use of information supplied by the program to optimize network usage.

Because of the uniformity of data acquisition interfaces, if the correct application programs and databases are supplied to it, any LCS operator console on the network could run the entire accelerator.

B. PSR Control System

The PSR Control System attempts to achieve high data throughput and reasonable operator interface response by tightly coupling a central database to external computers that are continuously polling data. Detailed descriptions of this system have been published elsewhere [5-6].

A diagram of the PSR operator interface and front end computers is given in Figure 3. All PSR operator consoles are attached to the PSR VAX. This machine serves as the operator interface computer and the central data concentrator. A typical PSR operator interface screen consists of a color graphics scope whose face is overlaid with a touch sensitive surface. The graphics scope is driven directly from the computer bus; the touch panel is driven through a terminal server on Ethernet. A set of analog control knobs is controlled via CAMAC. The top level screen on each color graphics scope provides an entry to a tree-structures menu of possible programs that can be started.

The PSR front-end computers are PDP-11s each connected to a CAMAC serial highway for data acquisition and control. These front-ends are know as Instrumentation Sub-Systems (ISSes). They continuously update their local databases with data from their serial highways. The ISSes are also connected to each other and the PSR VAX via a separate CAMAC serial highway over which the PSR VAX reads the latest data from the ISS databases and transfers changes in control values.

Serial Highway to ISSes   Serial Highway for this ISS

CAMAC   Knobs   Color Graphics Scope   CAMAC

Disks   PSR VAX/VMS   Touch Panel   ISS PDP-11 RSX-11

Terminal Server   Serial Highway   to other ISSes   CAMAC

Ethernet
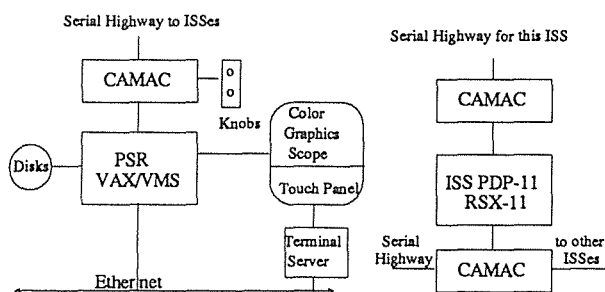
Figure 3. PSR Operator Interface and Front End Computers

Application programs running on the PSR VAX typically have exclusive access to a single graphics scope. The programs access data and set control values in the central database. They can also be notified asynchronously of changes in database values. Device data addresses in the central database are resolved at link time for many of the PSR application programs. This limits run-time overhead, but results in problems if the database structure changes.

C. Control Systems Integration

When the Proton Storage Ring was first commissioned, it was controlled entirely from LCR. LAMPF at that time had operator consoles in CCR and ICR, although most operations activities took place in CCR.

The first attempt at LCS/PSR integration was to place PSR control consoles in CCR. Since we also wished to keep LCR available for PSR beam development, we had to pull cables between the two control rooms to physically connect the remote PSR consoles with the PSR VAX. At the same time a LCS console (CPU, color CRT, graphics scope, and knobs) was installed in LCR.

We then approached the more difficult job of sharing data and controls between the two systems [7]. By adding software (mainly run-time libraries and a copy of the LCS database) and LCS console hardware, we were able to make the PSR VAX into an LCS operator interface computer. This meant that PSR application programs could access LCS data through the LCS data interface. Several PSR applications that were interested in linac data were so modified.
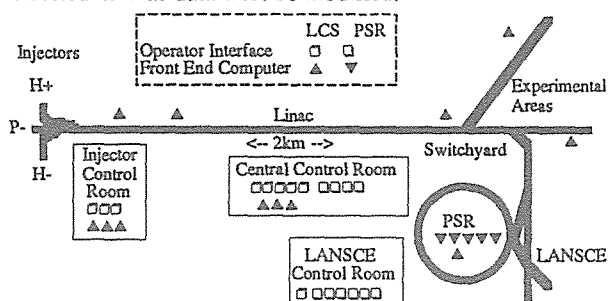
Figure 4. Distributed Functionality in the LAMPF-PSR Network

The situation with LCS programs was a bit more difficult. Since only programs residing on the PSR VAX could access the PSR database, we added an LCS Data System server process to the PSR VAX. This process (which is a variant on the standard LCS Data System server) handles network requests for information on PSR devices. Since the LCS programs make standard requests for data, all the LCS programs, including the supervisory display and control tool, the LCS knobs, and the LCS data archiver could get PSR data.

Figure 4 show the geographic distribution of LCS and PSR consoles and front-end computers. This figure does not show details of CAMAC highways, Ethernet connections, RICE cabling, or timing distribution.

III. EXPERIENCE

A. Front Ends

The RICE hardware system presents some unique problems to the control system. Since more than 60% of the LCS hardware is attached through the RICE system and is only accessible through the RIU front-end computer, this represents a major bottleneck. The fact that we have a limited amount of RICE hardware available means that we cannot easily expand the system.

The RICE interface can only perform one timed data take per beam pulse. This severely restricts tuning operations which are typically performed at low rep rates. We would like to improve on this performance even though we would like to keep the capability for doing synchronized timed data takes. Other RICE problems include not being able to send analog commands to more than one device in a RICE module at a time. This constraint creates problems when one is trying to scan wires in two wire scanners in the same RICE module.

The primary problem that we have found with the PSR ISSes is that of performance. The PSR equipment modules which reside in the ISSes are flexible and can acquire data rapidly. But the overall response is slow because data is scanned and transmitted regardless of its usefulness. The inherent flexibility becomes a problem because some modules respond differently to the standard read and command requests that are issued from the application programs. We need a carefully designed application program model of the world and disciplined equipment module implementations that ensure standard responses.

B. Databases

A database system in an experimental installation must be able to add and change device definitions quickly without taking the control systems offline. The LCS database succeeds in this respect; the PSR database does not. After a PSR device definition has been modified, it can take hours to regenerate a new PSR database. To be able to communicate with PSR devices, the LCS database must contain the PSR device names. This is possible because both systems use the same device naming scheme. Unfortunately, there is no automatic scheme for rationalizing the names that occur in the two databases. For now we use editors to compare lists of names to determine what should be a changed.

C. Application Programs

The difference in design philosophies between the two control systems is most noticeable in the application programs. The absence of a notification on change in the LCS data access interface makes it hard to allow PSR programs to access LCS data through the PSR data access mechanism. As a compromise, we have made it possible for PSR programs to access LCS data through the LCS access mechanism. On the other hand, it was relatively easy to enable LCS programs to access (possibly old) PSR data. For application programs driving a future common LCS/PSR operator interface we should like to have data from both systems provided in a uniform manner.

D. Operator Interfaces

Neither the LCS nor the PSR operator interface lend themselves to upgrade. The LCS technology is old and cannot integrate graphics and control functions. The PSR screens are becoming unmaintainable and cannot be moved to other nodes in the network because PSR applications need to access the central PSR database directly. The LCS supervisory tools allow operators to directly access any device. The PSR tools must be recompiled to allow access to new devices. We have found that the flexibility provided by the LCS interface is vital in running a basically experimental accelerator.

E. Reliability and Maintainability

Hardware reliability and maintainability has been a key issue during recent accelerator runs. The RICE hardware is getting old and becoming difficult to maintain. Replacement hardware is no longer being manufactured. While CAMAC hardware is available for replacements and additions, its use in harsh environments sometimes leads to short lifetimes.

The use of long serial CAMAC highways, especially in the PSR system, has led to difficulties in problem isolation. Frequently it is necessary to remove one crate controller/fiber optic driver at a time from the highway in order to isolate a fault. This can be a very time consuming operation, especially if it has to done during production.

We have also been concerned about single points of failure within the control system. As the systems stand now, the failure of a single LCS operator interface or front-end computer only means that the CAMAC attached to that machine is inaccessible. As mentioned above, with the correct data files and programs, any LCS operator interface computer in the network, including the PSR VAX, can run the accelerator.

The failure of the RIU front-end computer would be more serious for then we would loose all RICE data, a significant portion of the accelerator's data.

Loss of the PSR VAX would mean loss of the entire PSR system since the serial highway connections are only to that machine and the central PSR database resides on it.

IV. THE FUTURE

The concerns described in the previous sections are being dealt with through several projects being currently planned or implemented at LAMPF. The projects and several other considerations are described in detail in a companion paper at this conference [1]. In the remainder of this paper we will briefly summarize these projects.

At the lowest level, the front end data acquisition hardware will be upgraded to meet new requirements for reliability and maintainability. The plan is to use VAXELN-based micro-VAXes as the standard front end computer. These front ends will be used to replace the PDP-11s currently being used for the PSR ISS interface to CAMAC. At the same time, it is planned to replace the RICE hardware with CAMAC and again

use VAXELN front ends. The effect of these two projects will be to unify all device access through a common client-server model.

A new common operator interface project has also been proposed. A common interface will reduce both training and maintenance requirements. We plan to use VAX-based workstations as the primary operator interface. We plan to use a user interface management system to keep the development and maintenance of the operator interface more manageable.

Since the new data access mechanism will not automatically put the data in the PSR database, existing PSR application programs may have to be changed. There is the possibility that some of these programs may be rewritten to make use of the new common operator interface. In the long run, this is what we hope to do with all application programs.

To improve the overall responsiveness of the control system, we hope to pursue several hardware upgrades beyond replacing the front end computers and introducing operator interface computers. Since, for a while, some application programs will still be using the old interfaces through the LCS and PSR central control computers, we hope to replace them with higher performance VAXes which can be clustered to provide hardware and file backup for each other.

With these upgrades we will be able to respond to increased demands on the LAMPF control systems in the future. Of most immediate interest are the controls necessary to support the proposed Pion Linear Accelerator (PILAC) to be built at one of LAMPF's beam lines. There are also proposals to upgrade the PSR from 100 to 300 μA to drive a possible pulsed lepton source, and to use LAMPF for prototype work in using an accelerator to transmute radioactive waste.

V. REFERENCES

[1] R. Stuewe, S. Schaller, et. al., "Future Directions in Controlling the LAMPF-PSR Accelerator Complex," these proceedings.

[2] G. Carr, S. Schaller, et al., "The Status of the LAMPF Control System Upgrade," Proc. Europhysics Conference on Control Systems for Experimental Physics, CERN Yellow Report 90-08, (1990) p.107.

[3] S. Schaller and E. Bjorklund, "Distributed Data Access in the LAMPF Control System," Proc. 1987 IEEE Particle Accelerator Conf., Washington, DC (IEEE Publishing, New York, 1987) p. 745

[4] S. Brown, S. Schaller, et al., Proc. 2nd Int. Workshop on Accelerator Control systems (North-Holland, 1986) p. 122.

[5] P. Clout, et al., ibid., 1986, p. 116.

[6] P. Clount et al., "The Proton Storage Ring Control System," IEEE Trans. Nucl. Sci. NS-30 (1983) p. 2305.

[7] S. Schaller, "Providing Common Data Access for the LAMPF and PSR Control Systems," Nucl. Instr. and Meth., A293 (1990) p. 416.

Status Report on the Advanded Light Source Control System

S. Magyary, M. Chin, M. Fahmie, H. Lancaster, P. Molinari, A. Robb, C. Timossi and J. Young

Accelerator and Fusion Research Division
Lawrence Berkeley Laboratory
University of California
Berkeley, California 94720
U.S.A.

Abstract

This paper is a status report on the ADVANCED LIGHT SOURCE (ALS) control system. The current status, performance data, and future plans will be discussed. Manpower, scheduling, and costs issues are addressed.

I. Introduction

The ALS control system was designed around the concepts of parallel processing, high CPU and I/O bandwidth, and human-friendly interface. Figure 1 shows the system architecture and its five primary layers (for details of the system see References [1] and [2]). Layer 1, represented by the Intelligent Local Controllers (ILCs), interfaces to the accelerator hardware and communicates with Layer 2, the Collector Micro Module (CMM). Layer 3 is the Display Micro Modules (DMM) that has bus access to the CMM and in turn communicates with the operator stations (Layer 4) via serial links. The operator stations are high-performance Personal Computers that have Ethernet network (Layer 5) access to file servers and other network services.

The ALS consists of an Electron Gun, a Linac, a Linac-to-Booster line, a Booster, a Booster-to-Storage-Ring line, a Storage Ring (SR), and a number of user beamlines. The control system is currently operating the existing parts of the ALS accelerator hardware consisting of the Gun, Linac, Linac-to-Booster line, and the Booster; the Booster-to-Storage-Ring line is being implemented now. The Storage Ring accelerator hardware is under construction; completion is expected sometime during the second quarter of 1992. We will then be ready to begin commissioning the SR via the control system, both locally and from the Control Room.
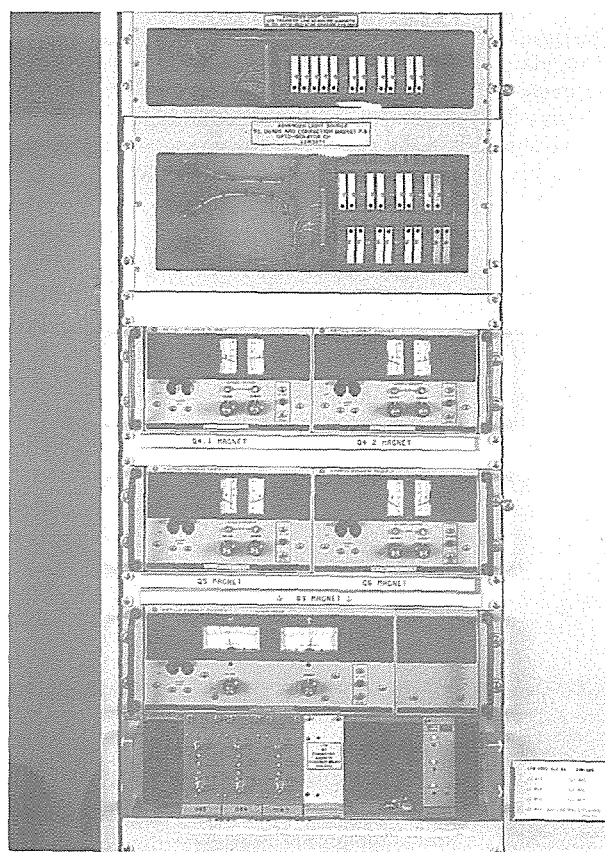
Figure 2. Typical ILC installation.

II. Layer 1 (Intelligent Local Controllers)

The ILC is an intelligent controller consisting of an 80C186 main processor, an 80C187 math co-processor, and an 80C152 serial-control processor sharing 64 Kbytes of battery backed

Figure 1. ALS control system architecture.

memory. In addition, it has on board I/O resources of four 16-bit DACs, four 13-bit ADCs, 24 bits of digital control, and an SBX bus for expansion. It is a low power (< 5 watts), 3U high Eurocard-based controller in a shielded metal can that can communicate at a 2 Mbit/sec rate using twisted pair cabling. We had commercial companies build 200 of these first generation ILCs. Twenty of them (10%) were not functioning when received from the manufacturer; ten had minor problems (chip leads bent, missing chips, infant mortality, etc.) and were repairable. The remaining ten we have not attempted to repair (they have missing traces or shorts on the circuit boards) since we decided the cost of repair was not justified. We are currently using about 140 ILCs in the accelerator, with an additional 20 to be used for the Booster-to-Storage-Ring line. See Figure 2 for a typical installation showing 3 ILCs, Opto-22 interface, and the 3 quadrupole power supplies. We expect to use an additional 500 ILCs to complete the project (Storage Ring, Beamline Front Ends), however, these will be the next generation design. These new ILCs will have 16 Mhz, 80C186EB chip as the main processor (a 60% speed improvement over the older ILCs' CPU, and also lower-power dissipation), 256 Kbytes of memory, 16-bit ADCs and DACs, 16-bit SBX interface, a serial channel, and 28 Boolean lines. The analog and digital design is completed; layout and prototyping/testing will start shortly. The cost of the current ILCs is $650 each; the new ones will be about $950. To exercise the ILCs, and test the accelerator hardware, we use laptop computers (80386/80486-based), connected directly to the ILCs, using much of the same software as we use on the operating stations in the Control Room.

Figure 3. DMM, CMM, fiber optic interface, and file server.

III. LAYER 2. (COLLECTOR MICRO MODULE)

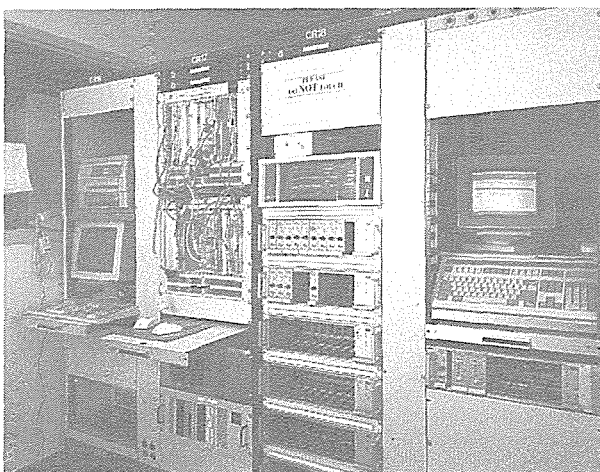The CMM (Figure 3) contains all the data gathered from all the ILCs (i.e., it represents the entire accelerator database at any moment). The ILCs are connected to the CMM via fiber-optic lines. The serial communication on these lines is bi-directional (though using only a single fiber), and the bit rate is 2 Mbit/sec. We are currently using 12 of these lines for a total I/O physical bandwidth of 24 Mbit/sec. We use a commercial (Intel) 20 MHz, 386-based Multibus I board with 4 single chip processors (via custom built SBX modules) to service 4 serial lines. Therefore, to service the 12 lines currently in use, 3 Multibus I processors

are required. These boards allow us to handle approximately 800 messages, of about 75 bytes each, per second per line. Therefore, the total I/O bandwidth currently coming into the CMM is approximately 720 Kbytes/sec (12 x 800 x 75). This represents, on the average, about a 10 refresh/sec of the entire active part of the current accelerator database. We are currently evaluating a 33 MHz, 486 Multibus I board and are designing a new SBX interface that would allow us to service 8 lines per Multibus board at about 1600 messages/line/sec. At project completion, we plan to have approximately 64 lines (using 8 Multibus I processors operating in parallel) for a total I/O bandwidth of 128 Mbits/sec, and a useful data rate 7.5 MBytes/sec. This would represent, on the average, about a 15 to 20 per second refresh rate of the active database of the entire accelerator. We are also exploring the possibility of a second CMM; this would double our I/O bandwidth to 240 Mbit/sec at a modest cost.

IV. LAYER 3. (DISPLAY MICRO MODULE)

The DMM (Figure 3) consists of a 20-slot Multibus II System, bought as a unit from Intel Corp., that has fast parallel access (via a bus converter) to the CMM. This system contains a SCSI disc controller, an Ethernet interface, and a number of high performance singleboard computers. Under current operation, we are using 2 DMMs (we had not planned to install the second DMM till Storage Ring commissioning, so we are ahead of schedule in this area) to access the database in the CMM. The DMMs currently use RMX II as the real-time operating system (we are evaluating using RMX III, a full 32-bit system) using standard Intel hardware and software. The first DMM currently supports 6 (we have tried 7) commercial (Intel) 25 MHz, 486-cpu boards, while the second DMM currently uses 3 CPUs. The CPUs within each DMM operate in parallel, and each has 2 serial links (we promised 1, so this doubles I/O performance in this area) that directly connects it to the operator station. Each of these links has the same configuration as the links between the ILCs and the CMM (i.e., 2 Mbit/sec). At project completion, we will support at least 6 operator stations per DMM, for a total of 12 or more for the whole accelerator. We have enough bus bandwidth in the CMM so we could support even a third DMM. We plan to upgrade the CPUs in the DMM with processors that will be at least 2.5 times a fast as the current ones.

V. LAYER 4. (OPERATOR STATIONS)

The operator station is the human interface of the Control System; it presents accelerator data, including real-time data, scope traces, and live video (via multi-media programs and hardware) to the operator. The operator can use mice, keyboards, or nine dynamically assignable/labeled knobs. Six operator stations make up a console (Figure 4), and each console is supported by one DMM. The operator stations use 33 MHz, 486-based AST Personal Computers (PC), but they are upgradeable to faster CPUs via a simple card swap. Most use Windows 3.0 as the operating system, a few use OS/2 1.2. The serial communication links to the DMM allow us about 800-1000 database accesses/sec for each PC in the database client/server "request" mode. In the "driven" (i.e., when the DMM drives the PC) we can achieve about 1500 messages/sec. Both

12

of these rates are CPU limited, we expect them to be about 2500 and 3000 messages/sec (at those rates we will be I/O limited) respectively at project completion. Since we plan to have at least 12 operator stations, we will attain accesses in excess of 30,000/sec; however, even at these rates, we are using only about 10% of the bus and data bandwidth available in the CMM-DMM combination. These facts demonstrate the very high performance available in our star-based, shared memory approach to control system design. The choice of the PC as our human interface has proved to be a judicious one. As the popularity of Windows grows, the availability of commercial software for use in the control system grows rapidly. We currently make extensive use of DESIGNER (graphic editing tool), EXCEL (spread-sheet), TOOLBOOK (HyperCard-like package), VISUAL BASIC, TURBO PASCAL for WINDOWS, as well as the usual languages C, C++, etc. With these PCs, we also have available the usual collection of word processors, databases, and utilities (screen capture, etc.). The use of this commercial software has greatly increased our productivity, and has allowed us to keep our staffing requirements very low. The one area where we need improvement is in support of full 32-bit modeling applications. These are currently done on workstations and can access the database via Remote Procedure Calls (RPC). We plan to shift this work to OS/2-2.0, or its equivalent, as soon as possible. In the meantime, X-Windows-based applications can appear as a window on the PCs under Windows 3.0 or OS/2, using commercial software.
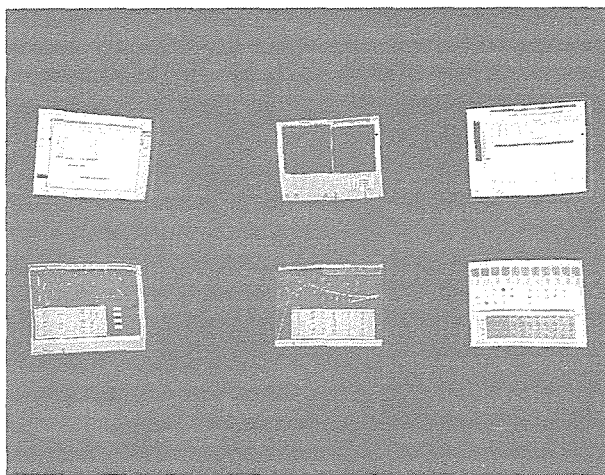
Figure 4. Operator console with six operator stations.

VI. LAYER 5. (NETWORK)

The PCs are networked via Ethernet to a Laboratory-wide network. This allows workstations, etc., to access the database via RPCs. An IBM RS6000 Workstation is used by the physicists for modeling and 32-bit numeric intensive applications. We also have a PC on this network with a 600-Mbyte disk as a file server for the operator stations and software development. A spare server is available to minimize down-time in case of failure by the main server. This network will also be the "user" interface into the control system for wiggler/undulator and beamline front end controls. A protection scheme to limit user control, to specified devices only, is under development.

A. Scheduling and Costs.

The control system is being used to commission the accelerator as it is being built. We are on target both in terms of cost and schedules, and are beginning to shift over to pre-operation. The staffing to date has been exactly as projected (with five programmers, one coordinator, and one half of an electronic designer) over the length of the construction project. Cost to date is approximately $3.65M of a total projected cost of $5M. Storage Ring commissioning is scheduled for the 2nd quarter of 1992, with project completion in 1993.

VII. ACKNOWLEDGMENT.

VIII. REFERENCES.

[1] Magyary et al., "Advanced Light Source Control System," Lawrence Berkeley Laboratory, LBL 26028 and Proc. 1989 IEEE Particle Accelerator Conf., Chicago (IEEE Publishing, New York, 1989).
[2] Magyary et al., "The Advanced Light Source Control System," Nuclear Instruments and Methods in Physics Research, A293 (1990) 36-43, North Holland.

LESSONS FROM THE SLC FOR FUTURE LC CONTROL SYSTEMS*

Rusty Humphrey

Stanford Linear Accelerator Center, Stanford University, Stanford, CA 94309

The SLC control system is the dynamic result of a number of forces. The most obvious force is the functional requirements of the SLC itself, but other forces are history, budget, people, available technology, etc. The plan of this paper is to describe the critical functional requirements of the SLC which caused significant development of the control system. I have tried to focus on functional requirements as a driver, and I will describe some solutions which we have implemented to satisfy those requirements.

The important functional requirements drivers for the control system discussed in this paper are:

⇨ Repetition rate
⇨ Sensitivity to orbit distortion
⇨ Stability/Automation
⇨ Accelerator Development

REPETITION RATE

The SLC runs for physics production at 60 or 120 Hz. At 120 Hz, 5×10^{10} particles per bunch, 3 bunches/beam pulse, and 50 GeV, the average power is 150 kW. If the beam has a small enough cross sectional area, such a beam has caused damage to beam vacuum pipes, beam vacuum flanges, collimators, or other beam line components by heating. Such events occur because the beam has become "errant"; that is, it has wandered from its nominal orbit, and is actually striking the device. If this situation is not detected, then more and more energy is put into the device, as the SLC pulses keep coming. The first issue is to detect the event, and turn off the beam. There are a number of classic methods of such detection (ion chambers, beam current comparators, etc.), and the SLC uses them.

Once the event is detected, how does one fix the problem? Usually the answer is to steer or tune the machine. But now a situation, which appears as a form of "relaxation oscillator," happens. To tune the beam, one needs beam in the machine. But because the beam is mistuned, the machine protection system detects the same problem again and turns off the beam again. How does one break this impasse?

The first, and obvious answer is to tune at a lower beam intensity; instead of running with 5×10^{10} particles, tune with 2×10^{10}. This doesn't work in general. The SLC

with 2×10^{10} particles is a sufficiently different machine from the SLC with 5×10^{10} particles that the problem often disappears at 2×10^{10}, only to reappear when the current is raised to 5×10^{10}.

The next answer is to tune at the same beam pulse intensity, but to lower the repetition rate. This is, in fact the technique that is used at the SLC. However, it does not work to simply lower the repetition rate of all components in the machine to 10 or even 1 Hz. Power is dissipated in the rf and pulsed magnet systems, and lowering the repetition rate in such components changes their characteristics. Therefore, an effective rate limiting strategy requires that the rate of running the pulsed components of the machine not be changed, but that only the injection of electrons and positrons be moved to the lower rate.

The above discussion is an overview of the simplest situation; and even it isn't really simple—how the creation and injection of positrons is handled is problematic even in this situation. More complicated scenarios are also possible in the SLC [1].

Another issue for the Machine Protection System is configuration flexibility. As the SLC configuration is changed during tuning or machine studies, the requirements on machine protection change. An obvious example is a repetition rate change from 60 to 120 Hz. A less obvious example is changing the place where the beam is stopped. It is a requirement of the machine protection system that it react to such configuration changes in as seamless a manner and as prompt as possible. At the SLC, this functionality is provided by means of the timing system, which includes distribution of timing "patterns" which allow pulse to pulse timing configuration changes. This functionality is being augmented because it is required by a project to upgrade our present Machine Protection System [2], and because it is needed for the next phase of our Fast Feedback system.

To summarize the functional requirements: The repetition rate for a linear collider can allow errant beam to damage or destroy beam line components. A protection scheme is required which detects such situations, which limits the beam, and which allows retuning of the machine to stop the situation. It is required that retuning be done at or near the beam conditions which cause the errant beam. In addition, the machine and its machine protection system must be easily and quickly reconfigurable.
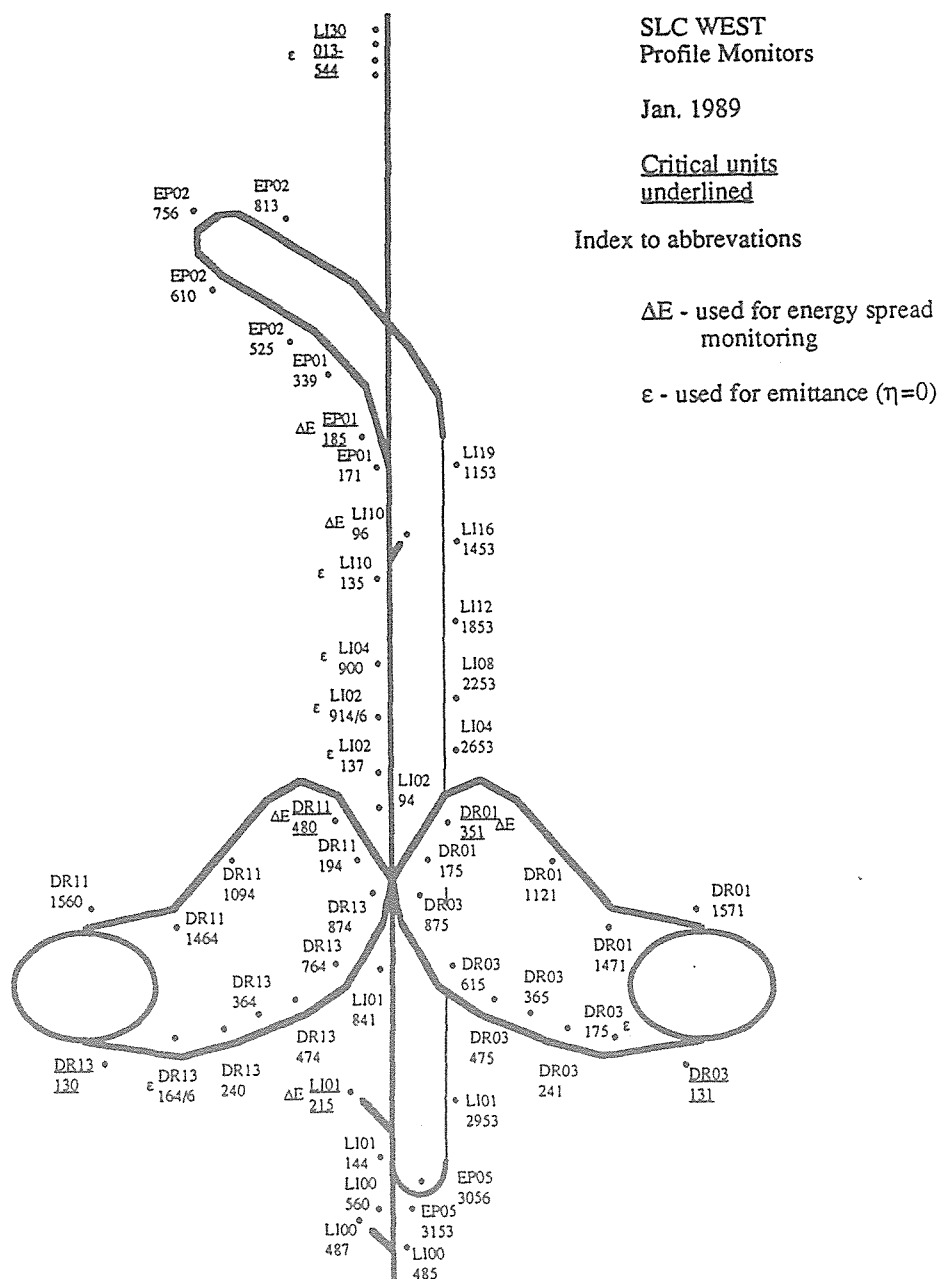
Figure 1. Location of beam profile monitors in the SLC injector, damping rings, linac and positron production systems.

SENSITIVITY TO ORBIT DISTORTION

In the SLC, emittance and other parameters of the beam are affected by orbit distortions. One easy way to understand this is to remember that wake field tails are caused by off axis beams in the linac's disk loaded wave guide. As a result of this sensitivity, the mix of beam diagnostic systems required for the SLC is affected. Diagnostics which measure beam shape, beam size, and emittance are many. As shown in Figures 1 and 2, there are approximately 100 beam profile monitors and 37 wire scanners.

The beam profile monitor system has been described elsewhere [3]. As noted there and elsewhere [4], the use of profile monitors is destructive to the beam, but they allow shape changes to be observed in real time and give detailed information of transverse tail formation. (See Figure 3.) In concert with an adjustable upstream quadrupole, beam profile monitors can be used to measure emittance [5].
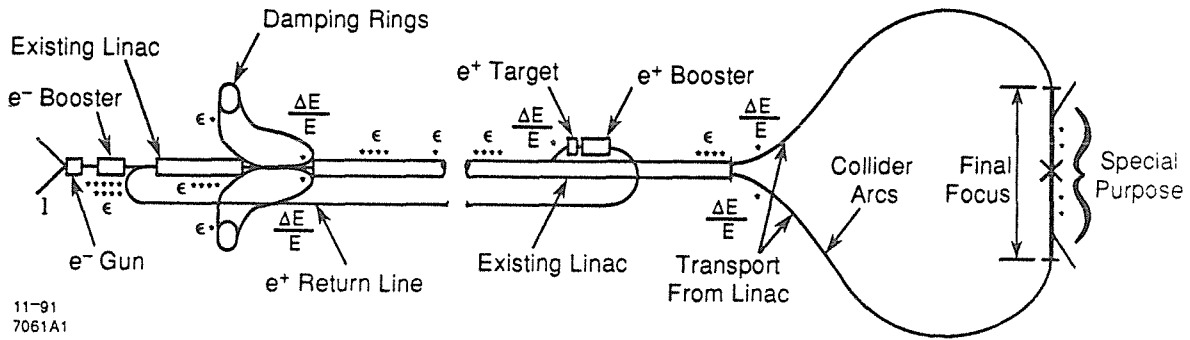
Figure 2. Location of wire scanners in the SLC.
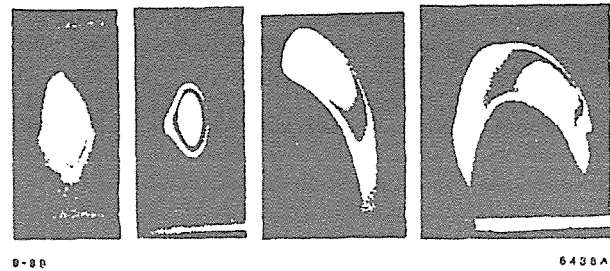
Figure 3. Images of an electron bunch on a profile monitor at 47 GeV showing wakefield growth with increasing oscillation amplitudes. The images from left to right are for a well-steered beam, a 0.2 mm oscillation, 0.5 mm oscillation and a 1.0 mm oscillation, respectively. The beam intensity is 2×10^{10} electrons. The core sizes σ_x and σ_y are about 120 mm.

The wire scanners have been discussed elsewhere [6]. The beauty of wire scanners is that they allow nondestructive measurement of the beam emittance, and thus could be used as an online device in, for example, beam feedback systems (we have not yet done so).

John Seeman has pointed out the need for what he calls "corroborating measurements." As an example of what this term means, consider the fact that emittance can be measured by both profile monitors and wire scanners. The presence of two techniques allows the results of such measurements to be compared. If the measurements are equivalent, then they corroborate (or confirm) one another. This increases the credibility of the results—an important factor in a prototype accelerator.

Beam position monitors (BPMs) used in the SLC number approximately 1700. All the BPMs in the linac itself are instrumented for single pulse data acquisition; every BPM so instrumented can be read out, under control of the timing system, on any given pulse for a particular beam bunch. BPM systems in the SLC arcs and in the damping rings have multiple BPMs which are multiplexed into a common data acquisition module; this precludes reading all the BPM inputs into one of these

modules on the same beam pulse. However, over the past year, we have had a couple of projects to "demultiplex" BPMs; that is, to instrument more BPMs in the same way as the linac BPMs so that orbit measures on a single beam pulse can be done. The builders of future linear colliders need to look carefully at the requirements for single pulse orbit measurement.

The impact of these beam diagnostic systems on the control system is large. Fundamentally, the data acquisition requirements for a linear collider correspond to that of the "first turn" for a circular collider. The ability to take a single pulse "snapshot" of the orbit, or a snapshot of many parameters associated with the beam or with individual pulsed devices is a requirement. As the references detail, emittance and beam shape measurements require sophisticated image processing and accelerator matrix manipulation and fitting. As the maps of profile monitors and wire scanners show, and as the number of BPMs implies, these systems are everywhere, and time spent on generalization and sophistication is well spent.

To restate the functional requirement: linear collider operation requires careful attention to diagnostics which measure beam orbit position and distortion, emittance, and beam shape.

STABILITY, AUTOMATION

The SLC is a large complicated device. Stability of the SLC is a large problem. Feedback systems, in which the control computer system is an active component of the feedback loop, have been operational at the SLC since 1988. Feedback based on signals derived from beam diagnostic instrumentation allows a much higher degree of control over the beams, since these data can be acquired from many sources and statistically fit. Single device tolerances could never provide this level of stability. The main application of these feedback systems is steering (launch angle and position); but feedback systems to correct energy, energy spread, and collision point are also used.

The earliest version of these was "slow feedback," with update times measured in tens of seconds; such loops are closed through the VAX mainframe which is the highest hierarchical level in the SLC control system. This was quickly augmented by prototype pulse-to-pulse feedback ("fast feedback") systems using a dedicated microprocessor based system, instrumentation, and controlled steering supplies. This prototype system was a very successful, but could only be replicated with difficulty and was difficult to maintain. We have since generalized this prototype and integrated it into the SLC control system. That generalization is propagating at a rapid rate to a large number of installations in the SLC, replacing both the prototype version of itself as well as many of the older "slow feedback" applications. This system is described in another paper being presented to this conference [7].

One of the major benefits of these fast feedback systems is the step forward in automation that they allow for accelerator operations. As described elsewhere [8], the SLC control system logs a number of different events on a continuing basis. One such class of events logged is "knob turns"; i.e., each time an operator turns a software-defined knob, that event is logged. As a result, we know that fast feedback has decreased the required intervention of operators to do knob turns by as much as 80%; fast feedback is doing the knob turning for us.

ACCELERATOR DEVELOPMENT

The SLC is the prototype for a linear collider. The SLAC staff is working to understand how a linear collider works. One of the SLC accelerator physicists has noted that "...there are more interesting accelerator physics tests being proposed each day than there is accelerator time to perform them" [4]. The environment is such that there are numerous questions to be answered and there is often the need find answers *quickly* so that the answers can be incorporated into operation. It is an essential functional requirement that the control system supply tools that allow the staff to do machine physics experiments which have never before been even considered.

The major tool—actually a set of tools—for this is the Correlation Plot Facility, described in a poster session paper of this conference [9]. This powerful software provides a set of tools for realtime online analysis, fitting, plotting, control and measurement of a large number of variables. The facility is well integrated into the SLC control system, and programs or functions which are developed for physics studies are often incorporated into operational software [10].

This functional requirement will exist for the next linear collider, since it will be built on an experience base of one—the SLC.

COMMENTS

The control system for an accelerator must satisfy many functional requirements—many more than the four described above. These four were described because SLAC's experience shows that they are, in some way, unique to the class of linear colliders.

There are other functional requirements which are common to all accelerators. And there are functional requirements which are unique to the SLC—a prototype linear collider based on the existing SLAC linac. Neither of these classes of functional requirements have been discussed, although some of the solutions described above help to meet them.

The four functional requirements described above have been a challenge which has been met by a large team of highly committed people. Some of that team is named in the references, but there are many, many more. I would like to thank Marty Breidenbach, Ewan Paterson, Nan Phinney, Marc Ross, John Seeman, and John Sheppard for recent discussions on this topic.

REFERENCES

[1] For a further discussion, see M. Ross, "Machine Protection Schemes for the SLC," 1991 Particle Accelerator Conference.

[2] S. Clark,   A. Grillo,   N. Spencer,   D. Hutchinson, J. Olsen,   D. Millsom,   G. White,   T. Gromme, S. Allison, K. Underwood, M. Zelazny, and H. Kang, "Smart Machine Protection System," *Proceedings of the International Conference on Accelerator and Large Experimental Physics Control Systems*.

[3] M. C. Ross, "Beam Diagnostics and Control for SLC," 1987 Particle Accelerator Conference.

[4] J. T. Seeman, "The Stanford Linear Collider," SLAC–PUB–5607.

[5] M. C. Ross, N. Phinney, G. Quickfall, H. Shoaee, and J. C. Sheppard, "Automated Emittance Measurements in the SLC," 1987 Particle Accelerator Conference.

[6] M. C. Ross, J. T. Seeman, E. Bong, L. Hendrickson, D. McCormick, and L. Sanchez-Chopitea, "Wire Scanners for Beam Size and Emittance Measurements at the SLC," 1991 Particle Accelerator Conference.

[7] L. Hendrickson, S. Allison, T. Gromme, T. Himel, K. Krauter, D. Nelson, F. Rouse, R. Sass, and H. Shoaee, "Generalized Fast Feedback System in the SLC," *Proceedings of the International Conference on Accelerator and Large Experimental Physics Control Systems.*

[8] N. Heinen, N. Spencer, and J. Tinsman, "Improving the Reliability in the SLC Control System," 1989 ICALEPCS Proceedings.

[9] L. Hendrickson, N. Phinney, and L. Sanchez-Chopitea, "The SLC Correlation Plot Facility," *Proceedings of the International Conference on Accelerator and Large Experimental Physics Control Systems.*

[10] For example, M. C. Ross, N. Phinney, G. Quickfall, H. Shoaee, and J. C. Sheppard, "Automated Emittance Measurements in the SLC," 1987 Particle Accelerator Conference.

Process Control for The Vivitron : the generator test set-up

J.R.Lutz, J.C.Marsaudon, R.Baumann, E.Kapps, R.Knaebel, J.Persigny.
Centre de Recherches Nucléaires, IN2P3-CNRS / Université Louis Pasteur
BP 20, F-67037 Strasbourg Cedex, France

Abstract

The VIVITRON is a 35 MV Van de Graaff tandem electrostatic accelerator under construction at the CRN since 1985. About half of the parameters are controlled by equipments which are highly stressed by their physical environment : sparks, electrostatic field, X-rays, vacuum, and gas pressure. It needs a dedicated process control system. The described control system is used since early 1991 to perform the voltage tests of the generator. It provides important information for the accelerator tuning and for the full size control under development.

I. THE VIVITRON

The Vivitron Van de Graaff tandem accelerator, under construction at the Centre de Recherches Nucléaires at Strasbourg France [1] is designed to reach a terminal voltage of 35 MV at its terminal electrode [2]. The tank (51 m long and 8.4 m in diameter) is filled with SF_6 at 8 bars. The charging system is a belt running close to the tube at a speed of 10 m/s. The column consists of a glass fibre / epoxy insulating assembly, supported by insulated epoxy posts. Seven porticos, large field-shaping shields, and discrete electrodes improve the electrical field homogeneity [6].

The expected energy will vary from 20 MeV/A for the light ions to 5 MeV/A for the heavy ions. The intensity should go from 10^{12} pps for the light ions to 10^9 pps for heavy ones.

II. THE FULL SIZE PROCESS CONTROL

A. Specific problems

The process parameters are spread over a large area. The control equipment, located at high voltage inside the accelerator tank and in the injector, are highly stressed by their physical environment : 35 MV breakdown flashes, 440 kJ stored energy, 1.7 to 10 MV/m electrostatic field, X-rays, vacuum, and SF6 gas pressure [3].

B. Architecture

A multi-level structure is implemented between the process and the operator.

Level 3 is in charge of the field equipment I/O interfaces, the handling, switching, buffering and communication of the I/O data. Some of these field equipment crates are located in-side the accelerating tank and in the injector. They are connected to level 2 by optical-fibre links crossing the 2 MV/m electrical field. At least one crate is requested for each electrical equipotential level. The small space available in some "dead sections" imposes the choice of a small-scale bus crate.

Level 2 is located outside the vessel at ground level. It achieves communication with level 3 and with level 1 and provides data switching, concentration and handling.

Level 1 includes the communication interface, the real time control and the operator interface.

III. THE SET-UP FOR THE GENERATOR TESTS

For the generator tests, only a reduced process control is needed. No beam control is necessary, no automation is required, the number of parameters is limited and no equipment crate is needed inside the machine. All the information is fed out at both ends of the tank. Optical wires are used for sensors and activators at high voltage. Shielded and protected galvanic wires are used for those at the ground level. Thus, data acquisition, data switching and operator interface are dominant.
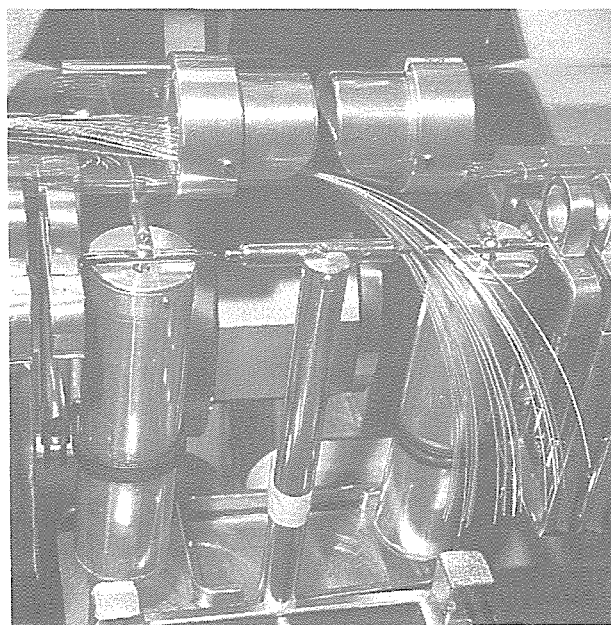
A. Current flow and terminal voltage

Figure 1. The current monitors in a dead section.

The most important features for the generator tests are the control of the terminal voltage and the current flow inside the accelerator tank. The GVMs (Generating Volt Meter), which register the electric field along the tank in a standard electrostatic accelerator, are inefficient because the internal electrode structure shields the field in the Vivitron.

Therefore, no beam being available, the terminal voltage is given by the sum of all the inter-electrode voltages. The only way to determine them is to measure the current flow along the calibrated resistor chains in each section. We developed therefore a powerless floating current monitor for the Vivitron (Figure 1), starting from an original Munich design [4]. The information is fed to the outside ground level by a plastic optical fibre. The measurement range extends from 1 µA to 1 mA with an accuracy of about ±1%.

B. Set-up

This reduced control system has been designed on the full scale control scheme but with some limitations (Figure 2).
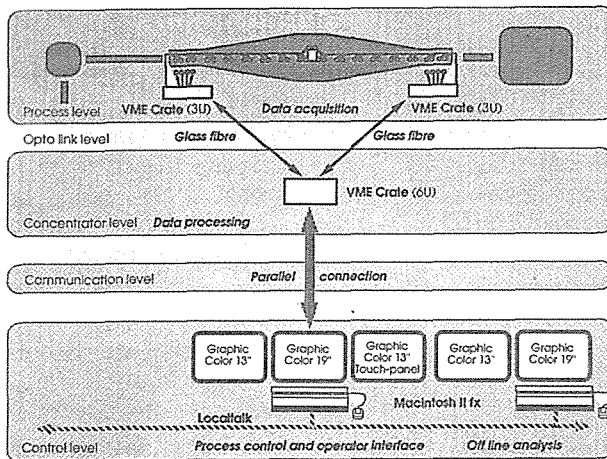
Figure 2. The control set-up for the generator tests.

Data acquisition and control is achieved by real time crates on levels 2 and 3, model of the final ones. Two diskless downloaded 3U VME crates on level 3, one at each end of the machine, are equipped with opto-isolated TTL I/O, 12 bits analog I/O, and timer/sequencer I/O boards. They communicate with level 2 by two serial glass fibre links. A 6U VME crate with hard disk on level 2 is in charge of the downloading of the former level 3 crates, of the message switching and of the data managing. The VME crates run under the OS9 real time operating system.

One standard, cheap, off-the-shelf available Macintosh IIfx 4/40 computer provides process control on level 1. It also provides the operator interface with multi-screen displays on one main 19' high resolution color screen (Figure 3), one 13' touch sensitive color screen (Figure 4) and one standard 13'

color screen (Figure 5). A second identical computer near the first one is dedicated to the off-line data analysis (Figure 6) and the back-up. A third identical computer is devoted to the software development and to the process data base management. They are linked together by the Localtalk/Appletalk LAN. A Fastpath bridge provides access to a SUN server via Ethernet. They all run under MacOS. Communication between level 1 and level 2 is made by a NuBUS - VME Micron fast parallel interface. The two operating systems share a VME mailbox.

C. Tests of the generator

The tests of the generator started in the early 1991. A terminal voltage of 17.5 MV, the half of its nominal value, has been reached within a few weeks. Some tens of sparks occurred without disturbance of the process control.

About 100 parameters are controlled. Unidirectional and bidirectional currents are measured by about 60 current monitors and displayed as bar graphs on the central main permanent color screen (Figure 3). The column currents are displayed in real time as well as their differences in order to monitor the currents distribution and to locate current leaks. The belt charging currents and balance are also displayed graphically in real time. Intershield and terminal voltages are determined and displayed numerically on the same screen [7].
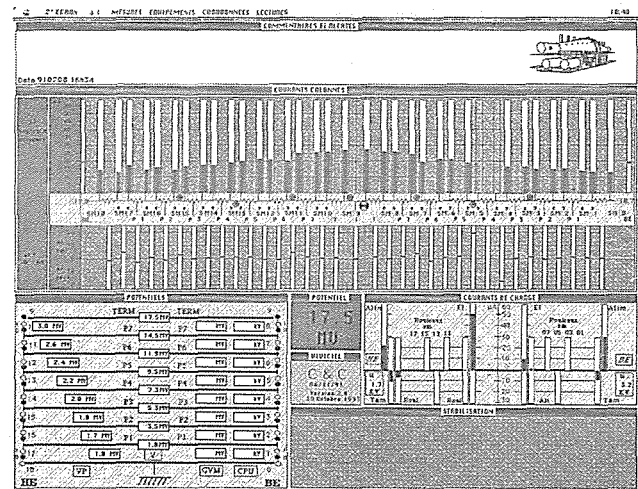
Figure 3. The operator main control display.

The control is achieved by the right hand touch sensitive color screen (Figure 4). Virtual push buttons switch the drive motors and the high voltage power supplies ON and OFF. They also control sliders which drive the charging systems. Interlocks are provided for security. The status is displayed by virtual green, yellow or red lights.

The left hand color screen is devoted to all the other functions : schematic display, wiring display, off-line and on-line
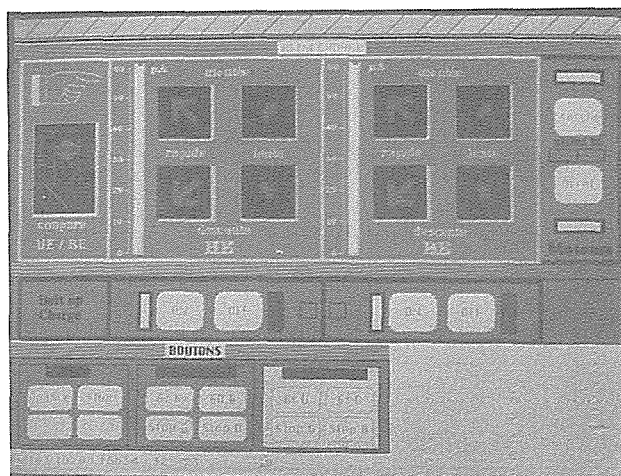
Figure 4. The command screen.

Figure 6. The 3D time / position current display.

Figure 5. The real time current graph.

numerical data display, time chart of any selected parameters (Figure 5).

The refresh rate, of about 500 ms, depends on the computer load. All the data measured for 100 seconds are stored in a file with date and time on an operator command or automatically before and after a flash for later use and analysis (Figure 6). Every hour also, all the parameters are automatically stored on a backlog file.

The off-line analysis of the machine behavior and the study of the sparks can be achieved in two ways. The playback of on-line recorded data files by the control program performs continuous real time replay or step by step reading. The second way makes use of specific data analyzing programs or standard spreadsheet and graphic representation of data like EXCEL or WINGZ. A MACRO Language or Hyperscript are useful in this case. Figure 6 represents the currents flow inside the volume of the tank during a spark.

IV. THE FUTURE

Equipment crates on level 3 will migrate inside the accelerator tank. The possibility to fit equipment crates inside the tank and their reliability have been tested in the MP accelerator. After more than one year and hundreds of flashes of up to 17 MV, we registered no transmission failure, no optical fibre failure, no measurement disturbance during flashes and only a few electronic failures [5]. The number of these crates will increase to more than ten.

The concentrator crates on level 2 will be connected to a high speed LAN and their number will grow. The process control and operator interface will be achieved by standard workstation clusters connected to the LAN. Processing power will grow from level 1 to levels 2 and 3. Software has to be more flexible.

V. CONCLUSION

The process control of the Vivitron had to be done in two steps. The first one has been used for the generator tests since early 1991. It represents a fast available control but it is reduced and limited in performance. Nevertheless, good experience is gained for the second step which concerns the full size control. Our aim is to perform a smooth transfer from the present status to the final one whereas the generator tests are going on.

VI. REFERENCES

[1] F.Haas, "Applications de techniques nouvelles à la construction des machines électrostatiques. Un exemple : le Vivitron", *Rev. Phys. Ap.* 23 (1988)1431

[2] M. Letournel and the Vivitron group, "The Strasbourg project. A 35 MV VIVITRON Tandem." *I E E E Catalog* n° 87CH2387-9, Vol.1, p.346

[3] J.R.Lutz, J.C.Marsaudon, R.Baumann, G.Frick, R.Knaebel,

C.Muller, J.A.Persigny, G.Rech, D.Stadelmann, "Contrôle et Commande du Vivitron ", *Rapport d'Activité 1987 Centre de Recherches Nucléaires,* F- Strasbourg

[4] L.Rohrer and H.Schnitter, "A light link coupled current monitor", *Rev. Phys. Ap.* 12 (1977) 1415

[5] J.R.Lutz, J.C.Marsaudon, R.Baumann, G.Biechel, E.Kapps, R.Knaebel, C.Muller, J.A.Persigny, G.Rech, D.Stadelmann, "MP tests for the Vivitron process control", *5th International Conference on Electrostatic Accelerators and associated Boosters,* 24-30 May 1989, F-Strasbourg - D-Heidelberg

[6] J.R.Lutz, J.C.Marsaudon, "The Vivitron process control", *International Conference on Accelerators and Large Experimental Physics Control Systems,* October 30 - November 3, 1989, Vancouver, B.C, Canada

[7] J.R.Lutz, J.C.Marsaudon, E.Kapps, J.A.Persigny, "The Vivitron process control", *2nd European Particle Accelerator Conference (EPAC),* June 12-16 , 1990, Nice, France

Recent Developments of the ALPI Control System.

G. Bassato, A. Battistella, M. Bellato, S. Canella

I. N. F. N. Laboratori Nazionali di Legnaro

Via Romea, 4 - 35020 Legnaro (Padova) - Italy

Abstract

This paper presents recent developments of the control system for ALPI, the new superconducting linear accelerator that will begin to operate at L.N.L. next year.

Both hardware and software architectures are described and some base choices are discussed. Results of tests performed in the last two years are also reported.

I. INTRODUCTION

ALPI is the name of a new heavy ion linear accelerator to be used as booster of the 16 MV XTU Tandem at L.N.L. The accelerator will consist of 93 quarter wave superconducting resonators with β varying from 0.055 to 0.15 and whose resonance frequencies are 80 and 160 MHz.

The installation of a first set of 48 cavities at 160 MHz and with $\beta= 0.11$ is now in progress and it is expected to be completed within the beginning of next year.

The components of the LINAC that need to be controlled may be grouped according the following scheme:

- beam pulsing system
- resonators and related devices
- cryogenic system
- vacuum system
- magnets
- beam diagnostic system.

The control system components that directly interact with the accelerator hardware are mostly supervised by embedded controllers. For instance the helium refrigerator and the magnet power supplies were bought with their own controllers; for other critical functions, such as resonators amplitude and phase lock, microprocessor based units were developed at L.N.L. In this way feedback loops, where necessary, are closed locally and the time constrains for the higher levels of the control system, where general purpose computers are involved, are much less tight.

The control system for ALPI has to supply the following capabilities:

- man-machine interface: a set of utilities to monitor and set all the operational parameters of the accelerators components

- data base management, that is the access to a centralized sets of calibration tables, set points, pre-configured machine setups
- recording, as the system must periodically dump the machine status and collect, display and record exceptional events (alarms)
- computer communications, as the system includes a number of computers which are connected through a high speed network.

II. SYSTEM OVERVIEW

From the hardware point of view the ALPI control system has a three level, network based, architecture (see figure 1).

At the first level there are some workstations (DECstation 5000/200), supplying the man-machine graphical interface. They will be mainly placed in the accelerator control room; for maintenance and test purposes some graphic computers or X-terminals will be distributed also in the LINAC room.

The second level of the system is made of network concentrators, which are VME machines, based on Motorola 68030 microprocessor. In the base configuration of these computers there is a Motorola MVME147 CPU board comprehensive of 8Mb RAM, an Ethernet controller and a SCSI controller, plus a 150Mb hard disk, 3 serial communication boards (each with 8 RS232 ports) and a set of 4 boards for analog and digital I/O.

The main functions of network concentrators are to read and send back commands and data from workstations to the low level embedded controllers. Besides, through analog and digital I/O boards, they directly supervise those sensors and actuators for which no intelligent controller is available. So, at the moment, network concentrators mostly work as configurable routers, but in the near future more complex capabilities are planned to be added to them:

- execution of pre-defined sequences of actions (resonators tuning, resonators amplitude and phase lock, ...)
- data recording (towards a network file server)
- supervising of selected subsystems (liquid helium distribution to cryostat tanks, resonators baking, ...).
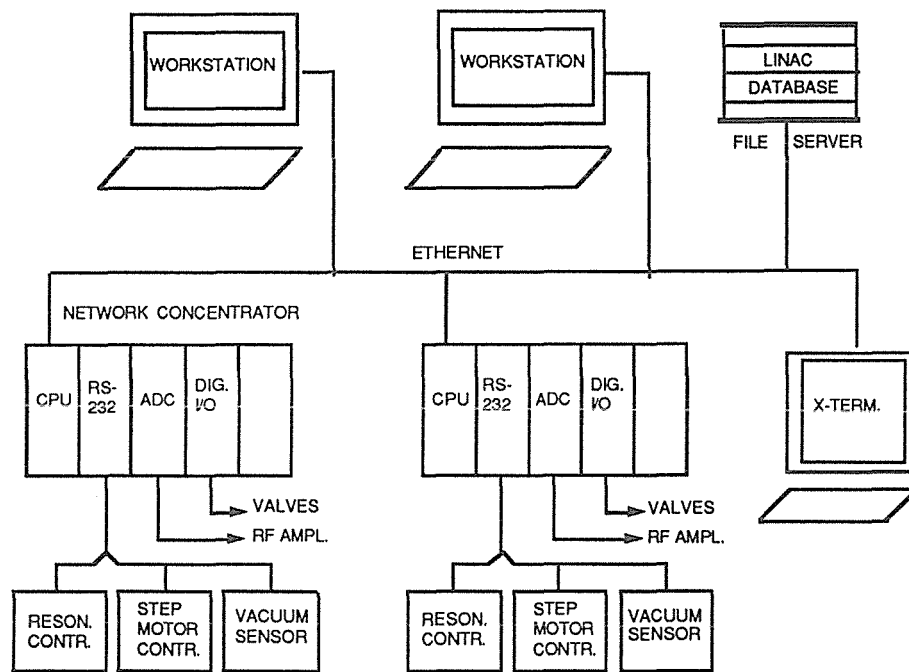
Figure 1. The hardware architecture

Workstations and network concentrators are connected through an Ethernet link.

At the third level of the control system there are microprocessor based units, completely dedicated to specific tasks. They are connected to the network nodes by standard RS232 serial lines.

III. SOFTWARE OVERVIEW

The goals in software design and development for the ALPI control system are:

- a simple man-machine interface, to be used even by people not trained on computers
- a fast response to the operator's commands
- a distributed control capability, to get the best performances from the hardware architecture
- standardization, maintenability, hardware independence together with a sufficient flexibility and a relatively short development period
- reliability.

To reach these goals a standard development environment providing graphics and networking tools has to be used.

Standardization, maintenability and portability from one side, flexibility and a short development period from the other side may be achieved using C-language in an UNIX environment, together with X-Windows graphics and socket based interprocess communications.

Therefore, the base choices for ALPI software are:
- UNIX
- X-Windows
- sockets with internet addressing and TCP protocol.

UNIX is used both on the graphic workstations (Ultrix 4.2) and on most network concentrators (System V).

Concerning with the graphics, besides X11 and Toolkit libraries, Athena Widget library (a collection of built-in high level graphic objects) is also used.

An important goal of the software design and implementation is to have a quick response to the operator's commands, that is the system must reply to most commands in some tenths of a second. This goal is achieved by using stream sockets and a client-server paradigm for networking and interprocess communications.

To reach a sufficient degree of reliability the software under test is given to the final users as soon as possible: in this way an immediate feedback is available and most bugs are early reported and fixed.

IV. SOFTWARE ARCHITECTURE

Software architecture includes a set of distributed processes running on the graphic workstations and on the network concentrators which read data from ASCII files
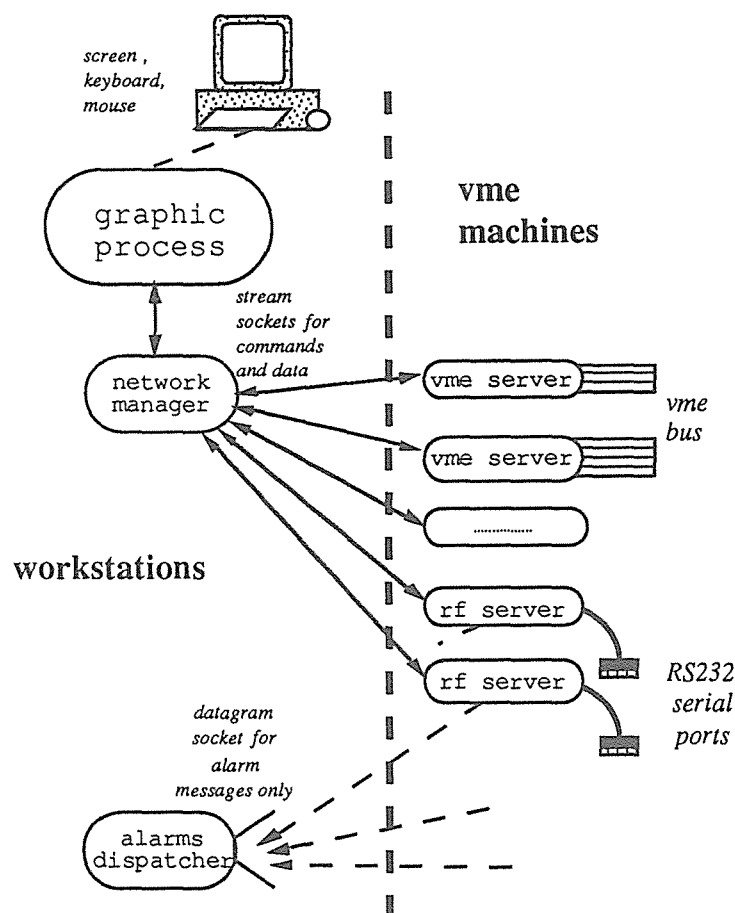
Figure 2. Software architecture: Network and Interprocess communications.

containing calibration tables, default values and network configuration data.

As shown in figure 2 there are four kinds of processes:

- a graphic process (man-machine interface)

- a network manager, which acts as a network configurator at start time (on the basis of a configuration data file), and as a router at run time

- network servers, which directly manage low level devices

- an alarm dispatcher, which has to collect and dispatch to the operator alarm messages coming from network servers.

The graphic process, the network manager and the alarm dispatcher run on workstations, while network servers run on network concentrators.

The man-machine interface is hierarchic: from a main menu the different control subsystems may be selected and each of them appears with a general page that summarizes the subsystem status (i.e. for the RF subsystem, the status of all resonators is displayed). Then, more detailed informations may be requested about a group of devices (i.e.: cryostat selection) , or about a single component (i.e.: resonator selection). Generally, it is possible to move both vertically (from a page to a more detailed one and back) and horizontally (from a group of devices to an other of the same class, from a single device to an other of the same class).

Virtual channels for interprocess and network communications are stream sockets for commands and data and datagram sockets for warning and alarm messages. Communication software is based on client-server paradigm. Internet domain and TCP/IP protocol are used.

There is a difference characterizing the control system for diagnostics: on the network concentrator a real time kernel (VxWorks) is loaded instead of UNIX. This is due to tight

constrains on the response time in beam profile data acquisition.

As this kernel configuration as far as application processes may be easily downloaded through the ethernet network to the VME machines, it is planned to bring all network concentrators to run Vxworks instead of UNIX, so eliminating hard disks from them. This will increase the system reliability.

V. CONCLUSIONS

In July 1990 a prototype of RF control system was used for a beam test on a cryostat containing four resonators. As test results were satisfactory both from the hardware and the software point of view, no substantial change was made on the base choices. The RF control package is now completed. Last summer the software modules for helium distribution control system were developed. They are now operating for the tests of the refrigerator and of helium distribution lines.

In the meanwhile the alarm management package was designed and some successful tests were performed.

Most of ALPI control system software is planned to be completed within 1992.

VI. REFERENCES

[1] I. Ben-Zvi, M. Birk, J. Brennan, C. Broude, G. Glitz, M. Sidi, J. Sokolowski: The control and electronics of a superconducting booster module N.I.M. A245, 1986, 1-12

[2] I. Ben-Zvi, Superconducting Linacs used with Tandems N.I.M. A220, 1984, 177-185

[3] J. R. Delayen, G.J. Dick and J.E. Mercereau, IEEE Trans. Nucl. Sci. NS-24 (1977) 1759

[4] D. W. Storm et al.: Status and operating experience with the University of Washington superconducting booster Linac N.I.M. A287, 1990, 247-252

The GSI Control System

U. Krause, V. Schaa, R. Steiner

Gesellschaft für Schwerionenforschung mbH, D 6100 Darmstadt, Germany

Abstract

The GSI accelerator facility consists of an old linac and two modern machines, a synchrotron and a storage ring. It is operated from one control room. Only three operators at a time have to keep it running with only little assistance from machine specialists in daytime. So the control tools must provide a high degree of abstraction and modeling to relieve the operators from details on the device level. The program structures to achieve this are described in this paper. A coarse overview of the control architecture is given.

I.   THE GSI ACCELERATOR FACILITY

At GSI the heavy ion linac UNILAC runs successfully for 16 years. It produces beams of all elements up to uranium with energies between 1.4 and 20 MeV/u. In '89 the heavy ion synchrotron SIS and the experimental storage ring ESR started operation with ion energies up to 2 GeV/u.

A new control system has been designed for SIS and ESR which is adopted step by step also for the UNILAC.

The system has to handle quite different facilities,
— the UNILAC with 3 injectors and repetition rates of 25...100 Hz,
— the SIS with repetition rates of 0.1...3 Hz,
— the ESR with repetition rates of $\leq 0.001...0.1$ Hz,
— transfer lines,
but has to provide an uniform operator access to all accelerators from one control room.

When the upgrading of the old components will be completed a total of 3000 devices scattered over an area of 250 × 250 meters have to be controlled. Complexity varies from simple DC magnets to ramped magnets (ramps generated from up to 900 samples) and diagnostic devices (up to 40kB of data). Fast switching of the machines allow time sharing between several experiments and injection to the next accelerator in sequence, all with independent beams.

II.   OPERATING REQUIREMENTS

There is a need for several identical consoles for independent operation of accelerator segments and hardware redundancy.

A consistent "look and feel" has to be provided for the operator interface of all application programs.

To keep the control system serviceable and extensible it has to be a modular and uniform system with definite allocation of tasks. The size of the facilities suggests a decentralized, distributed and hierarchical system. Therefore real–time aspects are limited to the device handling level, data for these operations are stored there. By this means a clear separation between accelerator and device oriented aspects is achieved.

The operating level ("physics of accelerators"), located on powerful workstations, deals with the equipment as a whole and handles devices in a modeled form only. The device level ("physics of devices"), located mainly on microcomputers close to the devices, maintains and controls single devices. Both levels are connected by standardized device accesses using identical mechanisms for all devices.

III.   CONTROLS COMPONENTS

On the operating level VAX 3100-VMS graphic workstations are installed forming a VAX cluster. Uniform software all over the system is ensured by cluster–disks. One to three VAXes are grouped as consoles for operator interaction.

The device level is equipped with two types of 68020 VME boards with a real time kernel. One is used as Master Processor (MP) for command evaluation. The other is equipped with an interchangeable communication interface and is used as Equipment Controller (EC) for device control and as Communication Processor (CP) for networking.

For communication between operating and device nodes ETHERNET is used. Devices are connected via modified MIL-STD-1553B serial bus with standardized InterFace Boards (IFB) as part of the devices. Besides this other interfacing like IEEE-488 is partly used by simply replacing the communication part of the EC boards. Typically 5 to 10 devices are driven by one EC.

Time signals for process synchronization are broadcasted from a central timing system to all ECs via serial bus (MIL-STD-1553B) and Timing InterFaces (TIF).

Each VME node is equipped with CP, MP, TIF and 1...9 ECs. A diagram of the components is shown in fig. 1.

Actually 33 operating computers are used in the main control room and in several local consoles close to experi-
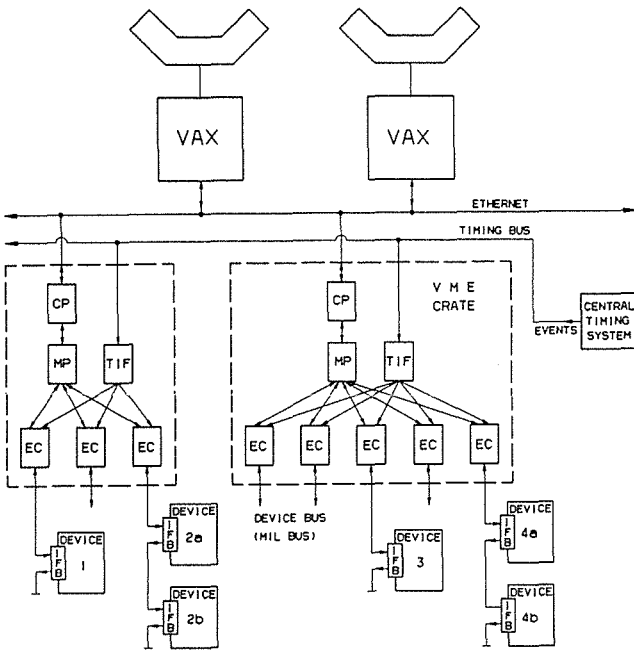
Figure 1: Hardware structure of the control system

mental facilities. For device controlling about 40 VME nodes with about 150 ECs are installed, driving more than 1200 devices.

IV. OPERATOR INTERFACE

For the variety of possible interactions with the control system, handling machines with a quite diverse nature, an operator console and operator interface has been defined combining every aspect of the operating modes.

The operator interface must offer a model of the process to be controlled. The process varies with the selected segment of the accelerator environment (SIS, ESR, UNILAC or transfer beamlines). Three software layers are present to support the selection mechanism to form the operating environment.

Layer 1 mainly consists of a code for selection of an accelerator segment. All computers forming a console are dedicated to the selected segment. Only codes which are usable in this context are provided, therefore all application, utility and service codes are grouped into categories. These categories are, e.g. usable at all consoles and accelerators, specific to an accelerator, to a beamline, a beamline segment or an experimental section. All specific codes serve only devices defined in the selected segment.

Additional codes are server for access to common databases for definition of application codes, accelerator and beamline segments, device names, properties, network addresses, error codes and help topics. Likewise alarm and error handling tasks and a logging facility for messages concerning starting and stopping tasks, abnormal conditions, failure of devices, are available in the console environment.

The second layer consists of two main process control codes, one for UNILAC and all beamlines, the other for ESR and SIS (SD[1], ZV).

— These application programs organize the access to devices using a number of specialized tasks. Each individual task uses the same context (i.e. same machine cycle, device setting etc.) and shares all process data.

— The acceptance of commands is acknowledged immediately, processed directly or passed to a subtask. All windows are organized by the central task and divided into areas with constant display of information and areas which are used by subtasks concurrently.

— The actual status of all devices in the selected segment is visible at a glance: device alarm (status or values changed) and appropriate refresh cycles guarantee an actual state of display. Icons reflect the current status of the device (switched on or off, positioned in beam or out, active or inactive etc.). Icons are unique for every device type. Additional windows are used for display of magnet read–outs showing deviations between actual and reference setting (see fig. 2), beam current profile and spill signals. History recordings offer an easy help to find a faulty element in case of beam–loss.
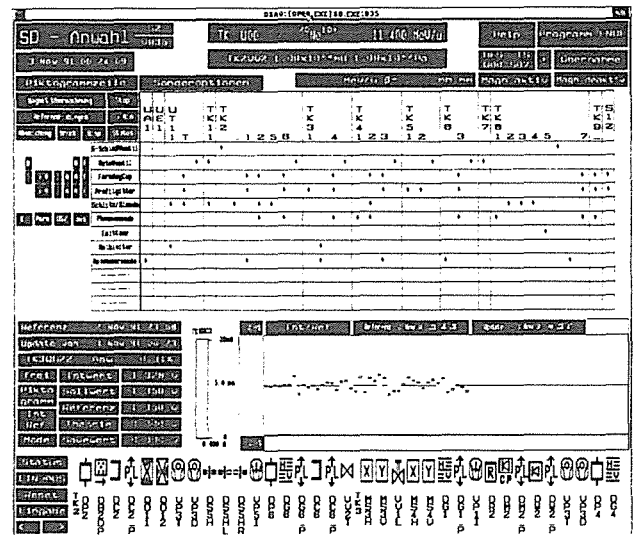
Figure 2: Process control for beamline

Additional codes of this layer are utility tasks, e.g. save magnet settings and restore saved or theoretical sets due to actual physical parameters, codes for manipulation of synchrotron devices according to algorithms, models and parameters, and for management and monitoring of the accelerator cycles. For supervision of the console environment (which task is running, on which node, network traffic etc.) a number of tasks is at hand.

The third layer consists of a number of specialized tasks which are managed by the main process control codes. Two groups are discernible, first codes used for supervisory control and update of device information. These tasks do not need output devices, servicing purely all other, sharing the

28

same process data structures. The second group consists of interactive tasks like emittance measurement, isotope and charge spectra, beam profile display, etc. needing input/output devices as there are graphics, knobs, beam current control monitors.

All application programs provide a consistent "look and feel" by a standard color scheme to present device status and action. There is an unique screen layout management for programs on graphic screens and terminals. Selection fields performing similar actions are standardized in colors, position and names. An uniform access mechanism for all devices and a transparent handling of processes and functions guarantees that the operator does not need the knowledge about the actual control sequences.

Beside the operating environment, for troubleshooting, debugging, test and running–in NODAL [2] is widely used by maintenance people.

V. DEVICE MODELING

To ease handling of the big number of devices they are classified by types (DC magnet, pulsed magnet, beam position monitor, rf–structure, ...). All devices of the same type are represented identically to the operating level and use the same software on the device level. Different device parameters like maximum current of power supplies or number of wires of a position harp are considered by device specific constants in databases on device level.

The set of relevant features of one device type is called Equipment Model (EM) of which actually 42 are supported. They are modeled and described in a standardized way to allow the same access mechanism for all devices.

Every device is identified by an unique name of 1 to 8 alphanumerics, called nomenclature. Each feature of a device of every EM is represented by so–called property and property class. The properties are described by a name of 1 to 8 alphanumerics, the property class as designation of data exchange (R read from device, W write to device, N no data exchange), and a description of the data associated with the property: the number of data to exchange and their representation (INTEGER, REAL, BITSET, ...).

For a magnet the feature "set value of output current" would be described as

property:	CURRENTS (S indicates set value)
property class:	W (data send to device)
data count:	1
data type:	REAL

Properties are EM dependent, but standard ones like mains switch (POWER), device status bit pattern (STATUS), reset (RESET), etc. are identical for all devices.

Accessing devices from the operating level means simply calling the associated property/property class with exchange of data for a device represented by a nomenclature. For greater flexibility several ways of calling are supported like execution with or without response (at least acknowledge) from the device level, synchronous (wait for response) or asynchronous (response may be read later), automatic command execution periodically, etc.

Before sending a command to the appropriate controllers the operating representation is changed to the device representation. This includes the transformation of the device nomenclature to addresses used for identification on the device level, property/property class to the identifier of the associated software module (see section VII.) and data from operating to device format since both may be different, i.e. REAL on operating level, INTEGER on device level.

Data in the responses to a command are converted from device to operating representation and then the response is supplied to the user or buffered for later reading.

Device accessing is handled by one universal software module, using the same mechanism for all devices. All EM and device informations needed are extracted from a Central Data Base (CDB).
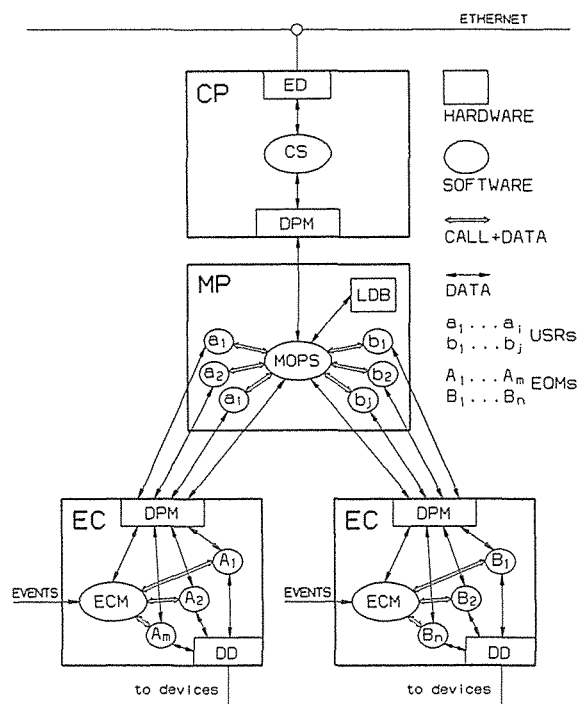
Figure 3: Software structure on the device level. ED: ETHERNET driver, DD: device bus driver, DPM: dual ported memory, LDB: local data base

VI. DEVICE CONTROLLER LEVEL

Device controlling in the closer sense of generating the command sequences for remote control is realized on the EC only. This is the only level where real time requirements have to be fulfilled. To enhance the power and to facilitate fast reaction with well defined response time the task of the ECs is limited to device driving only without interruption by higher levels. All data needed by the device are stored on the EC and it runs completely autonomously. Communication with higher levels is by dual ported memory on the EC and thus may be carried out asynchronously.

On the EC is coded *what* to do. Timing requirements (*when* to do) are managed by the central timing system which delivers 255 different time signals, so-called events, to all ECs.

Activities are coded in EM specific EQuipment Modules (EQM) as the smallest independently executable units on ECs. They may contain complex actions consisting of several device commands. EQMs are connected to events with an identical connection for all devices of an EM. Execution of EQMs is on reception of the associated event. This guarantees systemwide synchronous operation of otherwise independent devices if only connected to the same event.

Besides event driven EQMs other ones may be called by direct request from higher levels signaled in the ECs dual ported memory. Execution is not immediate but at command events always delivered in the gaps between accelerating cycles when real time reaction is not required. For pure DC devices no events are evaluated, all actions are executed directly when initiated by operating request.

EQMs cannot be interrupted so execution on the EC is strictly serial. If an EQM could not be started in time since a previously started one is still running, an error is signaled allowing EM dependent handling.

Event reception and calling of EQMs is done by the Equipment Control Monitor ECM, installed on every EC. Furthermore it supervises the status of the EC and the connected devices.

Process control by the timing system enables an easy way of fast switching between different accelerator cycles. To do so for every device 16 datasets for different accelerating cycles are stored on the EC. Selection and thus establishing the generated cycle is by a dataset number as part of the delivered event code. So the timing system determines the sequence of accelerating cycles with switching from pulse to pulse. This allows alternative supply of several experiments, each with different beams, including the injection to the next accelerator in sequence.

For a detailed description see reference [3].

VII. COMMAND LEVEL

The task of all components between the operating and the EC (implemented in hardware as well as in software) is to link the operational representation of devices to the device handling on an EC. ECs are linked to the operating by the MP to not burden the EC. The additional communication processor CP only manages the ETHERNET handling and is mainly transparent.

The MP has to evaluate commands from the operating level and to send back the responses.

EM specific command evaluation is coded in modules, called User Service Routines (USR). Every existing combination of property/property class/EM is represented by one USR on every MP where the EM is realized. Calling a property/property class simply results in execution of the associated USR. An USR may do any combination of data processing like evaluation of measurements, sending data to EC or receiving data from it, calling an EQM or another USR and performing consistency checks of exchanged data.

USRs are called by the Master processor OPerating System (MOPS). Since commands are sent in standardized form, analysis, dispatching and sending results to the operating level is the same for all devices and can be handled completely within MOPS. The handling of connected commands, i.e. periodically calling an USR, is implemented in MOPS, too. MOPS supervises and displays the status of MP and assigned ECs. Extensive multitasking allows quasi-parallel execution of commands on the MP level.

VIII. AUTOCONFIGURATION

Nomenclatures, device numbers and property description are stored on device level in local data bases, one per MP. From these databases the CDB, needed for device accessing, is automatically updated:
— At startup of an operating VAX it collects the local databases of all MPs to form the CDB,
— At startup of a MP the local database is sent to all control VAXes for update of the CDB.

A hierarchical still-alive supervision is implemented on every level of the controls system.

IX. CONCLUSION

The control system is in use since commissioning of the new facilities three years ago. During this period no severe problems were encountered. Routine operation of the machines now shows a reliable system. This experience confirms the structural concepts of the system being flexible enough to implement additional requirements and features.

Major effort now is to upgrade the old facilities and to install additional beam diagnostics both resulting in adding a lot of devices to the system. On the operating level main focus is in clarifying and unifying the complex handling and thus enabling a simpler and more reliable routine operation.

We would like to take the opportunity to thank all our colleagues who have contributed to the success of this control system.

X. REFERENCE

[1] V. Schaa, G. Fröhlich, M. Balloff, Jun-Ye Wang, "A New Generation of Operating Programs for the Accelerators", GSI Scientific Report 1987, p. 357

[2] M.C. Crowley-Milling and G. Shering, "The NODAL System for the SPS", CERN 78-07 (3 September 1978).

[3] R. Steiner, C. Riedel, W. Rösch, "Real Time Aspects of Puls to Puls Modulation", this conference.

VME Applications to the Daresbury SRS Control System

B.G.Martlew,M.McCarthy,W.R.Rawlinson
SERC Daresbury Laboratory,Warrington WA4 4AD UK

Abstract

The control system for the Daresbury SRS has recently been extended with a VME based alarm system which is operational. A further development is a steering system to provide servo control of the electron beam orbit position in the storage ring.

I. INTRODUCTION

The Daresbury SRS is a 2 Gev electron storage ring dedicated to providing synchrotron radiation to approximately 32 stations on 10 beamlines. It came into operation in mid 1981 and was upgraded with a high brightness lattice in 1987.

The control system for the SRS was designed and constructed in the period 1975-1980 and the original computers were upgraded in 1985.

The original control system provided an alarm condition monitoring system with a sampling resolution of 2 minutes. Recently, a dedicated VME system has been added which provides alarm monitoring and indication with sampling at the level of 5 seconds.

The high brightness lattice upgrade in 1987 reduced the source size by a factor of 10 and this led to difficulties with beam alignment and positional drift over the period of a stored beam. Work is under way to provide a VME based beam steering system to provide servo control of the electron beam position.

This paper will give a brief description of the SRS control system followed by a description of the new alarm system. A description of the beam steering system and its present status will be given.

II.THE SRS CONTROL SYSTEM

The SRS control system consists of a network of 4 Concurrent Computer Corporation(CCC) 3200 series computers as shown in Fig 1.

All computers in the network have a CAMAC system crate and communicate via CAMAC fast serial data links. CCC3230 is the operator interface computer providing service to three operator consoles in the main control room via a CAMAC parallel branch. Two of the operator consoles contain a colour display and keyboard (Tektronix 4207), a knob and tracker ball and a monochrome graphics display. The third console contains a Tektronix 4207 colour display and keyboard and serves as the personnel safety console.

CCC3205A and CCC3205B computers provide the interface to the plant via serial CAMAC highways. CCC3205A has two serial highways, one for the linear accelerator and Booster synchrotron and one for the Storage ring and Beamports. CCC3205B provides control for the beamlines with one serial CAMAC highway. Local control at the plant is implemented with Tandy 102 computers. The total parameter count presently stands at approximately 1800.
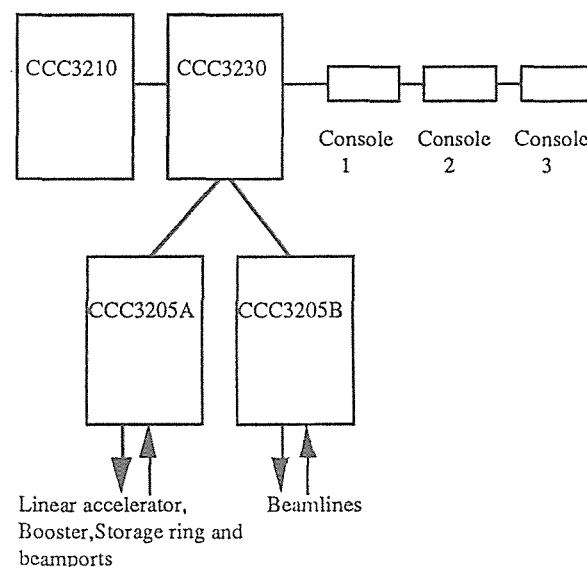
Fig 1. The SRS Control system

CCC3210 is offline to the control system and is used for programme development and testing.

High level programming is done in RTL/2 with more recent applications written in C.

III. THE ALARM SYSTEM

The original SRS alarm system was in operation for 9 years from the start up of the machine. Over this period, the system has been found to have several disadvantages:-

1. Noisy analogue input signals gave spurious alarm indication which led to alarms being ignored by operators.
2. Alarm conditions were applied to large numbers of parameters which led to 'swamping' of the alarm display and the operators being presented with more information than they could reasonably handle.
3. The large number of alarmed parameters led to difficulties in administration of the system.
4. The 2 minute sampling rate meant that the system was in fact no more than a fault indication system rather than a true alarm system.

5. The monochrome,text-only display did not provide an eye-catching indication of alarms.

Whilst it would have been possible to improve the existing system for points 2. and 3., the system could not have been enhanced to address points 1. and 4. without seriously effecting control system performance. It was therefore decided to produce a new alarm system with separate intelligence. The new system was to have the following features:-

1. The number of alarmed parameters would be kept to a relatively small number of important parameters.
2. Faster sampling of alarmed parameters was required.
3. Operator interaction with an alarm display was essential.
4. Audible alarms in the Control room were desirable.
5. Alarm history for at least the last 24 hours should be available.
6. A hard copy of alarms was required.
7. A clear,unambiguous display of alarm conditions was essential.

A VME based system was chosen both for its long term expandability and to allow us to gain experience with VME and the chosen operating system,OS/9.

The hardware chosen consisted of a Motorola 68020 processor running at 16MHz with floating point co-processor, 4 Mb Ram, 5 RS232 channels, interface for high resolution coloured graphics device, keyboard and mouse and 40 Mb hard disc and floppy disc drive. The system is shown in Fig 2.
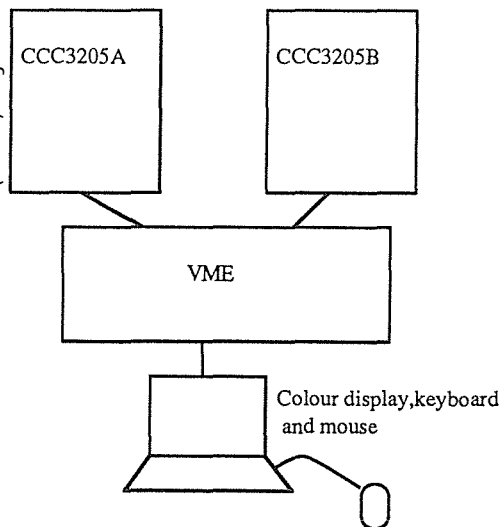
Fig 2. Alarm system

CCC3205A and B are connected to the VME crate with RS232 links operating at 9600 baud. A future improvement will be to remove the main control system CAMAC fast serial data links and replace with an Ethernet LAN to which the alarm system would be connected. Simple, low priority

processes in the 3205 computers monitor a table of parameters every 5 seconds and transmit raw analogue and status data over the RS232 links to the VME crate.

A matrix type display consisting of various zones corresponding to different machine areas, forms the alarm system indication. Each box on the matrix is normally depicted in black and white with a change to flashing red when an alarm for that zone is detected. By selecting the flashing zone with the mouse, the operator is presented with detailed information on the alarms in that zone and flashing ceases although the box remains red until the alarm is cleared.

The software is arranged as a ring of processes communicating via OS/9 signals as shown in Fig 3. Data from the 3205s is transmitted over the RS232 links to the input processes and piped to the alarm monitor process which performs alarm checking and passses information for display and logging to the display manager and then onto the disc logger for history storage and to the print logger for hard copy. All processes have access to the alarm system database. The alarm monitor process checks supplied analogue and status data against limits in the database to determine the presence of an alarm condition. Warning and danger levels may be specified for the alarm condition. The operator may request the system to ignore or notice individual alarmed parameters. Future extensions to the system include the implementation of rate of change alarm conditions and the provision of audible alarms.
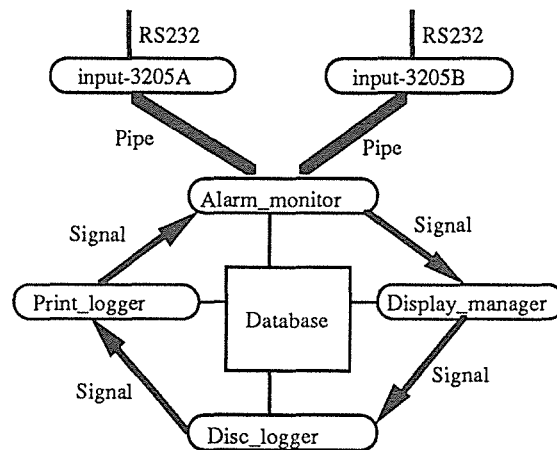
Fig 3. Alarm system software

IV. BEAM STEERING SYSTEM

Since the upgrade of the SRS to a high brightness configuration in 1987[1] there has been a growing requirement from the user community for improved beam position stability.

Beam stability measurements on the SRS and requirements for a new steering system have been described elsewhere[2]. Vertical beam position stability is most critical for users, with the largest movement being a peak displacement of $250\pm100\mu m$ in the first 3 hours of a beam fill followed by a roughly linear decay of approximately

20μm/hour for the remainder of the fill which typically lasts 24 hours. The existing electron beam monitors have an accuracy and repeatability of ±50μm and are capable of providing only gross orbit correction. A two wire photon beam monitor has been used at Daresbury to monitor photon beam position and has shown a useful range of 3mm with a resolution of 10μm at 21m from the source[2].

Steering elements on the SRS consist of 16 vertical steering magnets, 16 dipole trims produced by backleg windings on the main dipole magnets and 16 multipole magnets each made up of 12 individually powered windings which provide a combination of horizontal steering, vertical steering,octupole and quadrupole fields. In total there are 224 individual power supplies which are driven from 12 bit CAMAC DACs.

Planned improvements to the steering system are:-
- Upgrade of the steering magnet supply DACs from 12 to 16 bit resolution and accuracy.
- A VME based system to provide local intelligence and increased bandwidth for steering magnet servo control.
- Production of tungsten vane monitors for photon beam position monitoring.
- Provision of improved electron beam monitors having a resolution of 10μm[3].

The aim is to achieve a photon beam stability of 50μm at 20m from the source.

V. VME-BASED STEERING SYSTEM

The new steering system will involve disconnection of the steering magnet DACs and ADCs presently in CAMAC serial crates and connection to VME DACs and ADCs housed in VME plant interface crates. Fig 4. shows the arrangement of the new system.

The Steering Process system crate contains three processors, Gateway, Database server and Servo. The Gateway processor is the interface to the existing control system and contains code which makes it appear as another 3205 processor to the 3230. The 3230 and the Gateway processors are nodes on a dedicated Ethernet LAN. The Database server processor contains the steering system database and is connected to another Ethernet LAN which also has plant interface crates as nodes. The Servo processor contains global and local feedback algorithms. All three processors have access to the database which is held in shared memory. The plant interface crates contain a processor (PIP) and enough DACs and ADCs to service one quarter of the storage ring steering magnets. In addition, these crates will have ADC channels to read in signals from electron and photon beam monitors and various environmental parameters.

The Gateway processor has access to the hard and floppy discs while all other VME processors in the system are ROM based systems. The database will be continously updated by a process in the Database server communicating with the plant interface processors.

The processors are MVME147s with a Motorola 68030 cpu with a clock speed of 25MHz, 8 Mb RAM, 4 serial and 1 parallel ports and floating point processor. There is an 80Mb

Winchester disc and 1Mb floppy drive in the steering system process crate accessible to the Gateway processor.

pip: Plant interface processor   D/A: DACs and ADCs
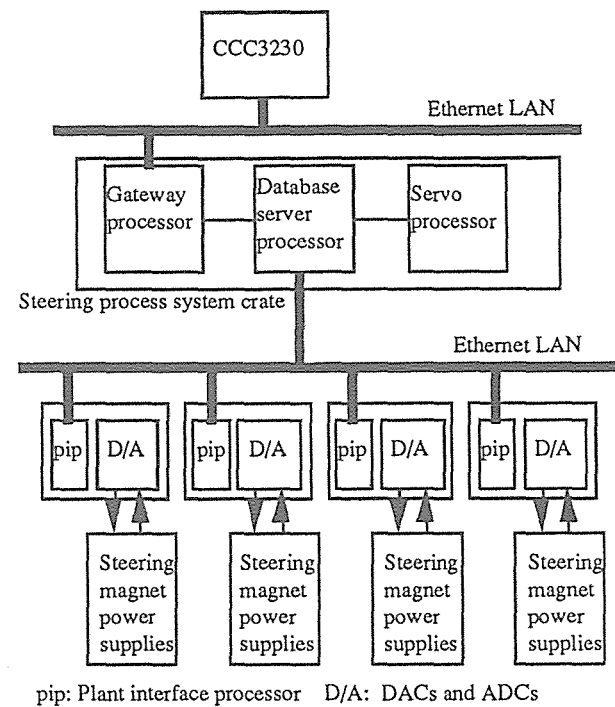
Fig 4. VME based steering system

The Gateway processor runs under Professional OS/9(V2.4) while all others run under Industrial(ROM based) OS/9. Application software is written in C. The communication protocol is TCP/IP.

VI. PRESENT STATUS AND CONCLUSIONS

The new steering system hardware has been purchased apart from the DACs and ADCs which are presently being evaluated. The interface software both in the 3230 and Gateway processors has been written and tested and the Database structure has been established. Work is progressing on the processes in the Database server and Plant interface processors for database updating. Further work is necessary to design and write code for the Plant interface processors. It is intended to install and test a minimal system without the new electron beam monitors in 1992.

The power and versatility of VME make it a strong contender to form a major part of a future accelerator control system at Daresbury.

VII. REFERENCES

[1] V.P.Suller et al., "SRS-2: Performance and Achievements" presented to the Particle Accelerator Conference, Chicago, March 20-23 1989.

[2] P.D.Quinn and T.Ring, "Developments in orbit control at the SRS at Daresbury", presented to ABI Conference, KEK, Tsukuba, Japan, April 22-24 1991.

[3] T.Ring and R.J.Smith, "Orbit measurement techniques at Daresbury", presented to ABI Conference, KEK, Tsukuba, Japan, April 22-24 1991.

Accelerator Control Systems in China

Yao Chih-Yuan
Hefei National Synchrotron Radiation Laboratory
U.S.T.C.

Abstract

Three accelerator facilities were built in the past few years, the 2.8 GeV electron positron collider BEPC, the heavy ion SSC cyclotron accelerator HIRFL and the 800 MeV synchrotron radiation storage ring HESYRL. Aimed at different research areas, they represent a new generation of accelerator in China.

This report describes the design philosophy, the structure, performance as well as future improvements of the control systems of the these facilities.

I. INTRODUCTION

The development and research of accelerators in China has made good progress in the past thirty years. Many low energy accelerators for research and application have been constructed, including high voltage type accelerator, cyclotron, linear accelerator, betatron etc. Their application covers a wide range: medical treatment, industrial irradiation, non-destructive inspection, isotope production and many other fields. The three newly completed high and medium energy accelerator facilities, Beijing electron positron collider(BEPC), Lanzhou heavy ion research facility(HIRFL) and Hefei synchrotron radiation source(HESYRL), aimed at different scientific research areas, represent a new generation of accelerator facilities.

Early accelerators are mostly controlled by panel meter and push button type manual control system. Application of microcomputers to accelerator control system started at the beginning of 80's when microcomputers were becoming popular in China.

This paper is not intended to be a general survey. Control systems of the three new accelerator facilities, which are relatively larger in scale and more complicate in structure, are described here with the emphasis on the system architecture.

II. BEPC CONTROL SYSTEM

BEPC is the first high energy accelerator built in China. The main facilities of BEPC are a 1.4 GeV electron linear accelerator injector, a 1.4 GeV beam transport line and a 2.8 GeV electron storage ring. The project was started in 1984 and completed in 1989.

Because of the strict time table for the

BEPC project, the leaders of the project decided early in 1985 that in order to reduce development work and shorten the construction period the new control system of SPEAR ring should be adopted as the base for BEPC.

BEPC control system is a typical centralized control system. A VAX11/750 serves as the central control computer. Serial CAMAC systems are the base for equipment interfacing.

1. System Configuration

Fig. 1 is a block diagram of BEPC control system. The system is functionally divided into three levels.
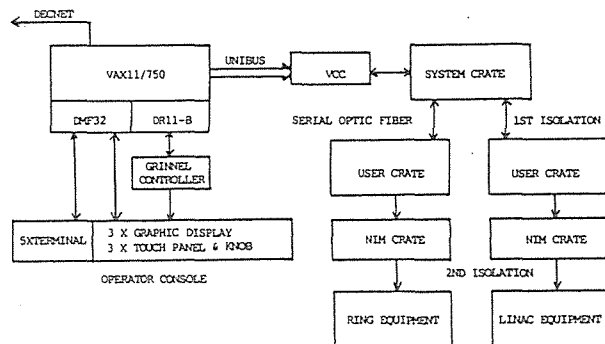
FIG.1 BLOCK DIAGRAM OF BEPC CONTROL SYSTEM

First, at the center of the system is the DEC VAX11/750 computer, which is equipped with a asynchronous serial interface board for connecting terminals and knobs, a DR11-B DMA interface for connecting color display monitors, and other standard peripherals such as hard disks, printers, tape drivers etc. A SLAC designed Vax CAMAC Channel(VCC) is the key element in data communication network. It is a DMA controller to interface .VAX11/750 UNIBUS with CAMAC system. Two CAMAC system crates are controlled by the VCC, one for the linear accelerator the other for the storage ring. One system crate houses several serial branch driver modules and each branch driver starts a fiber optic serial high way loop which connects up to 7 user CAMAC crates. The second level is the local control stations formed by user crates. I/O functions are performed by the local control stations. Each local control substation controls one

section of equipment, such as ring magnet system, vacuum system, transport line magnet system, RF system, linac magnet system, BPM system etc.

At the third level are NIM signal converter and isolator modules distributed at equipment site. The NIM modules must match BEPC hardware and are all developed by IHEP.

2. Software System

BEPC control software is a database driven system taking full advantage of multi-tasking ability of VAX/VMS operating system.

The database and its manager are the center of the system. All the machine hardware and program operation parameters are stored in the database which resides in a VAX global section shared by all software processes.

Software processes are executable programs under VMS operating system. They can be invoked by terminal command or by another process.

Two processes have special role in the system. A memory resident VCC I/O program CXCAMAC continuously performs I/O operation, sends out messages to CAMAC systems, collects status information from CAMAC systems and refreshes database records. The other special process is AVTX, which is a command receiver/interpreter and scheduler. It receives touchpanel and knob commands, decides which processes to call to perform the demanded operation, and activate appropriate processes.

Database generation and maintenance processes are also included in the system for ease of database initialization and modification.

Up to now, a total of 17 software processes are implemented for BEPC system, including ring magnet control and status display, ramping process control, BPM data taking and processing, closed orbit correction, ring lattice calculation, ring and transport line modeling etc.

3. Present Status

The system was completed in the middle of 1988, a few months ahead of first colliding beam experiment. After that it has been operated for beam colliding experiment and machine study for more than two years. System reliability is proved satisfactory. Some modifications has been made to the system to improve its performance. Dual speed Dual accuracy ramping function are developed for main magnets and correctors to increase the speed of ramping process. Analog control part of database is reorganized to enable more flexible and faster I/O operation. RF control function is added and a new BPM data acquisition program is developed to reduce BPM scan time and generate graphic display of beam position.

4. Upgrading Plan

Because of the centralized architecture, all control and monitoring operations must be initiated by VAX11/750. This takes substantial part of CPU time, especially during a ramping process. Many memory resident processes further increase the load of VAX11/750. The processing speed and response time of the entire system are not satisfactory. Further more, system reliability to some extent depends on the reliability of VAX 11/750, as it is the only controlling master. Another problem is that because of its limited processing power, VCC sometimes becomes the bottle-neck of message communication.

In order to overcome these problems, plan has been made to convert the centralized system to a distributed system. DECNET will be used to link the VAX11/750, A micro VAXII and a VAX Work-Station as three nodes. Fig.2 shows the upgraded system configuration.
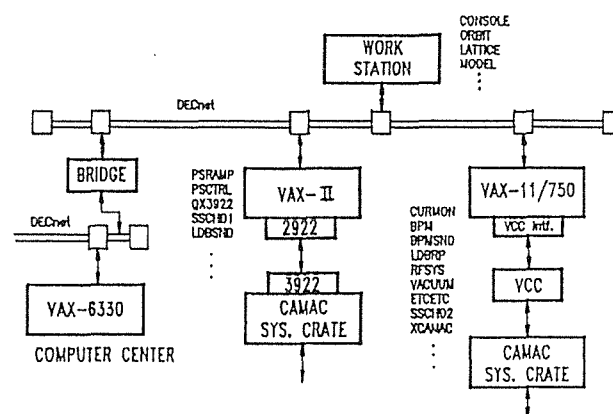
Figure 2   The upgrade of BEPC control system

The Micro VAXII will be dedicated to the task of magnet power supply control. The WS will be used to replace the present console terminals. The rich window software functions of WS will be fully utilized to update the man-machine interface. The WS will also share some of the calculation tasks, such as ring modeling, orbit calculation etc., with the VAX11/750.

A KSC 3922/2922 Q-BUS CAMAC adapter will be used to interface the Micro VAXII to the system crate for power supply control. The communication bottle-neck problem will be basically resolved, partly because the amount of messages through VCC are reduced substantially, partly because the QIO operation of 3922 is faster.

All hardware below the system crates will remain unchanged. This will ensure hardware compatibility.

The software compatibility among VAX family systems is an advantage for software upgrading. Except for the CAMAC I/O driver program, most of the software developed for VAX11/750 can be moved to Micro VAXII system and the WS with only minor modification.

In order to coordinate the processes on

different nodes, new communication programs will be developed to enable inter-node and inter-process message exchange.

III. HIRFL CONTROL SYSTEM

HIRFL is a variable energy heavy ion accelerator. It consists of a 1.7 meter radius sector focusing cyclotron modified from a former 1.5 meter cyclotron as injector and a new built separate sector cyclotron. The project was started in 1976, and completed in 1988.

Control system of HIRFL is a distributed control system formed by a Vax cluster as the central computer and several microcomputer controlled CAMAC control substations(CSS).

1. System Configuration

Fig. 3 is a block diagram of the HIRFL control system. Two VAX8350 computers are linked together through the CI bus and DECNET to form a cluster. One of them, which serves as the central computer of SSC control system, interfaces to serial CAMAC system through a UNIBUS adapter and a serial branch driver. The other serves as a data processor, software development system and a backup system. The cluster is equipped with 2X12MB of memory, 4X520MB hard disks and other standard peripherals.

A DZ11 asynchronous interface, a DR-11 parallel interface and a parallel CAMAC branch driver are also installed on the UNIBUS for connecting console devices, including 6 touch panel monitors and 6 color monitors.

Serial CAMAC high way communication link is used for the whole control system.

A control substation is an intelligent CAMAC crate containing a serial crate controller which is the main controller of the crate an auxiliary controller for substation computer and other CAMAC modules. Control command may either be issued by the central computer or by the substation microcomputers. Communication between the central computer and substation computers is also performed via the serial high way. A substation can be further expanded through a secondary serial CAMAC high way loop if more than one crates are required. Several types of control substations have been constructed during system development period. In order to reduce the number of different subsystems, only two of them, the PC based and LSI-11 based CAMAC control substations are adopted for present system.

Accelerator equipments are divided into several subsystems. They are controlled either by normal serial CAMAC crates or by control substations.

The injection and extraction transport line and SSC magnet power supplies are all controlled by serial CAMAC crates which are directly connected to the main high way loop and controlled by the central computer.

RF system is controlled by a PC based CAMAC control substation composed of 8 CAMAC crates.

Vacuum system is monitored by another PC based CAMAC substation.

Beam diagnose system is controlled by a LSI-11 based CAMAC control substation composed of 4 CAMAC crates.

All equipments are connected to CAMAC modules via signal conditioner electronics which was developed by HIRFL.

2. Software system

The software of HIRFL control system has two levels. The main control programs for the central computer are all VMS tasks and
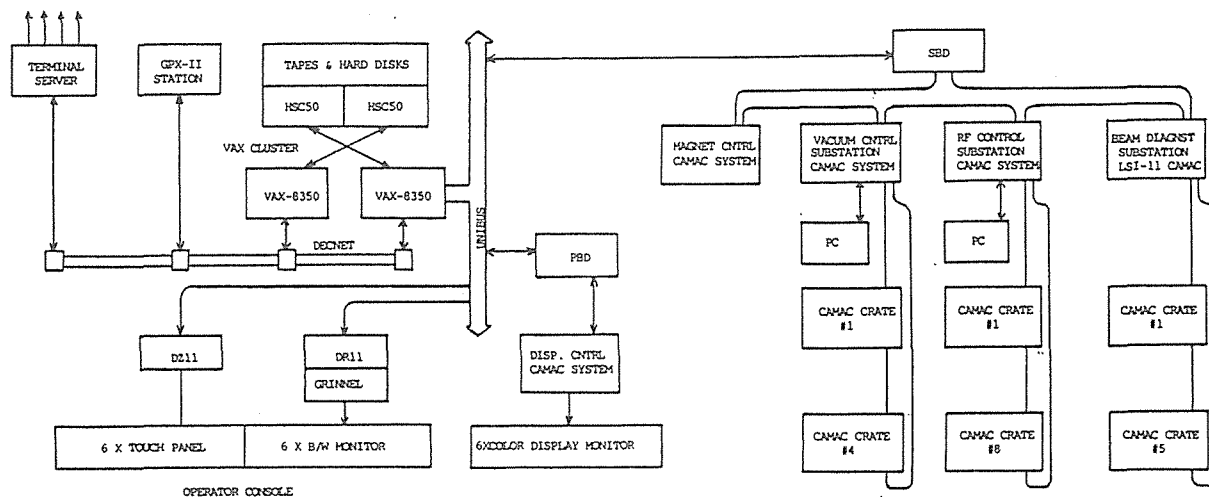
FIG.3 BLOCK DIAGRAM OF HIRFL CONTROL SYSTEM

written in FORTRAN. They accept main console command, generate various displays and send control command and data to CAMAC crates or to the control substations.

Software for the substations were written with FORTRAN and BASIC languages. Menu type man-machine interfaces are implemented for convenience of operation. The substation control programs perform all the basic control functions: equipment parameter adjustment, on/off control, status monitoring and generation of local status displays.

3. Present Status

All the control substations were completed in 1988 and was in operation before the beginning of SSC commissioning. They have satisfied the requirement of HIRFL commissioning and operation.

A system composed of two 386 PCs and CAMAC systems were built for the SFC control recently.

The VAX cluster was installed in Jan. 1988. A VAX control program for SSC magnet power supply system is completed and in operation. Development of VAX central control programs for other subsystem is under way.

The next goal of HIRFL control group is to realize central control of all SSC and the SFC equipments.

IV. HESYRL CONTROL SYSTEM

HESYRL is a dedicated synchrotron radiation source designed for UV and X-ray experiments. The project started in 1984 and completed in 1989.

The main facilities of HESYRL are a 200 MeV electron linear accelerator, a beam transport line, and an 800 MeV electron storage ring. The control system are mainly divided into three relatively independent parts: a linac control system, a ring control system and a timing system.

1. System Configuration

The HESYRL ring control system is a distributed computer control system composed of a PDP11/45 computer, two 286 PCs, a communication microcomputer and up to 40 local control microcomputers(LCM). Fig. 4 is a block diagram of the ring control system.

Originally we plan to use a VAX11/780 as the main control computer. Both budget and delivery time problem led us to the choice of a PDP11/45 computer as a temporary alternative. Now because of the small memory size and poor display ability, its function is reduced to part of communication. Main control functions and console operation are all performed by the two PCs which are connected to serial ports of the PDP.

Ring equipments are controlled by local control microcomputers. Most of them are MULTIBUS system composed of a crate, a CPU board, a serial communication/memory

expansion board and a several other interface boards. Because PCs are more convenient in programming and display we have also used some PCs as LCMs. The LCMs are located near the equipments. One LCM controls only one or one type of equipments.
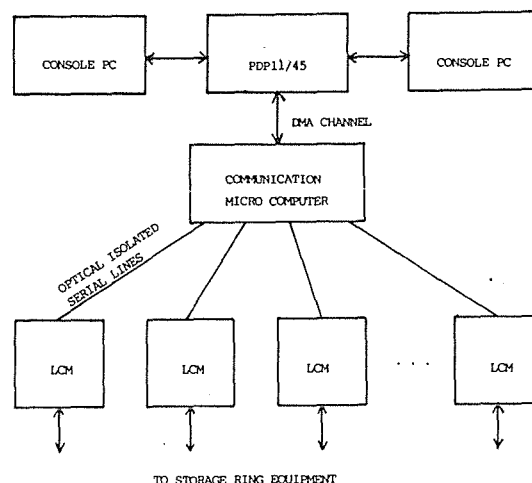
FIG.4   BLOCK DIAGRAM OF HESYRL RING CONTROL SYSTEM

Because of the local intelligence of the LCMs, direct control and monitoring of the hardware equipments from the main computer are unnecessary. Most of operations are performed by the LCMs, system reliability and speed is improved. This is more obvious for ramping operation. With preloaded ramping table and hardware synchronization, ramping time can be less than 1 minute.

The CMM is a microcomputer specially designed for communication. It is a MULTIBUS system consists of a master CPU board, a DMA communication board and up to 10 intelligent serial communication board. CMM exchanges data with PDP through DMA channel and communicates with LCMs through serial links. Maximum number of channels is 40.

Low cost and convenient optical isolated asynchronous serial lines are used for data communication between the CMM and LCMs.

2. Software

The software for the control system consists of three levels: ROM monitor for the LCMs, communication program and ring control program.

ROM monitor of the LCM micros manages the LCM resources, handles communication and performs various control and monitoring functions.

The communication program includes ROM control monitor for the CMM, DR11-B driver and data buffer manager for the PDP.

The main function of CMM monitor is to handle the data exchange between the LCMs and the PDP. All data are exchanged in records. Error detection and retransmission are

implemented.

The PDP DR11-B driver is installed as a device driver under the RSX11-M operating system and can be invoked by QIO. LCM status informations are stored in a shared data region on the PDP which is accessible by the console PC programs.

Ring control program includes transport line control, ring device status display, ring lattice setting and changing, beam energy ramping process control, close orbit correction and equipment testing programs.

Presently only transport line control, lattice setting, ramping control program and orbit correction programs are executed on the console PCs. Ring modeling, ramping table generation and other calculations are performed off-line.

3. Present status

The system was completed in March 1989 and has been operated since then. During the commissioning period the control system has performed satisfactorily.

Ring magnet control assured stable ring operation.

Ring control programs serve well for injection and ramping process control. But off-line calculation of physics parameters is inconvenient for machine study.

Beam energy ramping control is very successful. With the table ramping technique energy ramping or transfer to different lattice is very easy and smooth. Beam loss rate from 200 MeV to 800 MeV with 200 mA current is less than 7%. This is a good indication of ramping accuracy.

System communication has been reliable. But the response is slow. This is due to the speed of serial lines between console PCs and the PDP. LCMs are reliable and easy to maintain.

4. Upgrading plan

Several problems exist in the present HESYRL control system: 1) Transport line control and vacuum monitoring are now controlled by two standalone PC control system. They are not linked to the ring control system. This is not convenient for system management and data logging. 2) PDP seems unnecessary in the communication system, also it becomes a weak point. 3) Lack of on-line calculation ability is not desirable.

In order to solve the above problems, system improvements are planned.

DECNET will be implemented to link a Micro VAXII, the console PCs, transport line control PC, vacuum monitoring PC and a recently installed VAX6310 system together. This will give us a fully linked system with much expanded computing power.

The PDP computer will be replaced by two BIT3 bus connector cards which connect PC bus to MULTIBUS of the communication microcomputer. This will eliminate the

bottle-neck and increase data speed of the whole system.
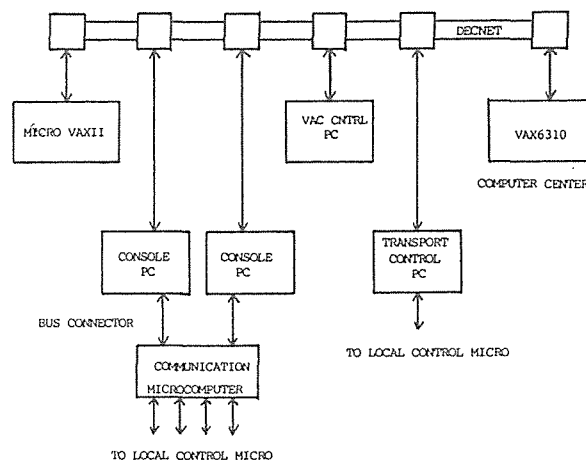
FIG.5 UPGRADE OF HESYRL CONTROL SYSTEM

Console display and computation power will be upgraded by replacing the console PCs with more powerful 386 or 486 stations.

Control programs and system communication programs will be rewritten to include on-line calculation function and node to node communication.

V. CONCLUSION

Started almost in same period, the control systems of the three accelerators are now all in operation. In parallel with the development of the systems, our chinese colleagues have gained experiences and established their qualified cooperative teams. This is a promising start.

Compared with world advanced laboratories our systems are not advanced no matter in architecture, hardware or software. Some system upgrading and optimizing work remains to be carried out.

VI. ACKNOWLEDGMENT

During design and construction periods we have received great help from SLAC, NSLS, KEK, CERN and many other laboratories. we would like to take this opportunity to express our thanks.

I would like to thank Prof.S.Y. Liu, D.K. Liu of IHEP and Prof. T.S. Jiao of IMP for helpful discussions and letter communications in preparing this report.

VII. REFERENCES

1. BEPC control group, The Status and Upgrade of BEPC Control System, BEPC workshop, June 1991, Beijing
2. T.S.Jiao, HIRFL Control System, Proc. of The Fourth Symposium of PASC

S01SRA09

HESYRL Control System Status

Yao Chih-Yuan
Hefei National Synchrotron Radiation Laboratory
U.S.T.C.

Abstract

HESYRL synchrotron radiation storage ring was completed in 1989 and has been in commissioning since then. Now it has met its design specification and is ready for synchrotron light experiments. Control system of the project was completed in 1989 and some modifications were made during commissioning. This paper describes its present configuration, status and upgrading plan.

I. INTRODUCTION

Hefei National Synchrotron Laboratory (HESYRL) is a dedicated synchrotron light source. It's main facilities are a 200 MeV electron linear accelerator, a beam transport line, an 800 MeV electron storage ring and the experiment stations.

Design and construction of the ring and linac control system started in Oct. 1984. The linac control system was completed in 1987 and the ring control system was completed in 1989. Modifications have been made to the system during two years of machine commissioning, including RF control, vacuum monitoring and new control programs.

The ring control system is a distributed computer control system consists of a PDP11/45 computer, two PCs, a communication microcomputer system(CMM) and up to 40 local control microcomputer systems(LCM).

The linac control is essentially a manual control system.

A timing system consisting of two microcomputers provides all necessary triggering signals for the linac and the injection system.

II. RING CONTROL SYSTEM

A. System Configuration

Fig.1 is a block diagram of the ring control system.

Two PCs perform the main control function. Console display and command input are also performed on the PCs. The PDP11/45 minicomputer system, originally planned to be employed as the main control computer,is now only a part of communication system because of its limited memory and poor display ability. The two PCs are connected to the DZ11 ports of the PDP. Ring control programs

are executed on the PCs.

Storage ring and beam transport line equipments are controlled and monitored by local control micros. The LCMs are located near the equipments. Each LCM controls only one or one type of equipments.
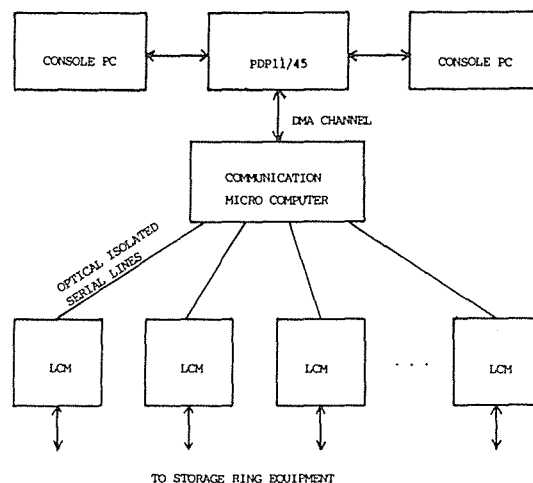
FIG. 1 BLOCK DIAGRAM OF HESYRL RING CONTROL SYSTEM

A LCM consists of a MULTIBUS crate, a SBC80/24 or SBC80/20-4 CPU board, a home designed serial interface and memory expansion board(HCOM) for communication and a few interface boards for equipment interfacing.

System communication is performed at two levels. The console PCs exchange data and commands with PDP through its serial lines. The PDP communicates with LCMs via a dedicated communication microcomputer system, which is a MULTIBUS system composed of a master SBC 80/24 CPU board, a DMA communication board and between 1 to 10 intelligent communication boards(COMM).

The COMM board is similar to Intel SBC544 intelligent asynchronous communication board. It has a Z80 CPU, 4 serial ports,on board RAM/ROM and dual port RAM memory which can be accessed by both a MULTIBUS master and the on board Z80. One COMM board can handle communication with 4 LCMs.

The DMA board controls DMA data transfer between the dual port memory and a DR11-B DMA interface of the PDP.

The main control console is made up of 9 modified industrial cabinets which are equipped with terminals, video monitors and various control panels. It is functionally partitioned into several subpanels: transport line magnet control, injection system control, timing control, ring control, beam diagnosis, interlock and personal safety etc. Most of the operations are performed at the console PCs.

B. System Software

There are three levels of control programs.

ROM monitors of the LCMs are at the lowest level. The basic design idea of LCM monitor is brought from NSLS. It consists of two parts: a basic control monitor and an application part. The former initializes the micro, handles communication messages and schedules operation of different control processes. The latter is essentially a collection of interfacing routines designed for the particular hardware configuration. The Basic control monitor is identical for all the LCMs.

At the second level are communication programs including ROM control monitors for the communication microcomputer, and a DR11-B I/O driver for the PDP.

The main function of SBC80/24 monitor is to handle the data exchange between the dual port memory and the PDP. All data are transferred in DMA mode.

There is a ROM monitor program on each COMM board of the CMM, which manages communication with LCMs.

The I/O program for DR11-B interface is installed as a RSX11-M device driver which can be called by QIO.

Ring and transportline control programs are at the third level. Three programs are used for normal operation. A program called RINGTEST performs ring device status display, ring magnet current and RF voltage setting, console command interpretation, ring lattice adjustment, file saving and restoration, beam energy ramping process control etc. It mainly deals with hardware signals. The second is BLOS, which performs transport line and linac magnet setting and adjustment. The third program, CORRECT, does closed orbit correction.

Ring lattice modeling, ramping table generation, closed orbit analysis and other calculation are performed by off-line programs.

C. Equipment Interfacing

The LCMs are mainly partitioned into the following seven subsystems:

HESYRL ring has 1 dipole group, 8 quadrupole groups and two sextupole groups. The main magnet power supply control system consists of 12 LCMs, 11 of them are ramping magnet LCMs each controls one magnet power supply. A LSI-11 controlled CAMAC system and a DVM is used to read back DCCT values of all the magnet currents. The dipole LCM also contains a ramping timer board which is a synchronizer for all other LCMs. Fig. 2 shows the configuration of the main magnet control system.
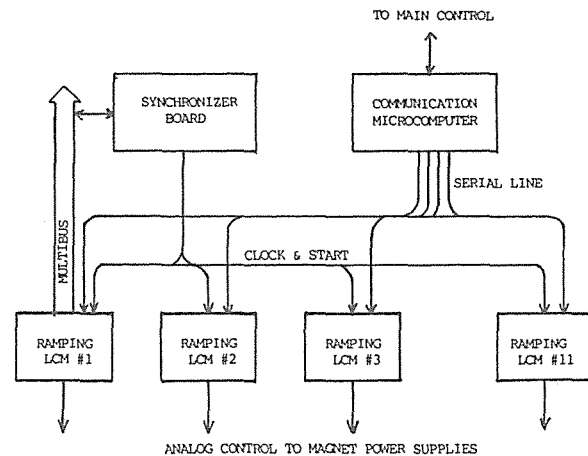
FIG.2 BLOCK DIAGRAM OF RAMPING CONTROL SYSTEM

The injection energy of HESYRL ring is 200 MeV, which is only 1/4 of its operation energy of 800 MeV. Ramping process is critical for high beam current.

During an energy ramping or ring configuration change process the currents of the magnet power supplies must follow certain calculated curve synchronously in order to keep the beam stable. In a centralized system the central computer has to modify magnet current one by one in many small steps. It will takes a lot of CPU time. The local intelligence of LCM allow us to take different approach. It is called table ramping. Ramping data are pre-stored to the LCMs. Synchronization of different LCMs is kept by hardware signals.

A ramping magnet LCM is composed of a CPU, a serial line interface, a ramping controller board and a 16 bit D/A converter board.

A ramping curve is divided into many small straight segments. Each segment is defined by its end current and slope values. As many as 1000 segments can be defined for one curve. A ramping segment is generated by counting up or down of a 16 bit counter on the ramping controller. The clock input to

S01SRA10

the counter is divided from a common clock output signal of the synchronizer. The ramping LCMs are programmed such that when a segment end is reached the new segment and slope values are loaded. The synchronizer also provides a start signal for all the ramping LCMs. Tracking of different magnets is automatically assured.

In order to start a ramping process, the main computer just needs to load the segment table, send necessary command to the LCMs and then issue a start command to the dipole LCM to start the synchronizer.

A fitting program is developed to calculate optimized ramping table for any required ramping process. The theoretical fitting error is 1 bit.

12 auxiliary windings of dipole magnets and 64 auxiliary windings of quadrupoles are configured as 12 horizontal and 32 vertical closed orbit correctors. Two LCMs are used to control all these corrector power supplies.

Control of transportline focusing and correction magnets are performed by system composed of a SBC80/24 LCM in the power supply room and a PC on the operator console. The PC acts as the control master and the LCM accepts PC command and performs the actual control.

The beam position monitor signals are needed on line for closed orbit correction. A PC data acquisition system is used to read and display the BPM data.

Ion pump currents and vacuum gauge pressure readings are monitored by a PC data acquisition system.

Injection system includes a pulsed septum magnet and three fast kickers. The triggering signals are provided by timing system. Pulse amplitude control are performed by a SBC80/20-4 LCM with a terminal on the operator console for command input.

HESYRL ring has only one RF cavity, so phase control is not necessary. A tuning loop is built for cavity tuning control. Detuning angle is controlled by a manual adjusted phase shifter. Cavity voltage is regulated by a feedback loop. A ramping LCM is used to control the cavity voltage. During ramping process the cavity voltage is controlled in the same way as the ring magnets and therefore can follow any desired curve.

III. LINAC CONTROL SYSTEM

The linac control system is divided into 8 separate subsystems each has its own control panels and controls one part of the linac equipment. The modulator control panels, klystron focusing control panels, electron gun control panels, microwave control panels, vacuum control panel are all manual control panels.

The 200 MeV linac has an extension section for future beam energy expansion.

Linac focusing and correction magnets fall into two groups: 30 focusing and steering magnets for linac proper and 32 magnets for the extension section including a beam energy analyzer. They are controlled by two SBC80/20-4 LCMs and a PC. The PC serves as the master and operator console. The LCMs perform the local control and monitoring functions. The links between the PC and the two LCMs are fiber optical serial lines.

A SIMENS S5-101 programmable controller is used for linac interlock and protection.

IV. TIMING SYSTEM

The timing system provides synchronization triggering signals for the linac, injection system and beam diagnose system. Since timing is only adjusted during injection and linac set up time and most of its parameters are fixed, It is built as an independent system. Two LCMs are implemented: a main timing micro(MTM) installed in the main control room and a linac timing micro(LTM) in the linac control room.

The MTM generates master start and clock signals which are the reference of the whole system. The master signals are transmitted to the linac timing micro. MTM also provides trigger signals for the septum, kickers and for beam diagnose system. The LTM generates trigger signals for the electron gun, the microwave source, the klystron modulators and the switching magnet pulse generator.

Structurally the two micros are almost identical. They consists of a CPU board and a timing generation board(TIG).

Present timing system can only be operated in full bunch injection mode.

V. PRESENT STATUS

During the commissioning period the control system has performed satisfactorily.

Timing system, linac and transport line magnet control subsystems greatly eased linac and transportline tuning. Resolution and delay adjust range of timing system are suitable for optimization of linac operation.

Injection control and ring timing systems assured conveniently setting of correct injection condition. The ring control program has a linear approximation trimming function to adjust the ring energy slightly up or down. This is very useful for matching ring and injection beam energy to improve injection rate.

The measured accuracy of ring magnet current setting is better than 7×10^{-5}. This assured stable and repeatable ring lattice setting. The measured tune variation is less than .001.

The simple practical approach of RINGTEST control program served well in

machine commissioning and normal operation.

Beam energy ramping control is very successful. With the table ramping technique energy ramping or transfer to different lattice is very easy and smooth. Current tracking was measured better than 10^{-4}. Beam loss rate from 200 MeV to 800 MeV with 200 mA current is less than 7%, which is of same order as normal lifetime loss. This is also a good indication of ramping accuracy.

System communication has been reliable, but the system response is slow. This is due to the speed of serial lines between console PCs and the PDP.

VI. PLANNED SYSTEM UPGRADE

The operation experience showed us that there exist several problems in the present HESYRL control system: 1) It is not a fully linked system. A few standalone subsystems are not connected. This is not convenient for system management and systematic data logging. 2)PDP is an old system and difficult to maintain. It becomes a weak point. 3) Lack of on-line calculation ability is not desirable.

Having analyzed these problems, and considering our available resources, we plan to make the following improvements:

1) Introducing DECNET to link the MicroVAXII, console PCs, transportline PC, vacuum monitoring PC and a recently installed VAX6310 system together. This will give us a fully linked system and much expanded computing power.

2) Replacing the PDP by two BIT3 bus connector cards which connect PC bus to MULTIBUS of communication system and directly map the dual port memory to PC memory. This will eliminate the bottleneck and increase data speed of the whole system

3) Improving console display and computation ability by replacing the console PC with more powerful 386 or 486 stations.

4) CATV is very convenient to display machine status information to the users and other laboratory staffs. Installation of a 10 channel CATV system is under way.

5) Control programs will be rewritten to include on-line calculation function and node to node communication.

VII. ACKNOWLEDGMENTS

J.Yang of SSRL, J. Sheehan and J. Smith of NSLS contributed their suggestions during their visit to our laboratory.

The control group have been working cooperatively over the years to build and maintain the system.

VIII REFERENCES

1. S. Magyary et al, Operating Experience with a New Accelerator Control system Based upon Microprocessors, Proc. of 1981 Particle Accelerator Conference, Washington D.C.

2. K. Batchlor et al, Distributed Control System for the National Synchrotron Light Source, Proc. of 1979 Particle Accelerator Conference, San Francisco

3. Yao Chihyuan, Beam Control and Diagnostics of HESYRL Ring, Proc. of the 4th Japan-China Symposium on accelerator and its application, 1990, Beijing

FIG.3 UPGRADE OF HESYRL CONTROL SYSTEM

The Control System of HIRFL

Jiao Tianshu, Li Tianyou,Ma Siwen,
Chu Zhensheng, Huang Tuanhua, Zhou Xun,
Wang Zhen,Shen Zhiqing
Institute of Modern Physics, Accadmia, Sinica
No.57, Nanchang Road, Lanzhou

1. Introduction

The Heavy Ion Research Facility in Lanzhou (HIRFL) is a multi–purpose and variable energy machine designed to accelerate wide range of ions. [1] In order to obtain a designed beam (particale and energy) and to transport it to a proper experimental areas in a short time, it re- quires to modify a great number of parameters, this cannot be easily achieved without the help of a computer.

The control system design and construction was started in 1983. First of all, some local control station of accelerator subsystems were fin- ished in 1988 and satisfied the needs of operating and commissioning at the elementary level. Control- ling the HIRFL process is implementing at a high level.

2. The brief description of control system

Fig.1 shows the general layout of the control system for HIRFL.[2] It is based on CAMAC dis- tributed process configuration.[3]

(1) The local computer control stations are de- signed according to the accerlerator subsystems, such as Magnet, R.F, Vacuum, Injection and Ex- traction, Beam line etc. and were finished in 1988. They can meet the case of beam tuning and accel- erator operating at elementary level.
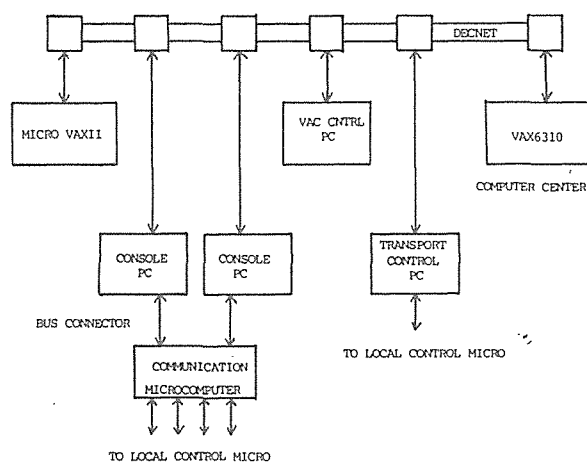

Fig.1   The block diagram of HIRFL control system

(2) All local stations microcomputer links to Host computer by CAMAC serial loop line. Two communication machanism are available:

(1) A memory "mailbox" CAMAC module accessible from both host and remote processors. (2) A CAMAC Dataway Communications module or "DMA port" between the remote processor memory and the Dataway.

2.1 Computer

HIRFL is controled by means of 2 kinds of processors: mini-computers as host computer and microcomputer as local stations. They are linked with CAMAC serial loop line.

The host computer is two VAX-8350 that are connected by cluster configuration. They share 4x520MB disks, 2x300MB movable disks and tapes. Each computer equipped with 12MB memory. One of them is used as a host computer for HIRFL control system. The other is used as a reserved computer when the former one is in fault. In addition, it is also used calculation and data processing for the experiments carried out in the experimental areas.

Microcomputers for the local stations are LSI-11/23 and IBM-PC. They are used to control subsystem for Vacuum, R.F, Beam diagnosis system and so on. These microcomputers are a completed system with memory, disk, terminal.

VAX-8350 control the SSC Cyclotron through the CAMAC bit serial loop. This loop was drived by serial highway driver. Serial rate is 2.5MB. There are about 20 CAMAC crates, corresponding CAMAC module are used for the interface of devices.

2.2 Device control

In order to bring into being link and matching between CAMAC modules and devices, 5V digitalized I/O-signals, 24V switch signals, and 5V analog voltage signals was planned. A variety of condition circuits, such as switich board, status board, DAC adjusting board with accuracy of 16bit-18bit and ADC data acquisition board with accuracy of 12-16bit, had been designed and made. They are used for on/off power supply, status monitor, current adjusting and data acquisition.

A step motor control is used for units requiring accurate position setting, such as electrostatic deflectors, magnetic channels. The controller of step motor is designed to provide with local control function. They are slow acceleration when movable units is started and slow deceleration when movable units is derived to the front of end. So movable units not only can keep running smoothly, but also can obtain higher operating speed.

In order to save funds, a number of digital and analogue multiplexer is used for devices of some slower action. Otherwise, the pressure operated control is mainly used for the units of two movable position such as Faraday cup, vaccum value, secondary emisson multi-wire profile monitor etc.

3. Console

Our console is designed after a careful analysis of which have been done in GANAL[4] and RIKEN[5].

The console, mechanically built with 10 benches is devided in 3 operating console units L, C, R. The central console unit C is devoted to equipments not linked to the computer, such as worksite monitor, RF waveform and beam signal obeservation. The console units L and R are identical.

Controlling the HIRFL process with the console system can be exercised at the elementary level or a higher level. At the elementary level the control system behaves as a large multiplexer. Equipments designed by their device name are handled one by one.

The operator uses the touch panel and the turn pages of the device name to choose one corresponding equipment he wants to control. Then, operator moves a cursor to a selected device name

on TV screen with a defined touch key and reads the various informations such as the controlled value the actual value of the parameter and a status word on the TV screen area designed.

Among the numerous signals collected along the accelerator, most of them concerning parameters values or status are digitalized and enter the data flow transmitted by the CAMAC system for processing. A little analog form is remain which are to be used in their anolog form. This is the case of some beam diagnosis signals and R.F signals with time. These waveform siganl observation is necessary.

These signals can be displayed on autoranging picoammeters or on oscilloscopes.

4. The present situation

The HIRFL operating software called HIRFLCSF which is written in Fortran 77. Presently, center console programs are developping for the man-machine interface with Touch Panel.

With time goes on, HIRFL operating level will improve.

Refernce

[1]. Zhang Enhou, Wei Baowen etc., Survey of Heavey Ion Research Facility in Lanzhou (HIRFL), proceedings of Heavy Ion Research Facility in Lanzhou, Vol. 8, 1 (1989)

[2]. Jiao Tianshu, The Control and Beam Diagnostic System, proceedings of Heavy Ion Research Facility in Lanzhou, Vol 8, 87 (1989)

[3]. J. W. Tippie, etc., Discussion of Tools and Techniques for Distributed Processor Based Control Systems Using CAMAC, IEEE Transactions for 1985

[4]. I. M. Prome, "The GANIL Control System", IEEE Transactions on Nuclear Science, Vol. NS-28, No.3, June 1981

[5]. T. Wada, H. Tarebe etc., Control system of RIKEN Ring Cyclotron, RIKEN Accelerator Progress Report, Vol.201, 170(1986)

Control System for a Heavy-Ion Accelerator Complex K4 – K10

V. M. Kotov and R. Pose
Joint Institute for Nuclear Research
Laboratory of Computing Techniques and Automation
Dubna, USSR

Abstract

Control systems for newly created accelerators, perhaps for the first time, may be designed almost only around international standards for communication and control techniques. This is also true for the project of a control system for the accelerator complex K4-K10 at the Joint Institute for Nuclear Research Dubna. Nevertheless, open systems architecture with construction principles being essential for modern systems of such big devices as particle accelerators leaves designers enough possibilities for solving even very sophisticated problems.

I. INTRODUCTION

The control system of the heavy-ion accelerator complex K4-K10 is similar to the systems developed for controlling the accelerators of other physical laboratories the experience of which was used in preparing the given project [1,2,3]. Nevertheless irrespective of the similarity of accelerators and control problems there are no equal control systems. This depends not only on the differences of accelerators as such but first of all on the time the system was designed and on the present state of computing and communication technique and on the special features of the system architecture, which allow new technical acquisitions during the realisation.

II. THE ARCHITECTURE OF THE K4-K10 CONTROL SYSTEM

The architecture of the K4-K10 control system is based on a two level distributed computing system (Fig. 1). The upper level uses a modular system IEEE 1296 (Multibus) from the Siemens AG [4] (SIMICRO) as well as workstations with UNIX and X Window in the Central Control Room, and is integrated into a system via a Local Area Network (IEEE 802.3).

The standard IEEE 1296 and its realisation in the SIMICRO products allows a high computing power inside every crate (the CPU on-board in the SC^{TM}-systems based on RISC processors providing 5 MIPS and more) as well as the use of CPU with embedded PC/AT 386 in the OSM^{TM} and AMS^{TM} systems[1]. These SIMICRO systems are equipped with a complete set of communication modules for LAN in the IEEE 802.XX standards.

The functions of the system at this level are supported by the operating system SORIX^{TM} - a UNIX^{TM} Real Time version (reaction time for interrupt signals $\leq$ 100 µsec.) and

---

[1]TM   SIMICRO, SX, SORIX, OSM, ASM - Trade Marks of SIEMENS AG

[2]TM   UNIX - Trade Mark of AT & T

permit an exit to LAN via TCP/IP protocols[2].

Therefore, using the possibilities of the 7 level scheme for the OSI model of open systems, the architecture of the upper level of the control system provides an output to the equipment and to the real time programs as well as a complete support for the TCP/IP protocols for an output to a LAN IEEE 802.XX. According to the loading of the network at this level by means of bridges one may distinguish two LAN segments: one for the workstations WS in the Central Control Room running under UNIX and the other for the Front End Computing. The latter are mainly crates in the IEEE 1296 standard with Real Time UNIX, joining the node computers for data acquisition and handling in the control mode as well as that for graphics and monitoring. They are equipped with terminals, Touch Screens and other man-machine communication devices.

The lower level of the distributed computing system directly corresponding to the equipment and executing devices and based on standards for industrial systems (for example in the VEPP4 [5] CAMAC instrumentation is used at this level) is connected to the upper level by means of the communication environment FIELD BUS (MIL-1553-B). This standard mostly agrees with the demands for a multidrop bus and in 1983 was proposed as a standard protocol for the field bus in accelerator control systems [6]. Together with well developed hand shaking features for the message transfer between bus controllers and remote terminals (RT) this standard allows simple and cheap data transfer to single serving devices, fulfilling simplest functions. For connecting digital measuring devices to the upper level the standard IEEE 488.X is used.

For the most part all the bus controllers (BC) MIL-1553-B are placed on the upper level of the control system. They provide the interfacing of the node computing system to the lower level equipment. The BC for the K4-K10 control system is developed on the basis of a processor module in the Multibus II standard, designed in the Laboratory of Computing Technique and Automation of the JINR using the VLSI of a 32-bit microprocessor set K1839 software-compatible with a micro VAX [7]. The interface controller for the MIL-1553 bus is a piggy-back module for the CPU board. One such BC may control 30 standard interfaces (Remote Terminals) on each bus.

The standard interface of the MIL-1553 bus has to provide a prompt/reply regime. In this case a microprocessor is needed. In the case of simple messages of the type "adjust and/or read" it has to handle the transfer directly.

The use of the MIL-1553 for the communication with the equipment together with the time synchronization channel allows one to put the real time mode on the lowest control level. Such a solution proposed and realized in the GSI [1] relieves the upper control level of the real time business and allows on this level the use of the LAN 802.3 alone without a TOKEN RING (IEEE802.5) as it is done for the SPS/LEP in CERN [2].

S01SRA12
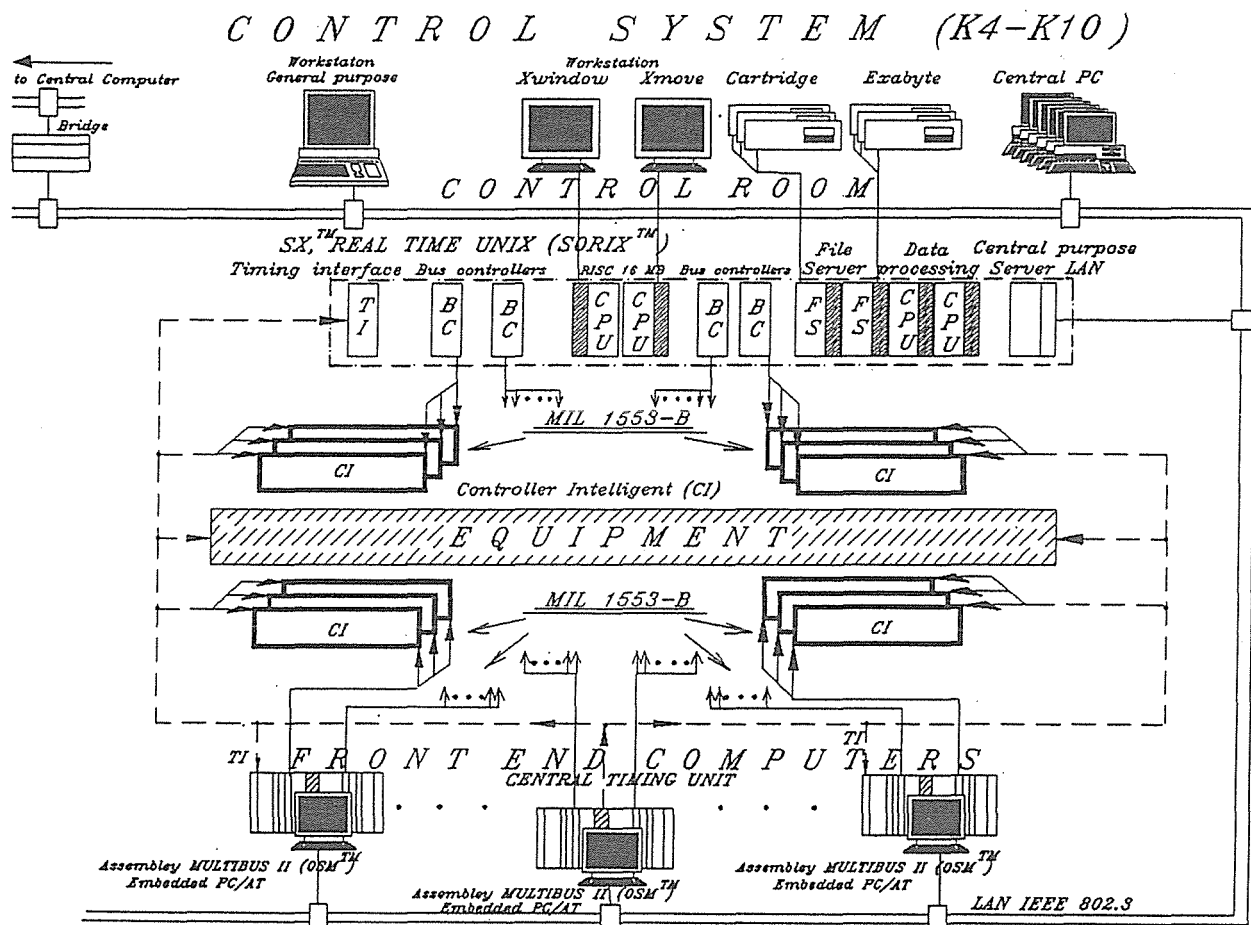
FIG.1

In the project of the control system for the K4-K10 it is also planned to use such a kind of real time system for the lower control level. But taking into account relatively small dimensions of the accelerator rings of the K4-K10 (the perimeter of the K10 is about 200 m) and the transfer rate of 1 MByte/sec. at distances up to 300 m for the MIL-1553, the connection of the main node computers of the control system at the front-end level may be done without a LAN 802.5, directly on the system bus of the Multibus II with a transfer rate of 40 MByte/sec and full message passing mode. In such a way the possibilities of real time control are widened allowing for this purpose the use of interprocessor transfer in the Front-End Computing of the upper level.

The control system architecture of the K4-K10 allows the use of the standard operating system UNIX and SORIX for the node computers. The main programming languages are C at the system level and NODAL [8] at the application level. The software at the lower control level [2] has to be as short as possible (up to 10 KByte because at this level one may use 8 bit microprocessors) and has to provide a good reactivity: up to 250 interrupts per second with quite long messages on the MIL-1553 and up to 20 Kbyte/sec. All the software on the lower level is part of the system data base. It will be picked out together with all tables of parameter sets necessary for the

real time operation and will be stored into the equipment controllers at the lower level.

III. CONCLUSION

The control system architecture for the accelerator complex K4-K10 represents an example of an open architecture system based on the use of control and communication software and hardware corresponding to international standards. Though the system designers have the possibility to introduce their own original technical solutions, the main effort will be directed to the software and hardware design concerning the interfacing part of the system, i.e. the hardware and software ensuring the application of system resources as some kind of a set of control facilities.

REFERENCES

[1] R. Steiner et al.: The GSI Control System. Europhysics Conference on Control Systems for Experimental Physics. Villars-sur-Ollon, Switzerland, page 21-25, CERN 90-08, 12. 10. 1990.
[2] R. Rausch: Control System Architecture of the European Accelerators LEP and SPS. CERN SL/90-109 (C0).

[3] S.Kurokawa et al.: The Status and Prospects of the TRISTAN Controls System. Europhysics Conference on Control Systems, page 15-21, CERN 90-08, 12. 10. 1990.

[4] SIEMENS SIMICRO, Industrial Microcomputers SIMICRO SX. Catalog MC, 4. 1. 1991.

[5] A. N. Aleshov et al.: Accelerator Control System at the INP. Europhysics Conference on Control Systems for Experimental Physics, Villars-sur-Ollon, Switzerland, page 25-30, CERN 90-08, 12. 10. 90.

[6] D. Francart et al.: MIL-1553-B Multidrop Bus for Controlling LEP and SPS Equipment. LEP Control Note 56 rev.

[7] V. Kotov, L. Dubovic, R. pose: Automation and Control System for a cτ Factory. Proceedings Workshop cτ Factory in Dubna 1991.

[8] M. C. Crowley-Milling and G. C. Shering: The NODAL System for the SPS. CERN 78-07 SPS Div., 3. 9. 1978.

Future Directions in Controlling the LAMPF-PSR Accelerator Complex at Los Alamos National Laboratory*

R. Stuewe, S. Schaller, E. Bjorklund, M. Burns,
T. Callaway, G. Carr, S. Cohen, D. Kubicek,
M. Harrington, R. Poore, and D. Schultz
Los Alamos National Laboratory
Los Alamos, NM USA 87545

Abstract

Four interrelated projects are underway whose purpose is to migrate the LAMPF-PSR Accelerator Complex control systems to a system with a common set of hardware and software components. Project goals address problems in performance, maintenance and growth potential. Front-end hardware, operator interface hardware and software, computer systems, network systems and data system software are being simultaneously upgraded as part of these efforts. The efforts are being coordinated to provide for a smooth and timely migration to a client-server model-based data acquisition and control system. An increased use of distributed intelligence at both the front-end and the operator interface is a key element of the projects.*

I. INTRODUCTION

The integration of the Los Alamos Meson Physics Facility (LAMPF) and the Proton Storage Ring (PSR) control systems is presenting a series of problems for the operations and support personnel using the two systems. The two systems were developed independently using different personnel, different underlying philosophies and different equipment but developed interdependency when the operating and support groups were combined in 1988. A detailed discussion of the current control systems is presented in a companion paper in these proceedings.

II. PROBLEMS AND IMPACT

A. LAMPF RICE System

The LAMPF Control System (LCS) was built upon the LAMPF-designed Remote Instrumentation and Control Equipment (RICE) System. RICE is the hardware and software interface between the actual accelerator devices, such as magnets and beam-line instrumentation, and the software that operators and developers use to control beams.This system is illustrated in Figure 1. RICE presently utilizes 73 of 80 possible modules handling 10,000 data and control points distributed along approximately 2 km of beam channels.
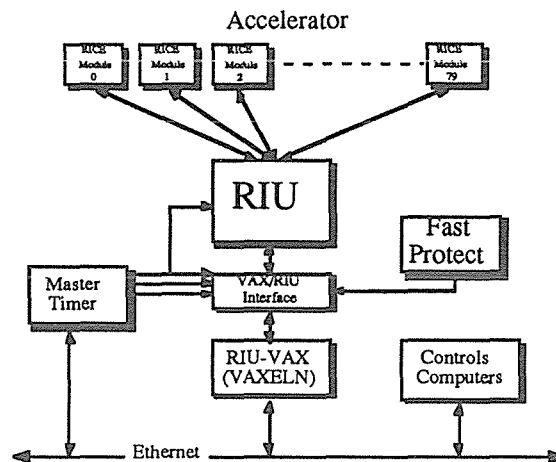
Figure 1. The LAMPF RICE System.

The RICE system has several limitations. First, RICE allows only one timed data request per beam pulse. At the low repetition rates characteristic of tuning beams, this feature can cause requests for device readout to become deeply queued and can cause tuning mistakes as operators react to "old" data. Second, the RICE star architecture places limits on the maximum data rates. Third, the RICE system is fairly rigid. For example, the current implementation of the linac harp system permits data from only two harps at any time and requires availability of greater than 50% of the RICE system to obtain that data. Fourth, there is a need for higher accuracy and precision than is available with RICE. Finally, it is estimated that 30% of the parts in the RICE system are no longer commercially available.

B. PSR ISS System

The PSR Control System is based upon a series of PDP-11 computers known as Instrumentation Sub-Systems (ISSes) which communicate with a central VAX system using serial CAMAC [1]. This is illustrated in Figure 2.
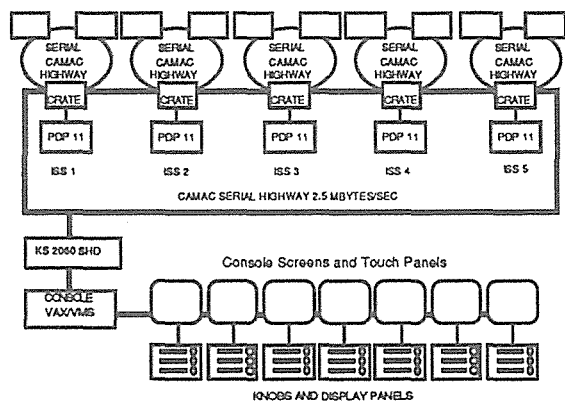
Figure 2. The PSR Control System.

The most significant difficulty with the PSR Control System is that the current update rate is one-fifth of what is considered optimal. This rate is directly limited by the system architecture which migrates all data continuously through the control system CAMAC loops to maintain the system's real-time database. Additionally, the complicated and time-consuming procedures required to modify the PSR database and the PDP-11 software discourage changes during run cycles, restricting the flexibility to correct problems and add devices. Finally, the PDP-11 database size limits are being approached on all five computers now that the system is twice its original design size.

C. LAMPF and PSR Operator Interface Systems

In the LAMPF control system, data display and entry is handled using color CRTs, graphics scopes, button panels, terminals, trackballs and knob panels [1]. Experience over the last two years has demonstrated that the demands for cpu power and data generated by consoles can easily outstrip any central control computer's ability to service the demand. This indicates that additional computing power is required to support future console demands. In the PSR control system, data display and entry is accomplished using color graphics screens, touch panels, and knob panels. The color graphics systems are creating an increasing maintenance burden due to difficulties in obtaining parts and adequate support from the manufacturer. The PSR touch panels are an inefficient and often ineffective method for interfacing to the control system. The graphics software of both systems is created at a fairly primitive level. This prohibits quick prototyping of application codes and increases the overall time required to create an application program.

## III. GOALS AND CONSIDERATIONS

A. Performance Goals

Goals have been established for the performance elements capacity and operator console response time. In terms of

capacity, the upgrades should provide the needed capacity to support twice the number of existing consoles and twice the number of data and control points in the system and increase the overall data throughput by a factor of three to 450 untimed read requests per second. Response time should be improved to provide a 5 Hz update rate at the operator consoles during normal operation and human speed, 1-2 Hz, beam profile information.

B. Availability and Maintainability Goals

Availability is a measure of the combination of failure rate and repair rate. Maintainability is the level of resources required to achieve a given availability. The availability goal for the upgrades is specified as 99.7%. This represents an improvement over the current systems which operate at approximately 99.4%. For these purposes the system is said to be unavailable if a failure occurs that prohibits planned beam tuning, development or production. To achieve this overall availability, the Mean Time To Failure (MTTF) goal is specified as 1 week, up from the current 4.2 days, and the Mean Time To Repair (MTTR) goal is specified as 0.5 hours, reduced from the current 1.3 hours. Achievement of these goals would provide approximately 11 hours additional beam time per typical annual operation period.

In order to support the capacity goals, the upgrades are being designed with maintainability in mind. Currently 9.5 software personnel support 21 control computers accessing approximately 16,000 data and control points. The upgrades specify as a goal that the current staffing level must be able to support a minimum of 40 front-end computers accessing up to 32,000 data and control points. To achieve this, the number of hardware diagnostics, software diagnostics and software tools must increase by at least a factor of two and the number of distinctly different hardware systems utilized must be similarly be reduced by a factor of two.

C. Considerations

Several specific considerations are being folded into the upgrade efforts. The first is long-term growth potential. This is directing effort toward the use of stable vendors and recognized and developing software and hardware standards. A desire for flexibility and robustness is to be addressed through increased modularity in both software and hardware. It would also be desirable if the systems were designed such that beam line developers could rapidly prototype and develop their own application software to reduce the support required by controls staff.

A major consideration of the proposed plan is the desire to migrate from proprietary systems, languages, and communication protocols to non-proprietary "open system" standards in all areas. For operating systems, the intent is to migrate to the POSIX open system standard when it is implemented. For communications, the Open System Interconnection (OSI) standard DECNET Phase V will be

preferred if it meets performance needs. Portable languages such as FORTRAN, C and C++ will be preferred.

IV. THE SOLUTION MODEL

To address the problems a simplified model of the control systems was used. This model, shown in Figure 3, is a statement of the elements and interfaces which must be standardized if the desired results are to be obtained. The model illustrates four levels; Data Acquisition and Control Computers, Communication Systems, Applications Systems and Operator Interface Systems.
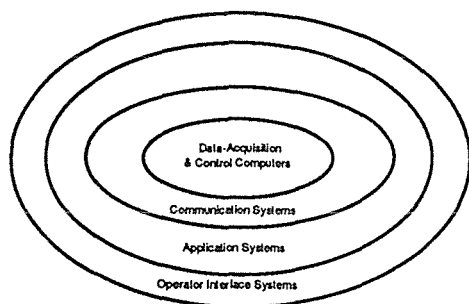

Figure 2. The Solution Model

A. Data Acquisition and control computers

This element of the model refers to the remote computer, hardware interface and low level software used to connect the front-end hardware to the communication medium.

B. Communication Systems

This element represents the hardware and software support required for a network based distributed computer control system. It provides the necessary interface between application software and the remote computers that actually acquire data and manipulate control points.

C. Application Systems

This element represents the algorithms and software systems that relate operator interface tools and beam line elements in a functionally useful manner. Physically, this element is an interface between the operator interface tools and the communication system.

D. Operator Interface Systems

This is the set of hardware and software mechanisms established to provide the user with the ability to interact with application software and thus the beam line instrumentation.

V. THE SOLUTION IMPLEMENTATION

The application systems, communication systems and remote data acquisition and control computers are already established in the form of existing LCS elements. The additional solutions developed must integrate well with both the existing and developing systems to preserve maintainability goals.

A. Data Acquisition and Control Computers

This element will be provided by the proven LCS VAXELN-CAMAC system. It is intended that this element provide both a RICE replacement and the PSR PDP-11 computer replacement.

VAXELN is the preferred remote computer operating system for the upgrade at this time. It provides a large array of development tools, integrates well with the current system and has an apparently solid upgrade path for the next 5-plus years. This implies a commitment to VMS, the development system for VAXELN, for at least this period. The vendor has indicated that both VMS and VAXELN will be POSIX compliant in the future, providing open system compatibility.

As stated, the device interface will be CAMAC, currently in use in the PSR system and in a significant portion of the LAMPF control system. As the PDP-11 computers are replaced, existing applications software will be modified to use a data-on-request approach thereby eliminating the resident PSR database and the current PSR control system over time. As RICE modules are replaced, the hardware that is removed will be used as spare parts to support the remaining RICE modules.

Experience with VAXELN controlled CAMAC systems provides confidence that the performance goals of the upgrade can be met. VAXELN/CAMAC driver testing has determined that CAMAC reads can be accomplished in 35-50 microseconds. A small number of these systems distributed throughout the site can meet the current and predicted data acquisition requirements.

B. Communication Systems

The communication system is intimately tied into the performance question. The LCS Remote Procedure Call (RPC) System will be utilized as the basic communication mechanism. Controls staff have performed, and are continuing to design, network performance tests to evaluate the system.

The level of network traffic anticipated for the new architecture has been estimated through measurement of peak demands on the current network with allowance for future expansion. Tests indicate that the current Ethernet/DECnet network is adequate for the near-term future even with the addition of the 10-20 computer nodes predicted for the RICE and PSR Upgrades. The effects of the 5-10 additional nodes needed to support proposed linac upgrades are currently being considered. An Ethernet/DECnet-based network provides a

standard that is compatible with a variety of computer platforms and operating systems.

While a simple network architecture will accommodate current projections, future requirements for segmentation must be included in planning. The addition of increasingly powerful computers to the network is the most likely future growth path and problem. When the networks are no longer cpu power limited, segmentation, through the use of bridges and routers, and new network technology such as FDDI will be required to provide solutions.

C. Application Systems

An effective application system is dependent upon the standardization of certain sub-elements and the development of a consistent "application viewpoint" of the machine.

The manner in which an application views its relationship to the data it desires is a critical element. The current PSR applications view data as continuously available since the system uses a continuous polling mechanism. This contrasts with the data-on-demand LCS applications. The upgrades recognize that a system that combines polled-data at the remote level with demand-data at the application level may be required to provide the performance that is desired and adapt to the conflicting application viewpoints. A version of such a system is being used in the RIU-MicroVAX [1]. This system continuously polls frequently requested data so that values can be supplied from a real-time database for this data. Less frequently requested data is read from the hardware when demanded.

Questions surrounding application requirements for device control locks, access to global data, data integrity and error handling remain to be addressed. These issues are further complicated in the highly distributed system that is envisioned.

Concurrent with standardization of the applications systems, the standards that are in place for control system software development will be re-examined These include requirement, design and user documentation procedures as well as configuration management systems.

D. Operator Interface Systems

It is believed that VAX-based workstations will provide the common interface hardware to replace both PSR and LAMPF console systems. The selection of VMS systems is to provide compatibility with the current VMS-based application and system software. Efforts are underway to evaluate user interface management systems against operator and developer preferences. The rapid development of commercial software tools in this area promises to assist in this effort. The first step in this process is a planned emulation of the existing LAMPF color CRT. This will provide a common interface for all parts of the installation and provide computing power to off-load the central control computer, thereby improving performance. This distributed interface system will similarly

contribute to an increase in control system availability by reducing the number of potential single points of failure.

VI. SUMMARY OF CURRENT EFFORTS

Several projects have been established to provide the upgrades to the systems as they have been described. The projects are the RICE Upgrade, the PSR ISS Upgrade, the Operator Interface Upgrade and the LAMPF Database Upgrade. Obviously, the pieces are not separate and must interact closely with each other to provide the standard elements of the proposed model. Existing resources and prioritization of the various problems are driving forces in selection of the solutions. Effort is already underway through work on all of the projects at various levels.

The RICE Upgrade has completed the requirements phase and is moving into design. Network and driver testing has been performed to insure that the systems selected will be capable of meeting the performance requirements. Evaluation of computer platforms is underway. The initial steps in defining a diagnostic system that will adequately support the new data system have been taken. Part of this effort is to design a CAMAC-based timed data system to replace the functionality of the RICE system.

The PSR ISS Upgrade is in the requirements phase, working rapidly in order to provide feedback to the RICE Upgrade project to insure that incompatibilities are not designed into the new front-end system. Efforts at defining the scope and complexity of the project are ongoing.

The Operator Interface Upgrade is also in the requirements phase. Much preliminary work has been performed. This work involved evaluating commercial Graphical User Interface's (GUI's), evaluating X-Windows as a graphics tool, determining what solutions are being used at other accelerator facilities and defining LAMPF control system user preferences.

The LAMPF Database Upgrade is in the requirements stage, although significant effort has been expended to bring the LAMPF database philosophically closer to what the PSR system requires. The LAMPF database can now support PSR devices, a necessary first step to integrating the systems [1].

A steering committee within the controls section has been established to coordinate the efforts. Quarterly internal project reviews and an external review of the combined projects are used as a means assuring the quality of the effort. The goals described in this document will be used as a measure of successful project completion.

VII. REFERENCES

[1] S. Schaller, R. Stuewe, et. al.,"Experience Controlling the LAMPF-PSR Accelerator Complex," these proceedings.

[2] G. Carr, S. Schaller, et al., "The Status of the LAMPF Control System Upgrade," Proc. Europhysics Conference on Control

Common Control System for the CERN accelerators

R. Rausch, Ch. Serre, editors
CERN - 1211 Geneva 23 - CH

Abstract

The PS and SPS Accelerator Control Systems are becoming obsolete and need urgent rejuvenation. After a control users forum, where users expressed their needs, two main Working Groups were set up, consisting of Control and Equipment Specialists and experienced Machine Operators. One Working Group studied the architecture and the front-end processing and the other a common approach to the application software needed to run the CERN accelerator complex. The paper presents the technical conclusions of their work and the policy to implement it, taking into account the necessity to operate both machines without interruption of the Physics Program.

I. INTRODUCTION

The complex of CERN accelerators is divided in two sets of different characteristics. The PS set is constituted of ten different accelerators which represent the source of all particles accelerated in CERN (maintained by PS division); they are small and mainly fast cycling machines. The SL set is composed of two bigger machines, the SPS and LEP which are slow cycling accelerator or particles colliders (maintained by SL division). The two sets are, broadly speaking, separated by the Swiss-French frontier.

The PS and SPS accelerators control systems were conceived and implemented some 15 years ago. They are based on 16 bit computers, with a proprietary operating system and a star network. These components do not permit the use of modern industrial software packages and communication standards. Their maintenance is expensive and is becoming more and more difficult. The consolidation of the control systems has become necessary and urgent, and it was felt that one should profit from this consolidation to aim at a real convergence of CERN's accelerator control systems.

In order to work out a common technical solution, the collaboration between the PS and SL control groups has been reinforced considerably since the beginning of the year 1990. A common consolidation project is the result of this collaboration and it was elaborated by working groups of the two divisions. Joint working groups were set up to study the different aspects of the project and to reach the necessary consensus on what should be done. [1]

The first working group designed the common control system architecture, the front end processing and discussed the network characteristics, the local control facilities and the interface between the controls and equipment groups.

The second working group defined a common approach to the application software needed to run the accelerators, discussed the programming environment and the possible software tools and studied the future layout of the work place in the control rooms.

Other groups, linked with the two previous ones, worked on specific subjects like the Equipment Control Protocols, the common on-line data base, the Man Machine Interface, the Timing and Synchronization problems. They generally worked out common solutions which are today in the implementation phase.

II. ARCHITECTURE

The new Common Control System consists of three layers: Figure 1
- the control room layer with its consoles and central servers;
- the front end computing layer distributed around the accelerators;
- the equipment control layer with the Equipment Control Assembly (ECA) crates which form part of the equipment.

The hardware and software used on each level reflect the considerable variety of accelerator components to be controlled. The new architecture offers more flexibility and will allow continuous partial upgrading as technology evolves.

A. Control Room Layer

It must fulfil two main functions:
- Provide the operators with a reliable, user-friendly interface to the accelerators. Modern workstations running the commercial software package X - windows with a suitable commercial tool kit to construct the user interface were chosen. The operator work places are very demanding in terms of graphic and interaction facilities. The key point in selecting the new platforms is the interoperability with the existing equipment and the portability of the software between them. We hope the choices made, UNIX operating system (OSF based in the future) with X-Window and Motif tool kit and communication by the TCP/IP protocol, will allow a smooth transition from one generation of hardware to another.
- Offer a number of central services through servers (which are generally more powerful machines of the same family as the workstations). These central services can be the coordination of various control tasks, central data and file storage, model computing and collection of alarms.
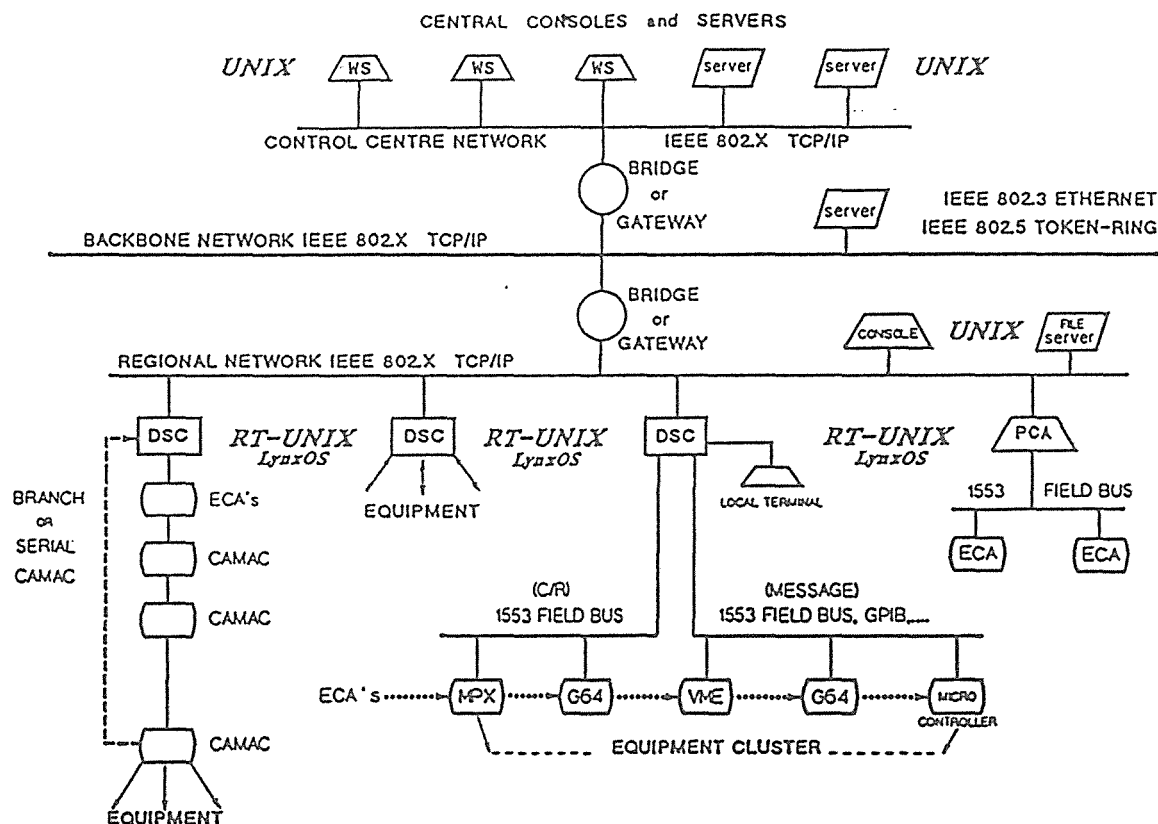
Figure 1. Control System Architecture

B. Networking

The communication between the Control Room layer and the Front End Computing layer, as well as the communication within these layers, is based on modern standards using TCP/IP as the main inter networking protocol suite. The network is composed either of Token Ring or Ethernet segments linked together through bridges or routers. However, where required, routers are implemented in order to filter the access to the control systems from offices, laboratories or the outside world.

The future use of FDDI as an additional fast network is in test between Meyrin and Prevessin sites (utilization by both control systems of an ORACLE on-line data base server).

A Remote Procedure Call (RPC) mechanism offers a way for the programmers to call the libraries located in remote computers, hiding as much as possible the network (CERN designed RPC including an interface to the old control network). TELNET and FTP are also available in connection with the TCP/IP protocol suite.

C. The Front End Computing layer:
The Device Stub Controller (DSC)

The Front End Computing layer is centered on the Device Stub Controllers (DSC) which are based on both standards PC

and VME crates with 32 bit embedded microprocessors. A diskless Real Time, UNIX compatible, and POSIX compliant, operating system is run in the DSC. The diskless solution was chosen because the disks of the Process Control Assemblies (PCA) of the LEP control system proved to be a weak point and because the back-up procedures and management of files and data are supposed to be easier if storage is less distributed.

The main functions of the DCS are:
- to provide a uniform interface to the equipment as seen from the workstations;
- to provide direct control and acquisition for equipment like beam instruments, interfaced directly to the DSC;
- to act as a master and data concentrator for distributed equipment, interfaced via a field bus.

The choice of the standards for the DSC, PC and VME bus, was not easy. With the limited resources (staff numbers decrease while the number of new projects increases) the Control groups initially intended to provide support for the VME bus only. However, after long discussions, because of the 65 existing LEP PCA (Process Control Assembly) based on PC and because of quite complementary advantages, it was decided to give support to the two standards, PC and VME bus. This decision mainly became possible with the availability of an open Real-Time, UNIX compatible, operating system which runs on both platforms. Figures 2 and 3

Figure 2. Device Stub Controller

This Real Time UNIX compatible operating system, LynxOS, has been chosen to allow running several tasks concurrently (beam measurement, 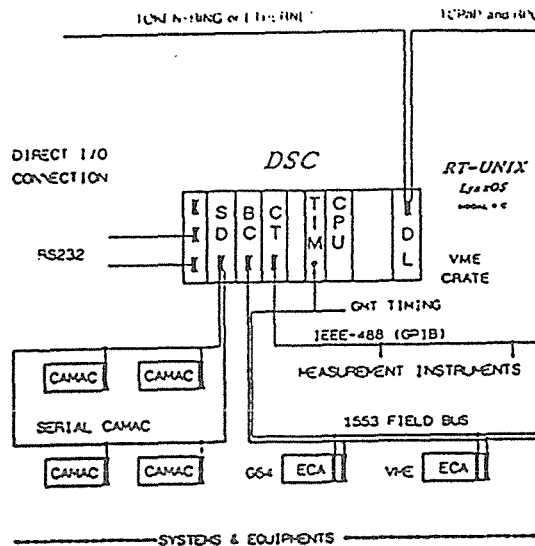statistics, alarms, diagnostic programs), and to provide a fast and deterministic response to external events when necessary (Pulse to Pulse Modulation for example). It will make it relatively easy to port the XENIX system libraries and servers developed for the LEP PCA to the new operating system.

To provide disk service for the diskless DSC and workstations (as well as boot server, secure disk space for application and data and back up facilities) file servers will be made accessible. Central file servers are used in general. However, in order to keep the load on the backbone network acceptable and to maintain local control alive in case of communication problems, it has been decided to implement regional file servers on important network segments.

Associated to the DSC, one can distinguish between three different types of local man machine interface:
- Local display (for the DSC based on VME): a graphic VME module is directly driven by the CPU. It could be completed by a touch-panel when a more convivial interface is needed for local operation.
- Local terminal: connected to the CPU of the VME crate with an RS232C line. This access can be used during debugging or testing.
- Regional console: a workstation, a PC with UNIX or a X-terminal, directly connected to the network segment, could be seen as a local use of the services available in the Control Rooms, and can be directly used as a local terminal with TELNET connection.

D. Equipment Control layer

The control crates of the third layer, the Equipment Control Assemblies (ECA), are connected to the DSC via field

buses. Since no predominant standard exists, a certain variety of solutions, both for the ECA and for the field buses, is considered to be acceptable. As the main front end computing device is supposed to be the DSC, the computing on the ECA level should be reduced to a minimum.

E. Field buses

In the present CERN accelerator Control Systems, three field buses are used to a large extent: the 1553 field bus [2] and the GPIB in the LEP area and serial CAMAC in the PS area and the SPS experimental areas. At the PS complex, the CAMAC crates will be controlled directly by a VME module that drives the serial loop [3] whilst in the SPS experimental areas the CAMAC crates will be connected to the VME or PC DSC via the VICbus. The SPS main ring equipments will mostly be interfaced to the 1553 field bus. Proprietary field buses delivered with turn-key industrial systems will be interfaced directly to the DSC. [4] Due to the large investment in the associated interface equipment, all three field buses will be supported in the DSC environment.

F. Support to Equipment groups

The equipment groups may need to develop software to control their equipment either in the DSC itself or at the ECA level. When the DSC is directly dedicated to an equipment through interface cards, the control group will offer assistance (cookbook, basic driver "frame") to write the drivers accessing the specific hardware. When the equipment is connected to the DSC through a standard field bus, the Control Group will
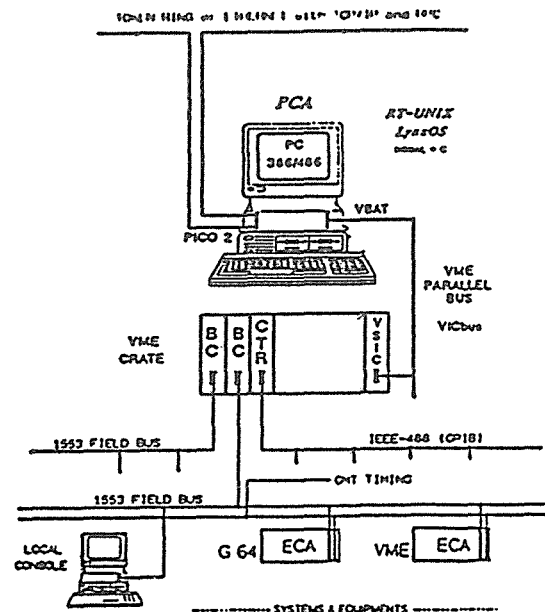

Figure 3. Process Control Assembly

provide the equipment access functions in the DSC. At the level of the ECA, the use of OS9 and LynxOS will get some support from the Control Groups.

G. Proposed Equipment Control Protocols [5]

The best result for the control of a given equipment is obtained when the Equipment Group is totally responsible for its implementation. It is proposed, and already in the beginning of implementation, to use a general operational control protocol which leaves the equipment specialists dealing with all the intricacies of their equipment (either at the level of the DSC or the ECA, where applicable), whilst the control groups take care of the operational requests for this equipment as well as the communication part.

In this scenario, the accelerator operation crew defines a uniform frame for a given equipment interaction, the specialist realizes the specific, local software in the most adequate way for each device, and the controls provide a uniform software interface to translate the operator requests into control sequences for the equipment.

III. APPLICATIONS

A. Basic buildings blocks and tools

The basic building blocks and the tools for the Applications are centered around Open Software Standards. UNIX is confirmed as the operating system of the workstations, the development environment of the DSC and the network servers. The DSC operating system is a UNIX compatible system with real-time performance. (LynxOS) TCP/IP protocol suite and NFS are the basic components for all distributed facilities. Compliance with standards is enforced. The POSIX IEEE 1003.1 interface definition must be respected. The X-Windows is selected as the base graphic system for workstations : for general data presentation facilities and user interactions, all future developments will be based on the X-Windows protocol and the OSF Motif toolkit.

B. Data Management

A well supported set of data management facilities inside the CERN accelerator control systems is needed to cope with the large quantity of data provided by modern electronic equipment and beam instrumentation. The study of the database services sub-group concluded that a dedicated PS/SL database server running ORACLE would be useful. The host machine for this service is installed and first tests and measurements concerning the performance of ORACLE as an on-line database are in progress.

C. Software development facilities

The use of modern CASE tools are encouraged for analysis and design of software projects, but one cannot expect to

enforce a systematic application for all projects and to all participants. Source code management tools provided with UNIX are available for keeping track of the history of consecutive versions. "C" is the common base language, as it is well integrated in UNIX platforms and offers a good portability (as accepted by the Portable C compiler). The NODAL interpreter (written in C) is available on the DSC and on both workstation platforms in use, DEC and Hewlett-Packard. Fortran may be used, but only for mathematical applications. (mainly modeling)

D. Environment of application software

The development environment should be as similar as possible to the operational environment in order to ease the transition from development to operation and to allow good productivity and maintainability by the knowledge of a single environment. Standard procedures must be established for validation by the users (operation, accelerator physicists, equipment groups) and controls exploitation staff. A complete application program must be designed with integration into the Console Manager in mind; the Console Manager is the supervisor of all activities induced by the workstation.

E. Application run-time environment

No synchronization with the cycle of accelerators will be done through workstations. The equipment cannot rely on software for its protection, and the equipment groups will have to provide means to protect their equipment against unallowed actions through the control system.

F. Man Machine Interface

The major goal is to reach as much uniformity as possible for the operation environment. To preserve the future software investment in MMI, the OSF tool kit MOTIF will be used on top of the well established and widely used X-Windows standard display protocol. The console manager activates and coordinates the various processes according to the requests of the user. Synoptics will be widely used in the controls of the accelerators (as they are also the natural tool offered for various industrial equipment); the DVDraw/DVtools product, encapsulated in a man machine interface, will be used for the production of such synoptics. [6] [7]

G. General Control room environment and operator desk

The operation of accelerator clusters through the control system is done by operator teams working 24 h a day in the central control rooms and using consoles as the main tool for machine interaction. The notion of "work place" is now preferred to the one of console. The powerful workstations are the main interactive tool composing a work place; they are the tools to select, visualize and drive the batch of application

programs to control a selected part of an accelerator. The work place is completed by tools to observe and select analog and video signals, and to display alarms.

Beside the workplaces, an accelerator control center must contain tools to display general information summarizing the status of the accelerator complex, means to communicate with local building and other control centers, means to access other network facilities, and tools to manage personal safety which must be treated separately from accelerator control.

IV. CONVERGENCE POLICY

Because of the large hardware and software investment needed, the limited manpower available today, the similarity of the control requirements for the various machines and the increasing complexity of the software tools for running them, it is mandatory to reduce the diversity of control implementations at CERN.

The constraints imposed on the convergence by the history of the different machines, the large amount of investment in existing hardware and software, the different types of machines to be controlled, and the habits adopted by the operation crews interacting with these machines must be taken into account when defining a common PS/SL policy.

This common policy will be based on a common control system architecture, using well established industrial standards for control networks, communication protocols, equipment, operating systems and man machine interface. This architecture will profit from and enforce the use of Open-System products supported by many manufacturers and consortiums. (UNIX, OSF, X-Windows for the software; Ethernet and Token-Ring with TCP/IP for the control network; UNIX workstations for the man machine interface; PC and VME for the DSC and equipment interfaces).

As much software as possible must be written independently of the development platform (using industrial products), opening the possibility to transfer application programs between the different accelerators.

An other main goal of the common policy must be to preserve the existing hardware and software investment. The application software developed for the exploitation of the accelerators represents hundreds of man-years of investment which has to be maintained, improved and optimized to ensure uninterrupted operation. The proposed architecture makes provision to integrate the hardware investment in CAMAC and MPX in providing adequate tools to access these crates. The software investment in the SL-PCA is preserved by using a UNIX compatible real time operating system in the DSC.

V. CONCLUSION

This is the summary of the consensus which has emerged from the discussions about the architecture of CERN's accelerator control systems, their main components and the general aspects of the application software. A major concern was to preserve the existing hardware and software investment, together with a non-duplication of the effort inside CERN's accelerator community.

Our main aim was to propose an architecture which permits continuous partial upgrading, as technology evolves, in order to avoid any "big bang" operation in the future. The main components of our control systems are based on open standards, for the hardware as well as for the software, to become independent of a given manufacturer.

Finally, to monitor the progress of the consolidation project and to ensure that all our efforts stay directed towards a common goal, the working groups (or at least an emanation of the two working groups) continue to meet during the implementation of the different steps of the project.

VI. REFERENCES

[1] The PS and SL controls groups, "PS/SL Controls Consolidation Project", Technical Report, CERN PS/91-09 (CO), CERN SL/91-12 (CO), April 1991.

[2] R. Rausch, "Control of the large European LEP and SPS Accelerators Based on the 1553 Field Bus", CERN SL/9005 Report, Conference on MIL-STD-1553-Band the Next Generation, London, United Kingdom, 29-30 November, 1989.

[3] W. Heinze, "Driving serial CAMAC systems from VME crates", these Proceedings, November 1991, ICALEPCS, Tsukuba, Japan, 1991.

[4] K.H. Kissler, R. Rausch, "New Control Architecture for the SPS Accelerator at CERN", November 1991, these Proceedings, ICALEPCS, Tsukuba, Japan, 1991.

[5] WOPRO Working Group, reported by G. Benincasa, Control Protocol : "The Proposed new CERN Standard Access Procedure to Accelerator Equipment : Status Report", these Proceedings, November 1991, ICALEPCS, Tsukuba, Japan, 1991.

[6] M. Boutheon, F. Di Maio, A. Pace, "General Man-Machine Interface used in Accelerators Controls: some applications in CERN/PS Control System Rejuvenation", these Proceedings, November 1991, ICALEPCS, Tsukuba, Japan, 1991.

[7] P. Antonsanti, M. Arruat, J.M. Bouché, L. Cons, Y. Deloose, F. Di Maio, "Workstations as consoles in the CERN/PS complex: setting up the environment", these Proceedings, November 1991, ICALEPCS, Tsukuba, Japan, 1991.

New Control Architecture for the SPS Accelerator at CERN

K.H. Kissler and R. Rausch

European Organization for Nuclear Research

CERN - 1211 Geneva 23 - CH

Abstract

The Control System for the 450 Gev proton accelerator SPS at CERN was conceived and implemented some 18 years ago. The 16 Bit minicomputers with their proprietary operating system and interconnection with a dedicated network do not permit the use of modern workstations, international communication standards and industrial software packages. The upgrading of the system has therefore become necessary.

After a short review of the history and the current state of the SPS control system, the paper describes how CERN's new control architecture, which will be common to all accelerators, will be realized at the SPS. The migration path ensuring a smooth transition to the final system is outlined. Once the SPS upgrade is complete and following some enhancements to the LEP control system, the operator in the SPS/LEP control center will be working in a single uniform control environment.

I. HISTORIC REVIEW

The SPS control system was designed in 1972 and brought into operation in June 1976. By that time no standard communication network protocol existed and pioneering work had to be done to interconnect initially some twenty minicomputers located in 6 equidistant equipment buildings and one central control room around the 7 Km circumference of the SPS accelerator ring. The minicomputers used were NORD10 from Norsk Data with 16 KByte core memory and 128 KByte drum mass storage. The equipment interface consisted of CAMAC crates connected to the computer's I/O bus with CAMAC modules linked directly to some of the beam instrumentation while all other equipment was controlled via a CERN designed multiplex (MPX) system composed of a serial field bus, MPX crates and user dedicated MPX modules.

On the software side, the manufacturer's operating system has been modified to suit the particular real-time control requirements of our distributed multiprocessor environment. While most of the software drivers were written in computer assembly code, an interpreter, called NODAL, has been developed to provide easy interaction between the operator and the equipment connected to CAMAC, remote access facilities and network functions. Every computer had a resident NODAL interpreter allowing to use it in interactive mode and in stand alone operation for test and commissioning purposes. No one computer would be the over-all master of the control system and the message transfer system was designed so that any computer could pass a message to any other without a preset master-slave relationship, implying that the system was completely symmetrical and transparent [1].

II. PRESENT SITUATION

Since the start-up of the SPS in 1976, the control system has been extended to cope with the changing requirements of the accelerator which was initially designed as a pulsed proton accelerator for fixed target experiments, then modified to act as a proton/antiproton storage ring for collider physics, and now also as an injector to LEP, accelerating electrons and positrons interleaved with proton acceleration. Such evolution has required a great flexibility of the control system with the ability to modify programs as necessary in a simple way and to add computers, network links and equipment interfaces where and when required, sometimes even during the exploitation of the accelerator complex.

Today the SPS control system is composed of 52 operational process and central computers (NORD100) interconnected by a multi-star network with 6 Message Handling Computers (MHC). The interfacing between computers and accelerator equipment is done by CAMAC (72 crates and 450 modules) and by the MPX system (693 crates and 5500 modules). Figure 1.

With the need to operate the SPS in a supercycle mode when using it as a LEP injector, major additions had to be made to the control system since about 1986. A more flexible and versatile exploitation of many accelerator components like beam monitors and pulsed power converters was required. At the equipment level, this requirement has led to the use of 8 Bit and 16 Bit microprocessor based systems, embedded in G64bus and VMEbus crates. These Equipment Control Assemblies (ECAs) are connected to the appropriate NORD100 process computer via 1553 field bus segments and a VMEbus crate housing the bus controllers and linked to the computer's I/O bus.

In the accelerator control room, Apollo workstations where installed to cope with the more complex supercycle operation. These communicate with each other and with an

Apollo file server via an IBM Token-Ring network. A gateway from the Token-Ring to the old proprietary star network of the SPS permits the Apollos to communicate with the NORD100 computers.

Figure 1. Present SPS Control Network.

The SPS was also used as a test bed for the LEP control system [2]. To this end a full network infrastructure including a backbone Token-Ring around the accelerator and a Time Division Multiplex (TDM) system for long distance transmission were installed at the SPS well before this was possible in the LEP tunnel. All SPS auxiliary buildings were equipped with LEP type Process Control Assemblies (PCAs). Each of these consists of a PC linked to a VMEbus crate via a 1553 connection, the VMEbus crate containing the bus controllers for the 1553 field buses which extend to the equipment. This infrastructure has permitted to validate the technical choices for LEP, but will now also serve in the framework of the new SPS controls.

III. CERN's NEW CONTROL ARCHITECTURE

The new control system architecture on which the PS and the SPS/LEP control groups have agreed after extensive studies and consultation with the control system users, is laid down in a comprehensive report [3] and presented in some detail in another paper at this conference

[4]. Here only a short summary will be given to make this paper self-contained.

A. Architecture

The new architecture is based on three computing layers:

the *Control Room Layer* which provides the operator with a user friendly interface using modern UNIX workstations with OSF Motif and X-Windows graphic software packages. This layer comprises also a number of central servers for data and file storage, model computing, alarm collection, network management and for an on-line relational database.

the *Front End Computing Layer* with its so-called Device Stub Controllers (DSCs) based on Personal Computers (PCs) and VMEbus systems, running a POSIX compliant real-time operating system. Secure disk space is provided locally or centrally by dedicated file servers. Local access and graphics are standard features of PCs and are provided on VMEbus crates by terminals and displays connected to dedicated modules. Alternatively, a local terminal server or an X-Terminal, connected to the Ethernet segment, will provide access to all computers (local or remote) on the control network.

the *Equipment Control Layer* with Equipment Control Assemblies (ECAs) connected to front end computers via field buses and to the equipment by mean of general purpose or dedicated electronic modules. Branch or serial CAMAC and 1553 field bus will allow to integrate existing CAMAC and MPX crates at this level. Figure 2.

B. Network and Communication

The data communication between CERN's different Main Control Rooms will be done via a Fiber Distributed Data Interface (FDDI) network. The FDDI is a Local Area Network (LAN) protocol defined by the ANSI X3T9 Standards Committee. The network has a ring topology, uses token passing access, a fiber optic transmission media, provides a 100 Mbit/s data rate, can span distances up to 2000 meters and supports up to 500 network nodes. This FDDI network will allow to share common PS-SPS servers and an ORACLE data base computer.

The communication within and between the Control Room and the Front End Computing Layers is based on modern LANs: Ethernet and Token-Ring conforming to IEEE 802.X International Standards and using the TCP/IP protocol. Program communication will rely on a CERN designed Remote Procedure Call (RPC) until a standard is defined and industrial products become available. Where long distances are involved (>500 meters) a Token-Ring backbone is implemented with transmission over TDM equipment conforming to the CCITT G700 Standard.
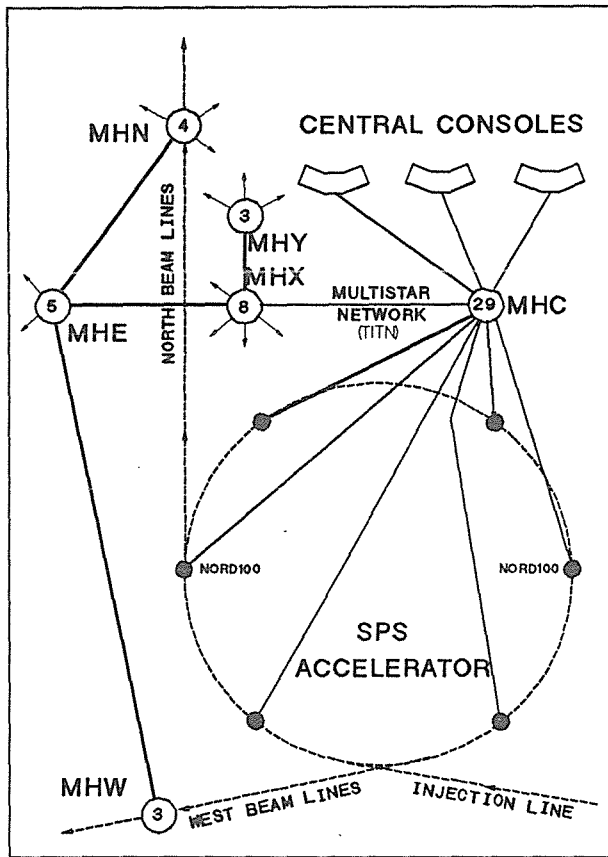
Figure 2. New Control Architecture.

C. Operating Systems

It has been decided to use a UNIX environment for the control of CERN's accelerators to the largest possible extent.

UNIX is the only operating system supported by virtually all computer manufacturers, from the PC clones up to the large CRAY or IBM mainframes. UNIX provides *scalability* which allows to move programs and data from smaller machines to larger ones. UNIX avoids getting trapped in a proprietary environment: overcharges, poor services, failure to provide an upgrade path, company which goes bankrupt, etc.. In this case the *portability* of UNIX permits to move programs and data to another brand of hardware. For the programmers it is easier to port an application between any two versions of UNIX than any two closed operating systems. In addition, combining UNIX with a public networking standard (TCP/IP), it provides *interoperability*; the ability of networked computers to share files and applications.

For the front end process computers, it has been decided to use a commercial real-time operating system, which runs on both industry standard 386/486 PC/AT compatibles and VMEbus CPU modules, based on the Motorola 68030 microprocessor, and which complies with the IEEE 1003-(1990) POSIX standard (POSIX.1 for the application program interface, POSIX.2 draft for shell and utilities and POSIX.4 draft for the real-time extensions).

The acronym POSIX stands for Portable Operating System Interface. POSIX is written by IEEE working groups as a US standard and acquires international status via its acceptance by ISO/IEC (International Standard Organization International Electrotechnical Commission) as International Standard 9945, [5].

POSIX is a software interface standard which guarantees portability of application source code but is not an operating system implementation standard. Real-time POSIX addresses the full range of real-time systems, from full scale UNIX down to small embedded kernels with the highest demands on true real-time performance.

It is well known that basic UNIX is not designed to be a real-time system. Its goal is to give a fair share of system resources to every user, not to have, for instance, a high priority process taking the CPU and keeping it as long as necessary. On the contrary, real-time POSIX ensures an operating system the ability to provide a required level of service in a bounded response time.

Following a CERN tendering procedure LynxOS has been selected.

IV. IMPLEMENTATION OF THE NEW ARCHITECTURE AT THE SPS

A. General consideration

CERN's new control system architecture described above defines the general framework and the guide lines to be respected when replacing the old SPS system. It leaves, however, the freedom to take account of other facts proper to the SL division: the existence of and experience with the LEP control system as well as the infrastructure created during the preparation of the SPS as a LEP injector. Both facts, in particular the experience with the LEP system and the desire to create a uniform controls environment for the two accelerators, have had and are still having a strong influence on the detailed choices made for the SPS.

B. The Control Room Layer

SPS and LEP are both operated from the same control room. Apollo workstations running under Domain OS 10.3 are currently used to provide the operator interface to LEP and also to control the supercycle operation of the SPS. More powerful machines of the same workstation family offer central services, such as data and file storage or model computing. Recently, Hewlett-Packard workstations, model 9000/400 and 9000/425, still running under Domain OS, have been added to the system.

We are now in the process of introducing a few DEC 5000/200 workstations, running ULTRIX 4.2, into the SL control system. Their main function is to provide centralized secure disk space for applications and data, to provide disk service for diskless front end processors and to be the bootstrap servers for front end processors, workstations and X-terminals. The DEC stations were mainly chosen in the framework of the collaboration with the PS division, permitting to set up common services like a common ORACLE database.

In the near future, HP 9000/730 and 9000/750 workstations will make their appearance at SL. Running HP UX 8.05, they will add to the diversity of UNIX flavors used at the level of the SL control room layer. This diversity is considered to be temporarily acceptable and should largely disappear with the future introduction of the OSF 1 operating system for both DEC and Hewlett-Packard workstation families.

While the general data presentation and user interaction in the SPS/LEP control room are presently still based on the Apollo proprietary user interface management system Domain/Dialogue, all future developments will make use of the X-windows protocol and OSF Motif (toolkit, User Interface Language, resource manager, style guide).

C. The SPS control network

As mentioned before, a backbone Token-Ring was installed around the SPS a few years ago already, completed by local Token-Rings in the main control room and in the SPS auxiliary buildings. To this installation, local Ethernet segments will now be added in order to guarantee a better connectivity of equipment from different vendors to the network.

D. Front End Computers

All NORD100 computers will be replaced progressively by front end computers based on PCs or on VMEbus crates depending on the type of equipment they control, on the necessity to have local mass storage, graphics facilities and local interactivity for testing purposes. It is essentially the responsibility of the Equipment Group in charge to decide which technical solution is best.

Typically, equipment like beam instrumentation which needs fast response and high throughput, will be directly connected to modules located in VMEbus front end computers. These in turn will be connected to the local Ethernet segment and share either a local or a central file server. Such an arrangement has recently proved to be very successful at LEP where some 40 VMEbus crates with direct connection to the network are used for closed orbit correction [7].

For systems with large numbers of identical Equipment Control Crates (ECAs), PC based front end processors will be used to regroup the ECAs via a 1553 field bus and enable local supervision, alarm reduction and alarm identification before transmission to the main control room. The Process Control Assemblies (PCAs) described in chapter II will be the most frequent configuration used in this context.

Local consoles, PCAs and VMEbus based front end computers installed in the equipment buildings will operate under LynxOS and will be able to run the same programs as the central workstations and servers located in the main control room, all will communicate over the control network using the TCP/IP protocol.

E. Equipment Field Buses

In LEP, ECAs usually contain microprocessor units and are addressed in message mode via the 1553 field bus.

This mode of operation will also be used in the SPS whenever old obsolete electronics will be replaced by new modern boards housed in intelligent control crates [6].

However, many of the existing MPX crates are still working reliably and will be preserved. In this case the old MPX multidrop bus, connecting them via a CAMAC crate to the NORD100 computers, will be replaced by a 1553 field bus, generally controlled by a PCA. As no microprocessor is

used in the MPX system, the 1553 field bus will be working in command/response mode and a special 1553/MPX interface card has been developed for this purpose.

It is at the field bus level that the General Machine Timing (GMT) is distributed to the equipment. For economical reasons and cabling simplicity, both control and timing signals are transmitted over two twisted pairs of wires in the same multidrop cable [8].

Other field buses can be used at the SPS to link ECAs to the front end processors. Industrial systems are often delivered with a proprietary field bus: Bitbus, Fip, Filbus, J-bus, Profibus, Proway, etc.. To be integrated into the SPS control system these industrial systems must be delivered with a VMEbus or a PC/AT bus controller and an adequate software driver, compatible with the LynxOS real-time kernel, must be available. In addition, the manufacturer's equipment control protocol must be known and be converted to the CERN standard to allow homogeneous access to all equipment of the accelerator from the operator's workstation.

The experimental areas of the SPS present a special case where the existing field bus, a serial CAMAC loop, will be replaced by Ethernet. A PC connected to the local Ethernet segment will be installed in each of the stations which regroup the equipment control hardware. This hardware will continue to be controlled by CAMAC modules and their CAMAC crate will be linked to the PC by the parallel Vme Inter-Crate bus (VICbus).

V. THE MIGRATION PATH

A major step of the migration towards the new control architecture is planned to take place during the 1991-92 winter shutdown of the SPS.

The existing gateway between the old multistar network and the new Ethernet/Token-Ring infrastructure only allows communication in command/response mode from the Apollo workstations to the NORD100 computers. It will be replaced by a new bidirectional gateway which will enable the NORD100s to address to services on the new network. The library computer used for data and file storage and other NORD100 based servers, all operating with obsolete disk units, will then be removed and their tasks taken over by modern workstation based servers.
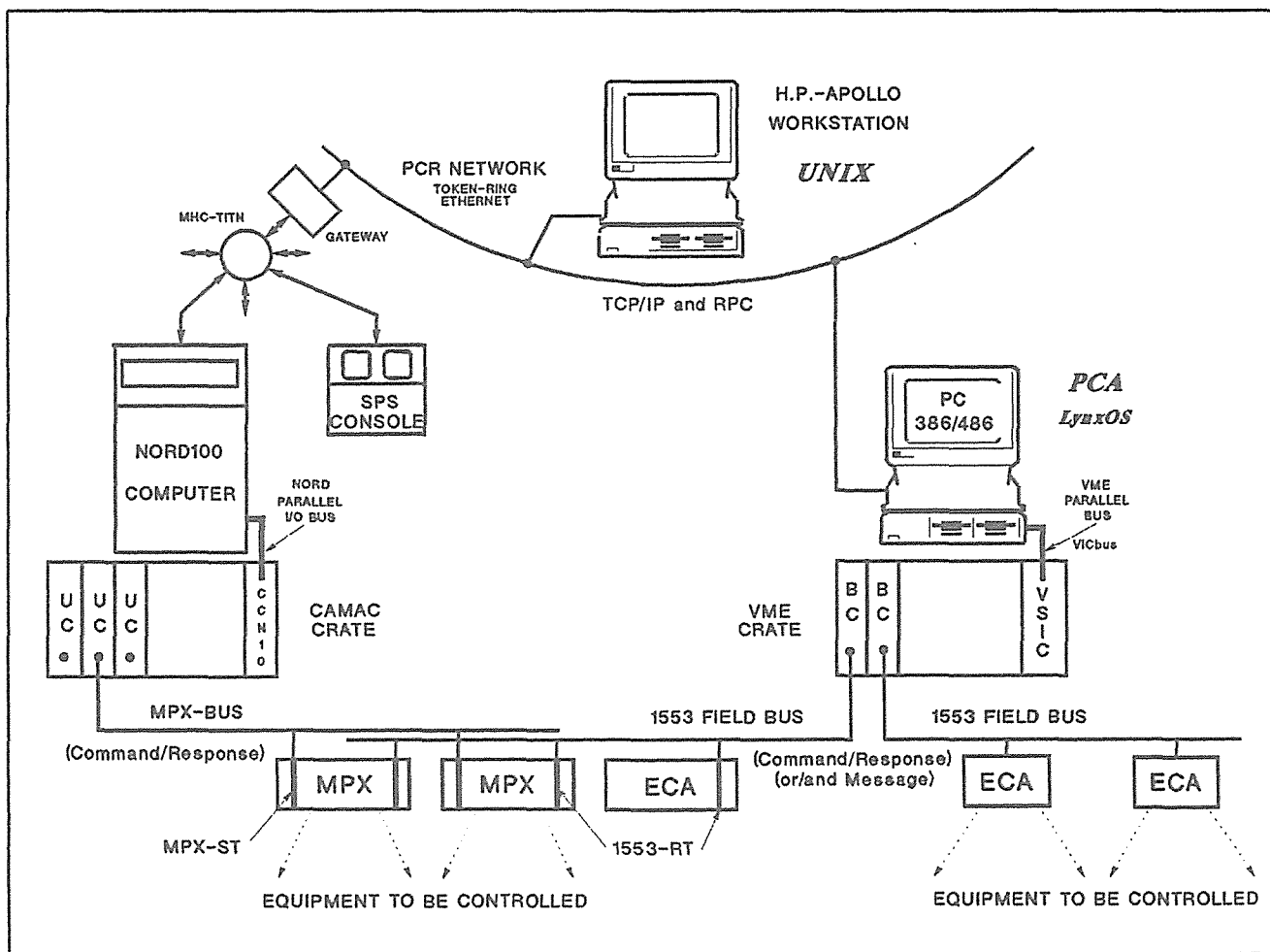

Figure3. Migration from NORD100 to PCAs

The new gateway will be installed during the month of October and put into operation during the last two months of the year. This new gateway will first operate in parallel with the old one to allow for software debugging during accelerator operation while keeping the possibility to switch back at any time in case of problems.

At the same time a new version of the CERN RPC (Remote Procedure Call) will be introduced as well as an improved version of the NODAL interpreter, written in C language, providing additional functionalities.

During this winter shutdown existing VMEbus based front end computers and PCAs will be connected to local Ethernet segments which in turn will be bridged to the Token-Ring backbone and all PCAs will be upgraded by using the VICbus connection to the VMEbus crates, instead of the 1553 link.

It is also foreseen to install LynxOS 2.0 in the PCAs, in replacement of the present SCO XENIX.

This upgrade combined with the general use of NFS (Network File System) and the hardware improvements described above is expected to provide a better overall performance and response time compared to that of the present LEP control system.

In the main control room DEC file servers have already been installed to provide secure file space for programs, applications, data storage and remote boot facility for the PCAs and the VMEbus based front end computers.

Existing Hewlett-Packard workstations will be used with X-windows and Motif to write new application programs for the systems linked to the new infrastructure. Thus the complete chain, from the operator workstation down to the ECAs including the network and the new front end computers can be tested and debugged.

To facilitate the migration of application programs from the old NORD100 consoles to the Hewlett-Packard workstations, for equipment remaining to be controlled by MPX crates, an alternative access will be possible for test and debugging purposes during the transition phase. Figure 3.

VI. CONCLUSIONS

Modernizing the control system of a running accelerator is a difficult task. Since its first operation in 1976 the SPS has undergone a number of smooth upgrades to cope with the changing demands of the accelerator exploitation. Generally more installation of the same kind of equipment or replacement of mini computers by more powerful ones of the same family, was fairly straightforward.

This time, a drastic change is required. Computers, networks, local intelligence, application and system programming have changed totally over the last 15 years. Hardware and software standards are available but the job has not become easier for the control specialists. New skills are required and system integration is the buzz word.

To preserve the integrity of the accelerator operation, the migration work must be done in well defined steps. The test bed chosen for the validation of the new architecture and the methodology adopted must be representative and include all aspects of the project. The overall planning is essentially determined by the duration of the annual accelerator shutdown and available manpower, particularly in the field of application software.

VII. ACKNOWLEDGEMENTS

Many members of the PS and SPS/LEP control groups have contributed to the definition of CERN's new control system architecture and to the application of the ideas within the SPS environment. The authors wish to thank them all. Particular thanks are due to J. Altaber and F. Perriollat who where co-authors of the first proposal to harmonize the controls of all accelerators at CERN.

VIII. REFERENCES

[1] M.C. Crowley-Milling, "The Design of the Control System for the SPS," CERN 75-20 Report, 22 December, 1975.

[2] P.G. Innocenti, "The LEP Control System," International Conference on Accelerator and Large Experimental Physics Control Systems, Vancouver, BC, Canada, 30 October - 3 November, 1989.

[3] The PS and SL Control Groups, (Editors: R. Rausch and C. Serre), "PS/SL Controls Consolidation Project," Technical Report, PS/91-09 or SL/91-12, CERN, Geneva, Switzerland, April, 1991.

[4] R. Rausch, C. Serre, "Common Control System for the CERN PS and SPS," International Conference on Accelerator and Large Experimental Physics Control Systems, Tsukuba, Japan, 11-15 November, 1991.

[5] C. Eck, "Standardization of Real-Time Software POSIX 1003.4," Real-Time 91 Conference, Jülich, Germany, 24-28 June, 1991.

[6] R. Rausch, "Control of the Large European LEP and SPS Accelerators Based on the 1553 Field Bus," CERN SL/90-05 Report, Conference on MIL-STD-1553-B and the Next Generation, London, United Kingdom, 29-30 November, 1989.

[7] G. Morpurgo, "The Software for the CERN LEP Beam Orbit Measurement System," International Conference on Accelerator and Large Experimental Physics Control Systems, Tsukuba, Japan, 11-15 November, 1991.

[8] C.G. Beetham, K. Kohler, C.R.C.B. Parker, J.B. Ribes, "The Transmission of Accelerator Timing Information around CERN," International Conference on Accelerator and Large Experimental Physics Control Systems, Tsukuba, Japan, 11-15 November, 1991.

The Next Generation Control System of GANIL

T.T. Luong, L. David, E. Lécorché, M. Ulrich
Grand Accélérateur National d'Ions Lourds
BP 5027 - F 14021 CAEN Cedex, FRANCE

Abstract

The existing computer control system of GANIL is being renewed to fulfil the increasing requirements of the accelerator operation. This medium term major improvement is aiming at providing the physicists with a wider range of ion beams of higher quality under more flexible and reliable conditions.

This paper gives a short description of the new control system envisioned. It consists of a three layer distributed architecture federating a VAX6000-410/VMS host computer, a real time control system made up of a dual host VAX3800 and workstation based operator consoles, and at the frontend segment: VME and CAMAC processors running under the VAXELN operating system, and programmable logic controllers for local controls.

The basic issues with regard to architecture, human interface, information management, ... are discussed. Lastly, first implementations and operation results are presented.

I. INTRODUCTION

The GANIL laboratory has been operating since 1983 an accelerator complex consisting of three machines in cascade : a compact injector cyclotron and two fourfold separated sector cyclotrons (Fig. 1).
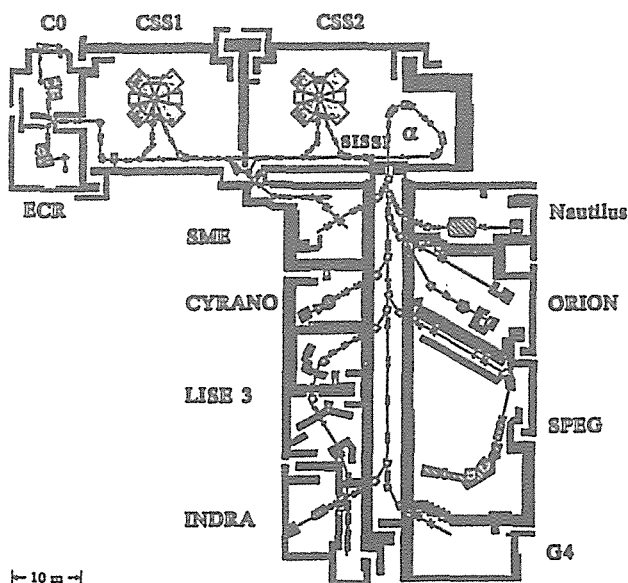

Figure 1. Accelerator and Experimental Areas

This facility provides the experimenters with fast heavy ion beams for fundamental research in the fields of nuclear physics, atomic physics and solid state physics, as well as for industrial applications.

Significant upgrades were carried out these last few years to augment the energy of heaviest ion beams and to increase their intensities by making use of new ECR source. Acceleration at GANIL henceforth encompasses ion species from carbon to uranium, with beam energy ranging from up to 95 MeV per nucleon for the ions with masses up to 40 u to 24 MeV per nucleon for the heaviest ions.

The rejuvenation of the GANIL computer control system is under way, aiming at two main goals : 1/supersedes the present control system which is technologically outmoded and driven to its ultimate capabilities, 2/matches the performances of the emerging control system with the widening scope of the services in a large variety of domains (beam setting and tuning, surveillance, diagnostics, expertise,...) within an operator friendly environment.

This paper emphasizes the main topics to be considered when designing and implementing our next generation control system. In particular, stress is laid on using : 1/acknowleged industry or international open standard hardware and software products to achieve minimization of investment over the life of the system, 2/modular structures to make easier future expansions.

II. ACCELERATOR CONTROL SYSTEM

II.1. General Layout

The first generation control system adopted a centralized architecture built with a 16bit minicomputer (MITRA 625) which ruled over other kinds of processors devoted to local or ancillary tasks : 8bit (JCAM10/INTEL 8080) and 16bit (DIVA/MC68K) microprocessor CAMAC controllers, programmable logic controllers (APS30-12 and PB400). These processors are connected to the MITRA via two bit-serial 2.5 MHz CAMAC loops which bind up 40 crates with about 800 attached modules.

This tight coupling with the computer MITRA considering architecture and non portable software makes the control system vulnerable with regard to collapse, obsolescence and ageing of that computer.

In contrast, the GANIL control system to come is based on a distributed architecture. Intelligence is therefore handed over to local processors which are responsible for dedicated field operations. The chosen topology features three functional levels which intercommunicate by means of an Ethernet local area network (LAN) :

1. The HOST level provides a general purpose and cosy environment for software programming, debugging, off-line calculations and displays, database management, and simulation. It is built around 1/ - a midrange DIGITAL EQUIPMENT computer, the VAX600-410, connected to video terminals through DECservers, 2/ - workstations VS3100-76 equipped with 19' high definition color screens for graphics oriented developments.

This level also realizes links with other facilities of the laboratory : compatible PC's and Mac serviced by a NOVELL server, CAD stations and the Physics acquisition Vax-cluster ; in addition, it provides connection to remote physics laboratories via wide area netwoks (WAN).

2. The REALTIME CONTROL level allows operators to control the accelerator by means of appropriate human interfaces. This level is based on a microVax3800, seven workstations VS3100 and X-terminals VT1300 to benefit from the enhanced graphic capabilities within the X11 standard multiwindowing environment. VAXELN controlled VME boards are used to drive the shaft knobs. Man-machine interaction will be emphasized in § II.4. The μVAX3800, which plays a key role in the real time control, is actually a dual host cluster equiped with redundant disks to achieve some kind of "failure tolerance ".

3. The EQUIPMENT level perfoms low level controls with different kinds of front end processors :

- CAMAC controllers (KSI3968 from KINETIC SYSTEMS) and VME controllers (VME300 from AEON, ...) running real-time applications under the VAXELN operating system. These controllers which are referred to as front end controllers (FEC) integrate the RTVAX300 chip, the CAMAC FEC replacing the present serial loop crate controllers.

- Programmable logic controllers (PLC) : S5-135U from SIEMENS, PB400 from TELEMECANIQUE/APRIL.

CAMAC and VME FEC, as well as the SIEMENS PLC are directly connected to the controls Ethernet LAN. Communication between VAX processors and SIEMENS PLC is achieved by DEC software packages : VSH1 which supports the application-presentation - session layers of the OSI/ISO standard and VOTS which provides services of the two next lower layers. The PB400 PLC are connected to the server node μVAX3800 by means of an asynchronous serial link that supports the master/slave JBUS communication protocol. The very first APS30-12 programmable controllers, which are devoid of LAN connexion capability, are phased out.

The general Ethernet LAN, is linked to the sensitive control Ethernet LAN via a bridge chosen for its filtering capability. Communication protocols which are currently DECnet, TCP/IP, LAT will comply with the OSI standard. Fig 2 displays the layout of the future control system.

II.2 Software considerations

Requirements

It is a matter of fact that software is taking a leading part in modern control system, as compared to hardware, with high added value caused by large human effort (many people involved over a long time to carry out).
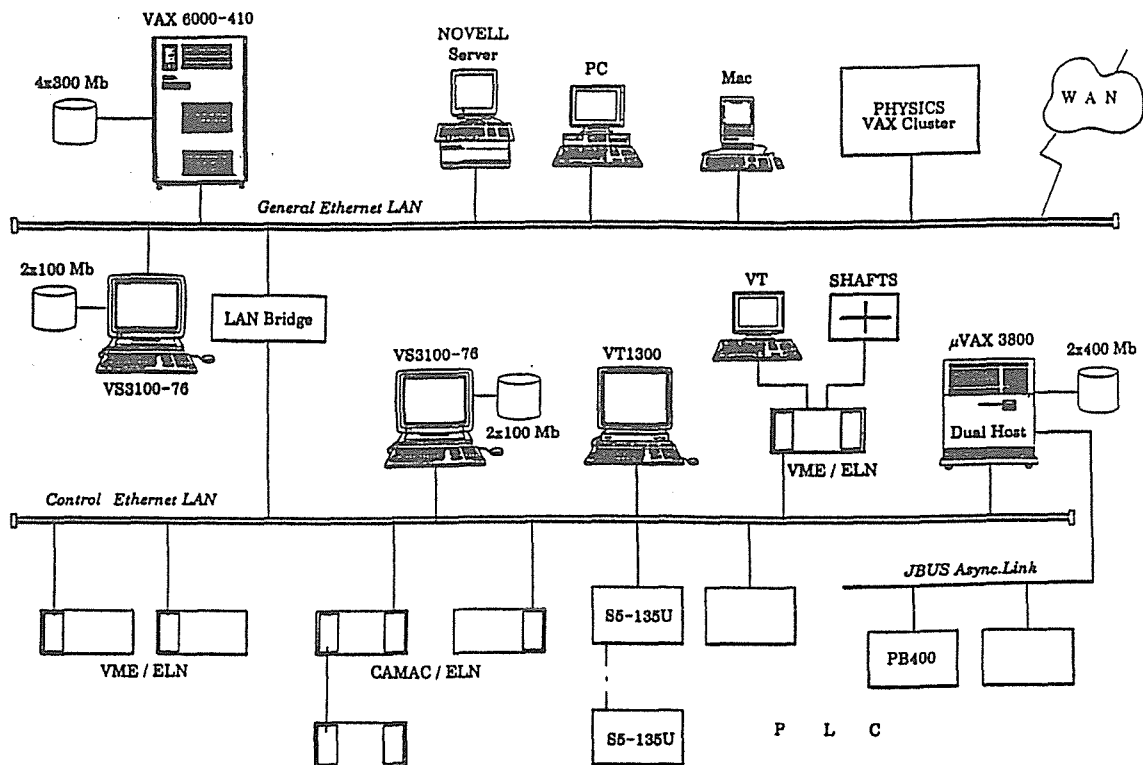

Figure 2. Schematic layout of the future Control System

Our basic requirements are :
- Dependable software to achieve productivity taking into account modern tools and methods.
- Trustworthy and easy operation which implies ergonomics.
- Well guided maintenance.
- Durability. GANIL evolutions, as a physics laboratory may be stressing, unforeseen and of large amplitude. The control system should face such a situation in the smoothest possible fashion.

Choices

To meet our requirements, standardization is the key.
Operating systems : VMS for VAX processors and VAXELN for real time controls are adopted. VAX/VMS is an industry (de facto) standard widely used in physics laboratories, VAXELN is chosen as a mature and powerful product which is tailored for real time performances. It supports a host/target connection over DECnet within a full VAX/VMS compatible environment. Therefore, the host processor VAX6000-410 performs developments and debugs while the targets (CAMAC and VME FEC) execute real-time controls.
Languages : A multilanguage system is chosen with ADA, Fortran and C. These languages comply with AFNOR, ANSI and ISO standards. ADA is our major programming language, while assembly language is relinquished.
Industry softwares : In addition to the software packages devoted to the VMS environment, LAN communication and management, some important software packages are selected :
. the INGRES family from ASK/INGRES devoted to relational database management (more in §II.3)
. the IMAGIN family from SFERCA devoted to supervision (more in §II.4)
. Xwindows and MOTIF (X11R4).

Implementations

A basic software layer has to be designed to meet the specific control requirements of the GANIL accelerator. This so called GANICIEL layer is mainly transparent to the users and is built upon the industry softwares.
It makes widely use of the client-server model and takes into consideration the distributed architecture which allows the clients and the server to run on different processors located anywhere in the LAN.
The following emphasizes the distribution of the GANICIEL functions over the control system levels :
At real time control level
The µVAX3800 assumes the function of a global server that is fanned out into dedicated functional programs to handle remote incoming requests.
Important ones are :
- Initiation of appropriate functions (e.g. communication, tasking, ...) on boot or on resumption.
- Surveillance of networking operation and FEC execution.
- Alarm handling which deals with concentration, storage and display for operator decision, in instant mode or differed mode with customized presentation.
- Data base management
. SQL translation of request stemming from FEC

. Dispatch dedicated run time database to the proper processor
. Update in partial mode for some pieces of equipment or in global mode (e.g. changing acceleration conditions)
. Archiving (e.g. beam parameters, profiles, settings, operation logs,...) processing and presenting.
Workstations assum human operator interface :
- Xterminal management for operator choice (task and hook names).
- Process server to run processes selected by the Xterminal.
- Translation algorithm to change hook names into hardware addresses.
- Local presentation of alarms.
- Beam control processes with MOTIF widgets as operator interface.
At equipment level
Controls are handed over to CAMAC/VAXELN and VME/VAXELN front end controllers. GANICIEL functions provided by these FEC are displayed on Fig 3.
Real time processes achieved at this level are :
- Communication servers to receive all the requests from workstations or other crate controllers.
- Hook to reserve and refresh the values of hooked pieces of equipment.
- Surveillance to check whether a piece of equipment is off specified limits.
- Alarms to send message to the main alarm server running on µVAX3800.
- Handlers to control piece of equipment with drivers that only control the CAMAC.
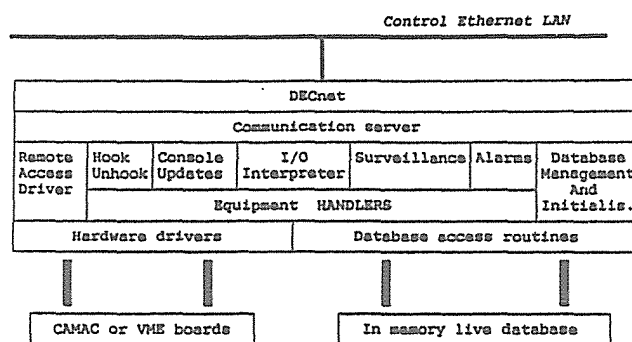
Figure 3. GANICIEL Functions on FEC

II. 3 Information management

Information management is of basic importance in a control system. Due to :
1. the various natures of information to manage :
- acceleration conditions, beam parameters (ion species, energies,...)
- realtime controls (node addresses, equipment identifications and characteristics, alarm messaging,...)
- operation logs
- reference characteristics of controlled parts (hardware installation and software)
2. the huge amount of controlled data (e.g. > 2500 pieces of equipment, beam parameters,..)

3. the distributed topology of our control system architecture (data flow vs time, up - and downloading, ...), information management has to be carefully analyzed and designed to meet performances (access, integrity, speed,...) and avoid conflicts which may be fatal.

Database management system

A careful investigation has led us to adopt a relational database management system (RDBMS) to fulfil our needs regarding NON-realtime purposes. We finally chose the commercial INGRES RDBMS as having specifications which fulfil our requirements and feature the open aspects we are looking for. Main reasons are listed as followed :

- Hardware independency, which allows straight forward integration in the VAX/VMS environment.
- Comply with SQL standard.
- Encompasss a 4th generation language in windowing environment (W4GL).
- High integrity, homogeneity of the products (QBF, ABF, ...) with good ergonomy consideration.
- ADA interfacing.
- and last but not least, a quality support.

Realtime database

Realtime database is hosted in the μVAX3800. It is fragmented into dedicated live databases reflecting the hardware configuration of the frontend CAMAC or VME/VAXELN local controllers. At these lower level locations, live databases are eventually reformated into appropriate structures for easy access by equipment handlers and for fast response, and reduced for memory saving. These live databases actually include static information and dynamic data about the operation of the controlled pieces of equipment or subsystems, under the management of a DBserver. Downline loading of live database may be a critical concern to be mastered.

II.4 Human interface

Human interface at GANIL integrates recent graphic enhanced processing units, namely VS3100-76 workstations and VT1300 X-terminals, to benefit from their color graphic high resolution capabilities and X-Window standard compliance. It is accomplished via operator consoles, supervision systems, and specific field graphic terminals.

Operator consoles

Operator consoles are installed in the main control room for centralized controls and along the accelerator for field controls and immediate interventions (e.g., the electronics backbone gallery and the experimental physics areas).

Two control levels:

1. Elementary level for individual equipment controls, the computer control system acting as a sophisticated multiplexer. This level provides utterly standardized controls for all pieces of equipment, by turning reassignable knobs,

2. Higher level for complex and global controls involving many different types of equipment, calculations and displays. This level achieves fully customized interactive controls, by running tasks.

Implementation:

Designation devices are currently trackball and mouse, to track and catch the graphic objects (task or knob image) on the screen of the X-Terminal.

Input device is keyboard to enter alphanum data, such as the name of a piece of equipment or an expected value.

Output device for control tasks is commonly the screen of the workstation for color graphics, associated with color print out devices.

Control devices are the popular reassignable knobs (shaft encoded potentiometers) which are used on the "one knob - one piece of equipment at a time control" basis. These shafts are grouped into four-unit module to achieve ergonomy and performances.

Pertinent alarm messages will be displayed on the console screens following a uniform presentation strategy with color coding to speed up interpretation. Operators can control specific actions from these consoles, such as activating the display of detailed DBMS messaging for inspection and diagnostics or acknowledging a specific or a whole class of alarms to clear the screens. Fig 4 shows an operator console unit.
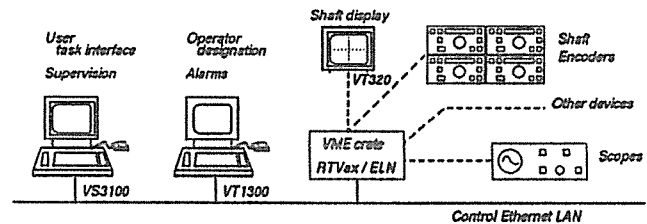
Figure 4. Operator Console Unit

Supervision

It uses worstations VS3100 to run the commercial software package IMAGIN supplied by SFERCA. This supervisory system allows control of any processor which is within the reach of the workstations, management and display of the collected data on animated synoptics over the background view. Supervised processors are currently programmable logic controllers (PLC) and the present control system. Imagin is composed of several software modules : a graphic editor, a configurer to create dynamic objects and an animator to perfom real-time display. Mailboxes provide communication threads between the animator module and the application process, as well as access synchronization to the shared data memory pool. PLC data acquisition is realized by the data server based on the SFERCA subsets PROLINK+ and its industrial database BDI (Fig 5).
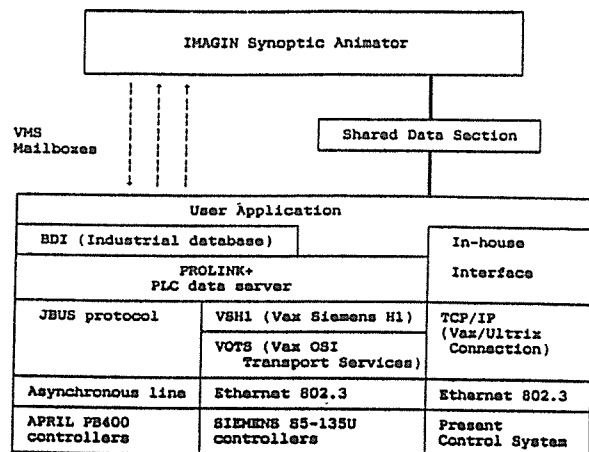
Figure 5. Supervision Structuration

Supervisory development environment is organized around wokstations linked to the VAX 6000-410 host processor, while supervision is performed on workstations related to the μVAX3800 cluster. ADA and FORTRAN application processes are interfaced to the various SFERCA modules through specific interfaces. Recent implementations encompass DECwindows environment. Examples of supervision will be presented in the following section.

Field graphic terminals

Programmable logic controllers are addressable by dedicated field terminals with attached keyboards to perform specific actions. These devices are for use by specialists for local inspection and expertise.

III. FIRST IMPLEMENTATIONS

III.1 Controls of the new ECR Ion source

A new 14.5 GHz ECR ion source, named ECR4, was installed on a 100 kV platforme to increase the beam intensity for metallic and heaviest ion species. This ECR source is controlled by a pair of SIEMENS S5-135U PLC. One PLC is on the high voltage platform and is linked by an optic fiber connection to its grounded companion. The grounded PLC is coupled to the Ethernet LAN to communicate with the supervision system running on a VS3100 workstation. An ADA application interfaces the IMAGIN software to animate synoptics. The ECR4 synoptics consists of several views to handle the RF power transmitter, to control the vacuum system and to drive the main parameters of the ECR source like the gas pressure of the UHF power, as shown on Fig 6.
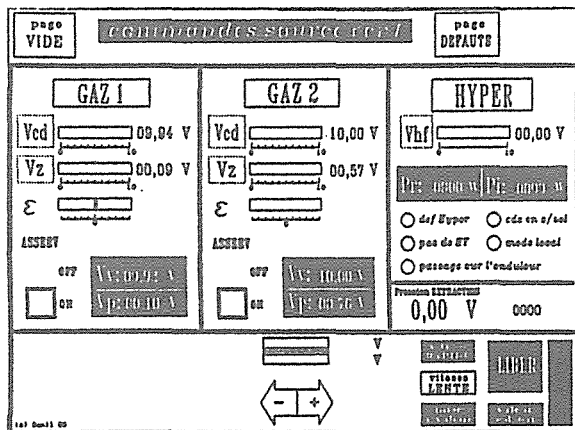
Figure 6. New ECR Ion Source Synoptic

III.2 Supervision of the internal temperature of the RF cavities

The temperature of water cooling the seven RF cavities of the accelerator are measured by the means of 34 PT100 probes connected to a SIEMENS S5-135U PLC and are supervised such as for the ECR source. In addition data related to beam characteristics, RF voltage and vacuum pressure are read from the present control system. Fig 7 shows the temperature measurements inside the northbound RF cavity of the first separated sector cyclotron (SSC1).
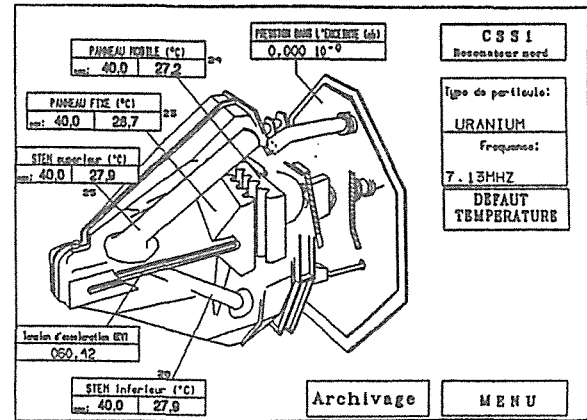
Figure 7. Temperature of a RF Cavity Synoptic

This application is the first experience of introducing relational database concepts into our controls environment. The INGRES RDBMS is used to archive measurements at operator request or in automatic mode, every 15 minutes. An off-line application using the Windows 4th Generation Language (W4GL) from INGRES displays graphs and presents result, depending on the stored data, as shown in Fig 8. It demonstrated the benefit that can be taken when developing with this language. It also led us to face the sensitive implementation of ADA/SQL interface and use of the ADA multitasking features in this case. The whole application is now about to be operational.
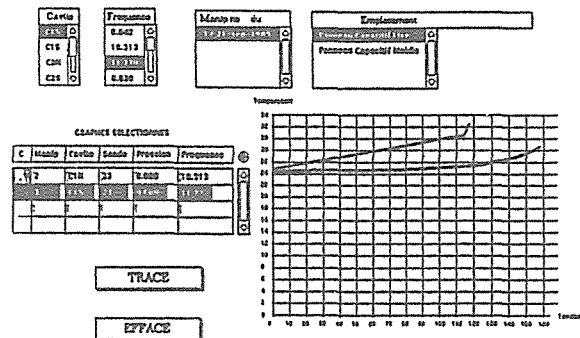
Figure 8. Off Line Display of a RF Cavity Temperature

III.3 Operation viewing and statistics

A W4GL application was written to store in an INGRES database the chronology of the GANIL operation. According to the screen displayed on Fig 8, the operator using a mouse has to point to or choose various graphic objects to specify what occurs while controlling the accelerator and the experimental areas. Every quarter of an hour, he has to indicate time and location of occuring failure, as well as beam tuning conditions and target experimental rooms. Later on, some of these manual operations will be filled up automatically.

The other part of this application to be developed will run on a Macintosh station. This application will read the INGRES database through the graphic query language (GQL) product to feed an EXCEL application for statics and report purpose. This application is planned to run by the beginning of the next year.
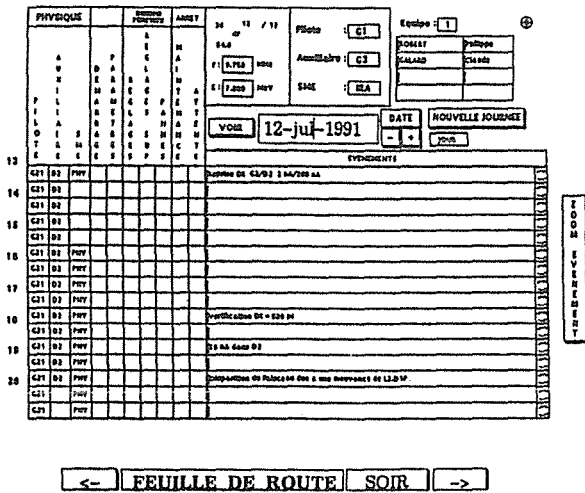
Figure 9. Operation Statistics Display

III.4 Beam profiler

It was important for us to validate by an example several real time control system basis components :

- Client server communication with workstations as clients and VME crates as servers.

- VME crates for the control of local pieces of equipment with data links to the control room workstation for display.

- The VME300 under VAXELN as crate controller.

- The DEC graphic editor VUIT to build MOTIF menus.

That was done with the beam profilers system in which the VME crate contains the 20 acquisition slots for 160 profilers.

On the VME crate, a communication server receives the requests from the workstations, according to the different allowed functions (gravity center, broken wires management, full width at half maximum). There is one ADA task for each function, so that simultaneous different requests can be satisfied. These functions find their data in an ADA task which is started by an interrupt at every end of the acquisition cycle.

After that, the computed data are sent by the LAN to the requesting workstation. An ADA program on this VMS workstation manages all the MOTIF widgets for the pop-up menus dedicated to the operator's choice, then displays and refreshes the graphic representation of eight beam profilers.
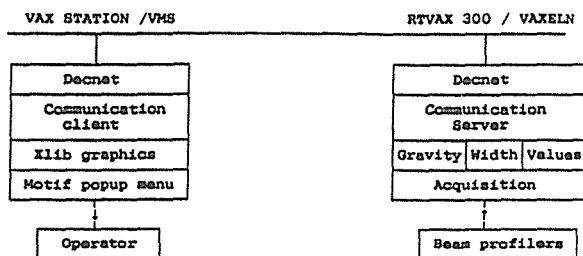
Figure 10. Beam Profiler Software Structuration

This beam profiler display application showed us that this kind of communication is fully satisfactory and can be extended to the fifteen other local control processes such as high frequency or beam phase management.
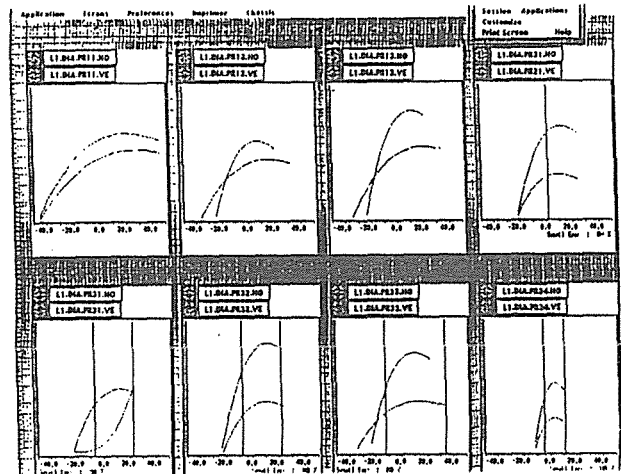
Figure 11. Beam Profiles

CONCLUSIONS

The host level is fully equipped and operating as well as the processors of the real-time control level : $\mu VAX3800$ cluster and workstations. The Ethernet LANs, to federate the three control levels are installed and provide satisfactory services.

Significant effort was deployed to analyze and build up the software architecture.

The next step will be mainly devoted to completing the GANICIEL specifications and to coding the system and user software.

The future control system is planned to run by spring 1993.

ACKNOWLEDGEMENTS

The control system described here is the result of continuous and combined effort involving others members of the Controls Group : P.de Saint Jores, E.Lemaitre, P. Lermine, C.Maugeais, F.Regnault, J.F.Roze, and our visiting guests : J. Galvez and G. Vega from the University of Bogota.

We benefit from discussions we continue to have with the people from the Electronics Group and the Operation Group of the Accelerator Operation Sector.

FURTHER READINGS

- E. Lécorché and the GANIL Operation group "A New Open GKS based Supervision System in use at GANIL" Proc. Int. Conf. on Accelerator and Large Experimental Physics Control System, Vancouver, Canada, 1989
- L. David, E. Lécorché, T.T. Luong, M. Ulrich . "The GANIL Computer Control System Renewal" Proc. Int. Conf. EPAC'90, Nice, France, 1990
- T.T. Luong "Real Time System for Local Intelligence" Europhysics News 22, 1991

REPLACEMENT OF THE ISIS CONTROL SYSTEM

R.P. Mannix, C.J. Barton, D.M. Brownless, J.C. Kerr

Rutherford Appleton Laboratory, DIDCOT, Oxon., OX11 0QX, U.K

Abstract

In operation since 1985, ISIS is the world's most powerful pulsed spallation neutron source. The decision has been taken to replace the existing ISIS control system, which has been in use for over ten years. The problems of such a project, given the legacy of processor specific hardware and software are discussed, along with the problems associated with incorporating existing interface hardware into any new system. Present progress using commercial workstation based control software is presented with, an assessment of the benefits and pitfalls of such an approach.

I. INTRODUCTION

ISIS is based on an 800MeV proton synchrotron, running at 50Hz, providing an average beam current of 120μA onto a Uranium target. The injector, synchrotron, extracted proton beam line and target station have been under control of the present system both during commissioning (1980-85) and operation. Work is in progress to attain the design current of 200μA within two years.

The control system (old CERN-SPS pattern), is based on 5 GEC computers, assembly language Data Modules, a general purpose multiplex system for equipment interfacing (MPX) , CAMAC based operator interfaces (see Fig. 1) and an interpretive control language. There are approximately 15000 lines of data module source code. The equipment interface consists of 700 modules in 70 crates. There are several hundred control programs in use.

The present control computers are very modest in power and are very limited in storage capacity. The operating system allows us to control hardware from a number of concurrent interpreter processes on each processor. High priority processes communicating with hardware completely lock out all others. Communication with other systems is nonexistent, peripherals such as floppy disks are obsolete and unsupported, backups require shutting down the control system and maintenance is expensive (24hr cover is essential). The various branches of the interface hardware are tied explicitly to particular processors— reconfiguring after a processor failure is impossible— a serious disk drive fault can, and has, shut down the accelerator for a few hours. An aver-

age ISIS experiment lasts two days, within which several data taking runs, all of which are essential, must be performed. These sorts of breakdowns, although infrequent, are highly undesirable.

The new system is required to take over all the current functions of the existing control (at least as well) and have the capability for much increased data storage, greater reliability, easy reconfiguration and extension and almost transparent communications with other computer systems.

The available effort is about 25% of that when the original system was created, when both decreases in staff and the load of supporting the existing system are taken into account.

II. PROJECT PLAN

The system currently being developed is based on the Vista Controls[1] software suite, and the Hytec Electronics[2] Ethernet CAMAC Crate Controller (ECC). The new software provides a fully distributed database driven control system with a graphical interface over a number of DECnet nodes (currently VMS only although work is in hand on a POSIX-compliant version). Each control or monitoring object is referred to as a *channel* and, by using the *channel name*, control screens can be generated with an interactive draw package. Our current development system is based on two DEC VAXstation 3100 colour workstations and two DEC VT1300 X-terminals, although it is not clear that the choice of processors is optimal. Figure 2 shows a general arrangement of the proposed new system, based on Ethernet (an alternative transmission medium would be FDDI).

The use of channel names, database and handlers (equipment routines) maps very well on to our existing system based on Data Modules. The graphical interface provided with the new software should enable the functions of 75% of the control software to be replaced without recourse to writing code.

The system-wide nature of the databases and the networked nature of the CAMAC driver crates makes access to all equipment possible from any processor— something which was not previously possible.

Terminals with any level of access to the control system can easily be added anywhere on site (if desirable!). A con-

trol system for a new beam line can be provided by buying a workstation, another CAMAC controller and local interfacing hardware. The incorporation of this into the existing system is automatic (subject to the licensing agreement and an upper limit on the number of databases).

The current operator interfaces (touch sensitive screens , tracker balls, colour displays, knobs etc.) which are all obsolete or nearly so, will be replaced by high quality mouse/tracker ball and keyboard driven workstation type displays.

No new system will look or behave as the old one did. Users are familiar with the old system and will be reluctant to change. Any shortcomings in the new system will be picked on and amplified, shortcomings in the old system having been assimilated years ago. Use of the new system from a user-written program is more complex and less flexible– we are currently developing a simple interface to improve this..

There are three major parts to the management of change:

1. Transport of existing software.

2. Connecting to the new control system.

3. Training users

Most of #1 can be done off-line, that is to say without interfering with the operation of the accelerator, and this is currently in progress.

#2 is straightforward because all equipment below the dotted line on figure 2 is to remain the same. The CAMAC crates which drive our MPX branches merely have to have their crate controllers removed and replaced with the ECC controller modules. Changing the existing main control desk is highly problematic. Workstations will have to be installed side by side with the old control system on the first live runs, so that the old system can be reverted to if necessary. It must be stressed that ISIS runs as if it were a commercial enterprise, and our "customers" will not allow scheduled run time to be removed or interfered with for development purposes.

Training of staff, given our limited resources, is difficult. The operation of ISIS, although still to be further developed, could be said to be stable, so a straightforward functional replacement would be an acceptable first stage.

There is a long shutdown period of 3 months every year when it is expected that changes will take place.

The original time scale was for a two year project, culminating in a long shutdown. This is not feasible given the current manpower levels, so April 1993 is now the target date for a live run of the new system..

III. PROGRESS

Three databases have been written, those for the injector magnets, the injector timing system and the injector general purpose status modules– a total of 1200 channels so far. Handlers for the programmable timing modules, the most complicated of our magnet power supplies, and the status reading hardware have been written. We are preparing a specification for extra functions to be added to the Ethernet CAMAC crate controller, to reduce the overheads on equipment access. Control screens for the timer modules and magnet power supplies have been prepared and a suite of hardware test programs written for test access of interface modules via the ECC controller (external to the database).

Fig. 3 shows a workstation in use in the new control system. The menu windows running around the bottom right hand corner are replacements for our existing touch screens, where further menus and/or control screens are called up. The main part of the display is occupied by a control screen for the quadrupoles in the Low Energy Drift Space of the ISIS LINAC, showing control sliders and a monitoring strip chart. Fig 4 shows the operation of the control screen for the ISIS LINAC programmable timing system. No high level programming was required to generate these windows.

A significant amount of time has been spent deciding how to modify the standard usage of the Vista software to suit our needs and in moving to new versions. Now this has been done, production of software should speed up.

The choice of processor platforms on which to mount the system is not obvious. It is driven by the need to maximise perceived response time. We are not sure at this stage whether workstations are more or less appropriate than a powerful multi-user VAX running several X-terminals. Increases in workstation power may overtake this problem.

There is a lack of flexibility in hardware calls routed through the database, stemming from the inability to parameterize calls. One might wish to be able to retrieve the value of a single status bit from a 16-bit status port. In the present system, this is done by specifying the module, port and bit number– a 0 or 1 is returned. In the new system, to be able to randomly access any single bit in this manner would require database channels for every single bit, which would be very wasteful. We have opted to assign an integer channel to each 16-bit port, individual bit channels being set up on demand. On the other hand, this rigidness leads to a strong "typing" of database channels, minimising errors.

Complex control screens may be devised without any programming effort, however the hierarchical control over which screens may be displayed at any one time seems lack-

ing to us and we are devising our own software to handle this.

Quick turn round support is available from the vendors by e-mail, phone or Fax.

On any physical Ethernet segment up to 256 CAMAC crates (each driving up to three MPX branches with up to 16 MPX crates per branch) can be simultaneously available from any processor with the ability to restrict access to individual modules to particular processors if required. The ECC modules have been reliable in operation with excellent support from the manufacturers.

The major difficulty with the ECC controller is an overhead (on single operations) of 10mS per transfer. The 10mS is a fixed feature of the VAX-Ethernet configuration and is not affected by transfer size. In a data acquisition environment the problem starts to disappear as the blocks of data transferred get larger. For a control system such as ours, where accesses to the hardware are much more random and multi-sourced, this is a large problem for which there is no clear solution as yet. The advantages of the ECC solution are the total flexibility and the overcoming of many of the limitations of CAMAC (to the extent of giving it a new lease of life).

IV. CONCLUSIONS

The choice of commercially available software must be correct for those establishments where lack of effort and staff turnover are problems. Good local support and constant updating of the product are also essential, as is the ability to feed requirements into the supplier's development plan. Even so the effort involved in changing to a new system is always under-estimated, both by the customer and by the supplier.

No commercial product will allow the easy assimilation of an existing control system. Whatever practises and methods prevail in an operating machine are the "right" ones by virtue of the fact that they are in use and familiar to those who use them. Any new system must be modified to fit what exists– not the best way to proceed. It is also clear that the only timing information of any interest to the user are the times taken to (1) present the control window required on the screen and (2) to operate a piece of hardware and see the effect on the screen, what happens underneath being irrelevant.

We are happy that the new control system will meet all our expectations with regard to extensibility, reconfiguration and communication and will perform as well as or better than the existing system. It seems to be an unwritten law of control systems that, as the power of the processors increases, the complexity of the software rises until the perceived response time drops to the minimum acceptable. We would hope that the extra complexity in the new system will be achieved without a drop in perceived response time.

The suitability of Ethernet (or any general purpose network system) as the main i/o channel is unclear. On the one hand emerging computer systems are frequently only supplied with an Ethernet port and moving away from this becomes expensive. It is truly distributed and configuring the system and expanding it become trivial. On the other hand, the generality of the system means that it is slow. It is clearly unsuitable for a fast data acquisition system but may be well suited to a supervisory control system such as ours. It allows the integration of PC's, CAMAC, VME, STEbus, and other systems in a controlled manner. Together with distributed software such as that described, it provides a system which is easily reconfigurable should a processor fail. In the ECC CAMAC controller itself there is also a large amount of untapped power although an easier way of accessing this would be an advantage.

REFERENCES AND ACKNOWLEDGEMENTS

1.   Vista Controls Inc.
         127 Eastgate Drive
         #30800
         Los Alamos, NM 87544
         USA
         Phone: (+1) 505 662 2484
         Fax: (+1) 505 662 3956

2.   Hytec Electronics Ltd.
         Ladbroke Close
         Woodley
         Berks   RG21 3YT
         U.K.
         Phone: (+44) 734 697973
         Fax: (+44) 734 441068

(These companies are reciprocal agents)

GEC Computers are now GPT Datasystems Ltd.

DEC, DECnet, VAXstation 3100, VT1300, VMS are registered trademarks of the Digital Equipment Corporation.

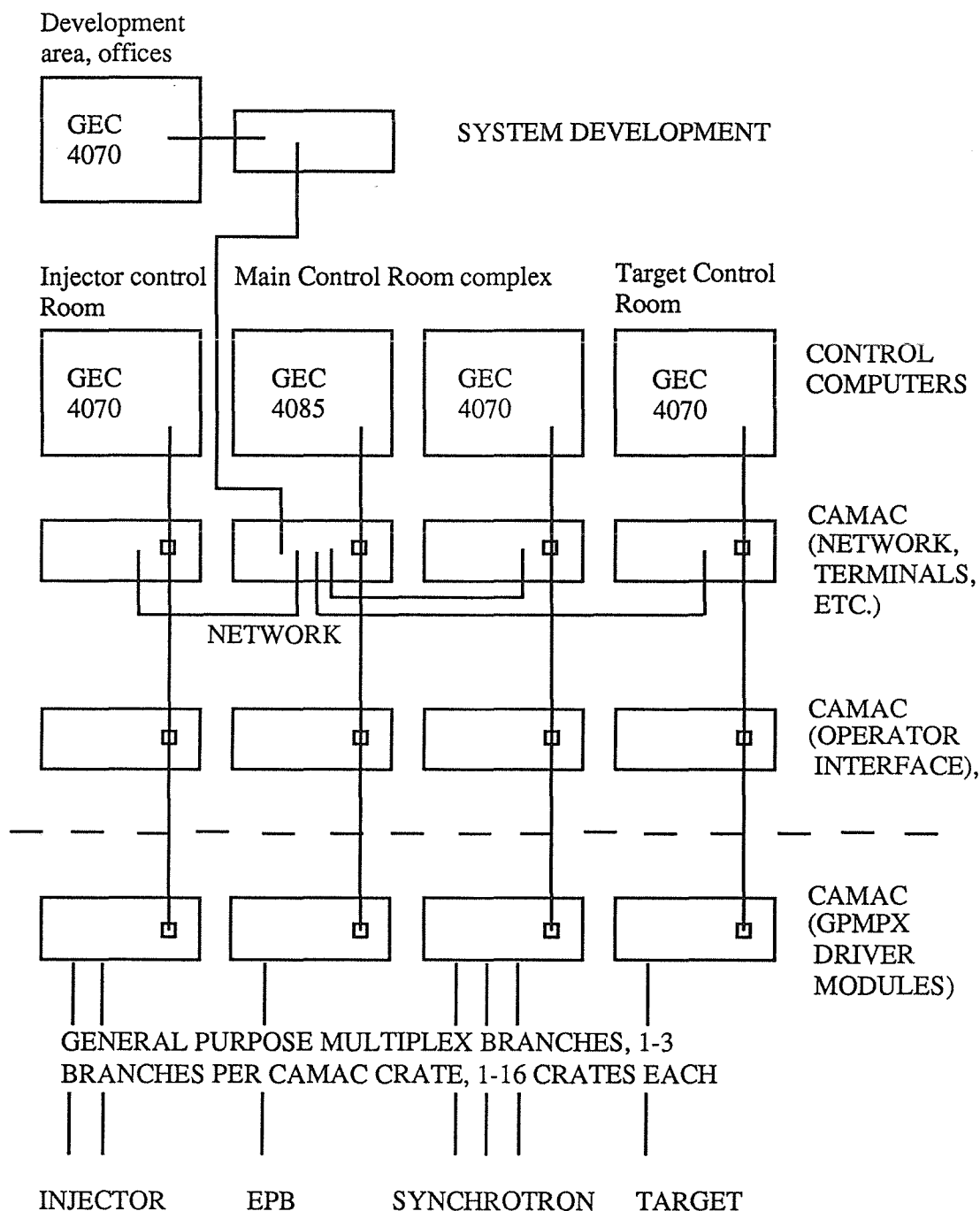Ethernet is a registered trade mark of the Xerox Corporation.

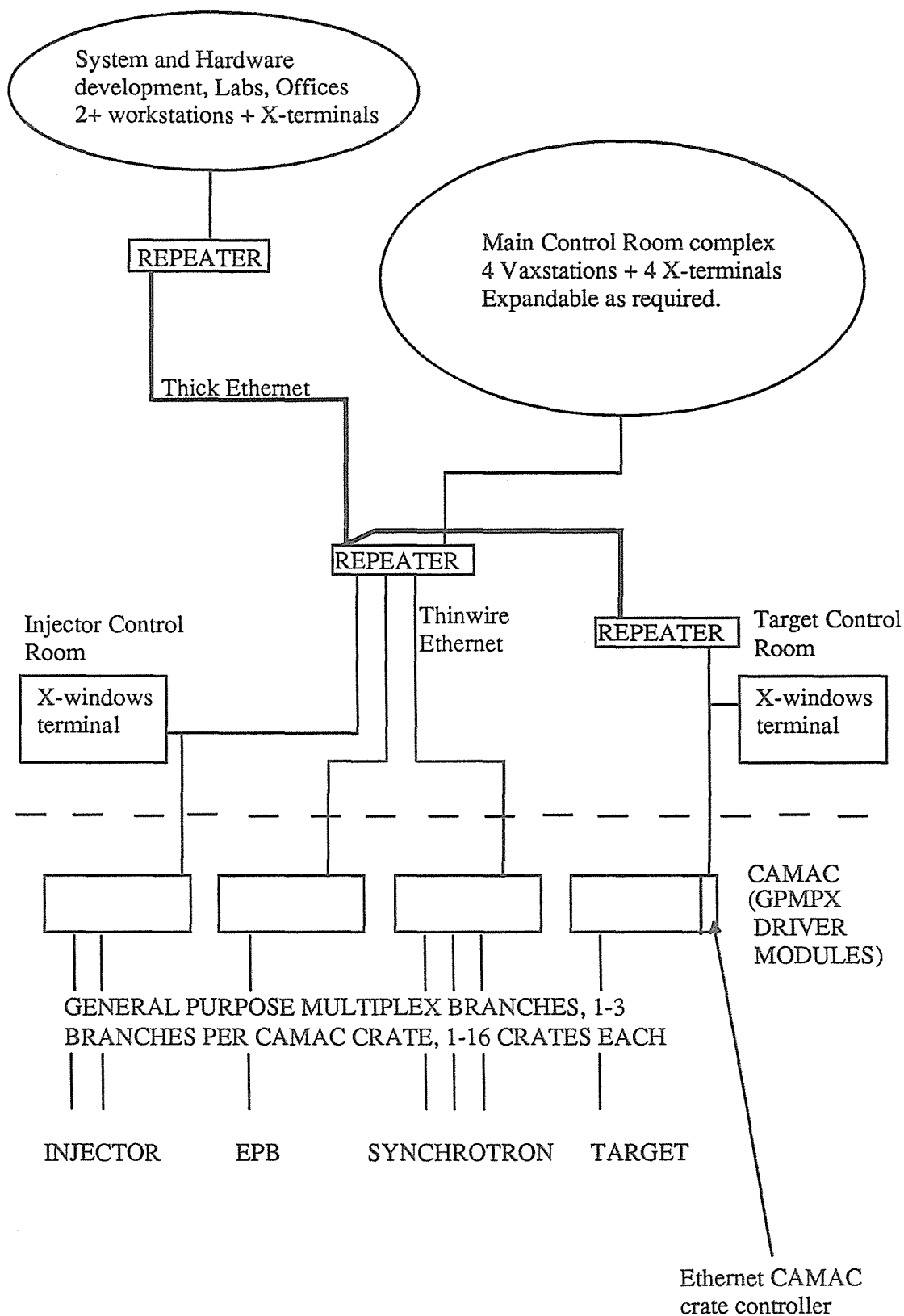FIGURE 1. General arrangement of present system
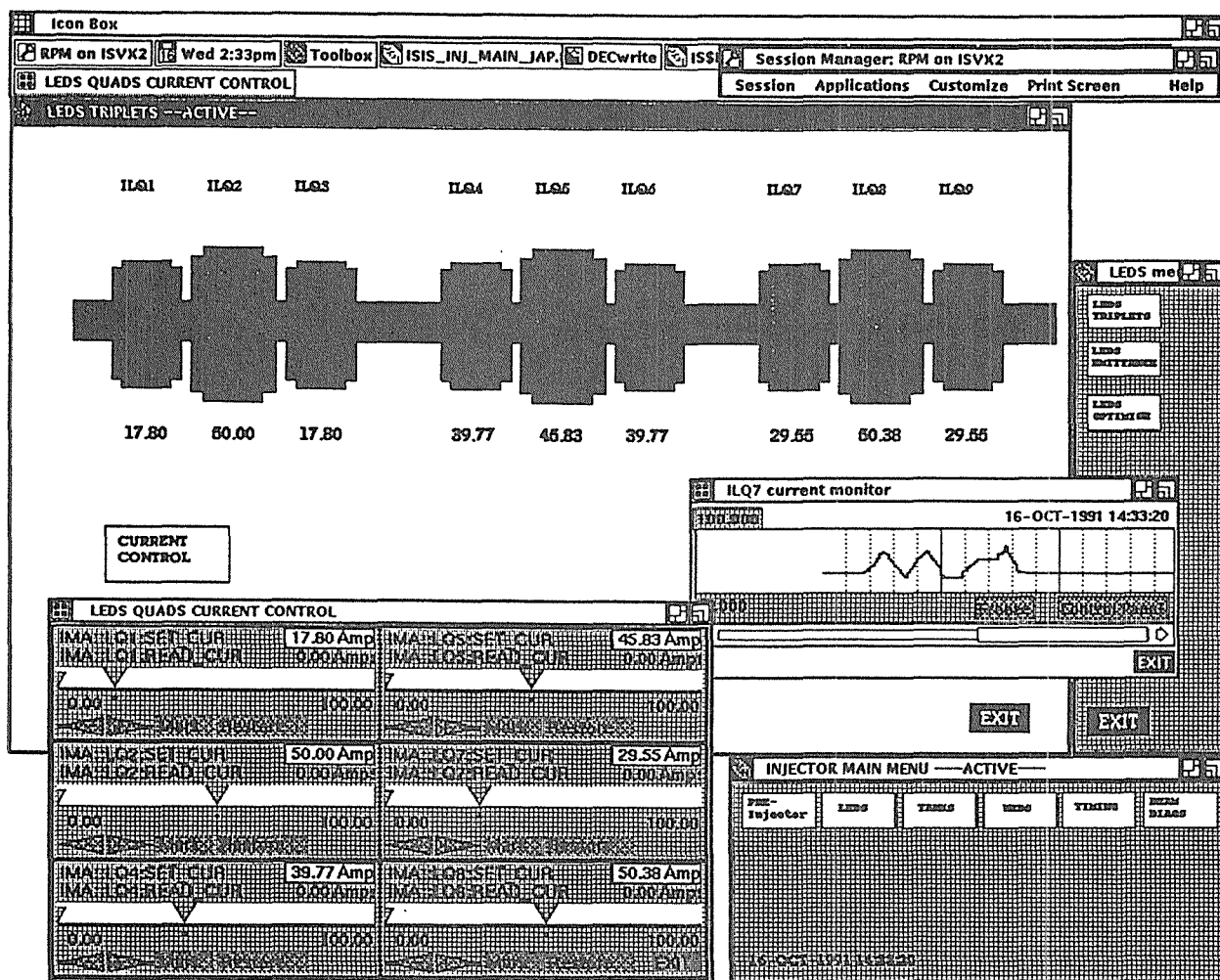
FIGURE 2. General arrangement of proposed system

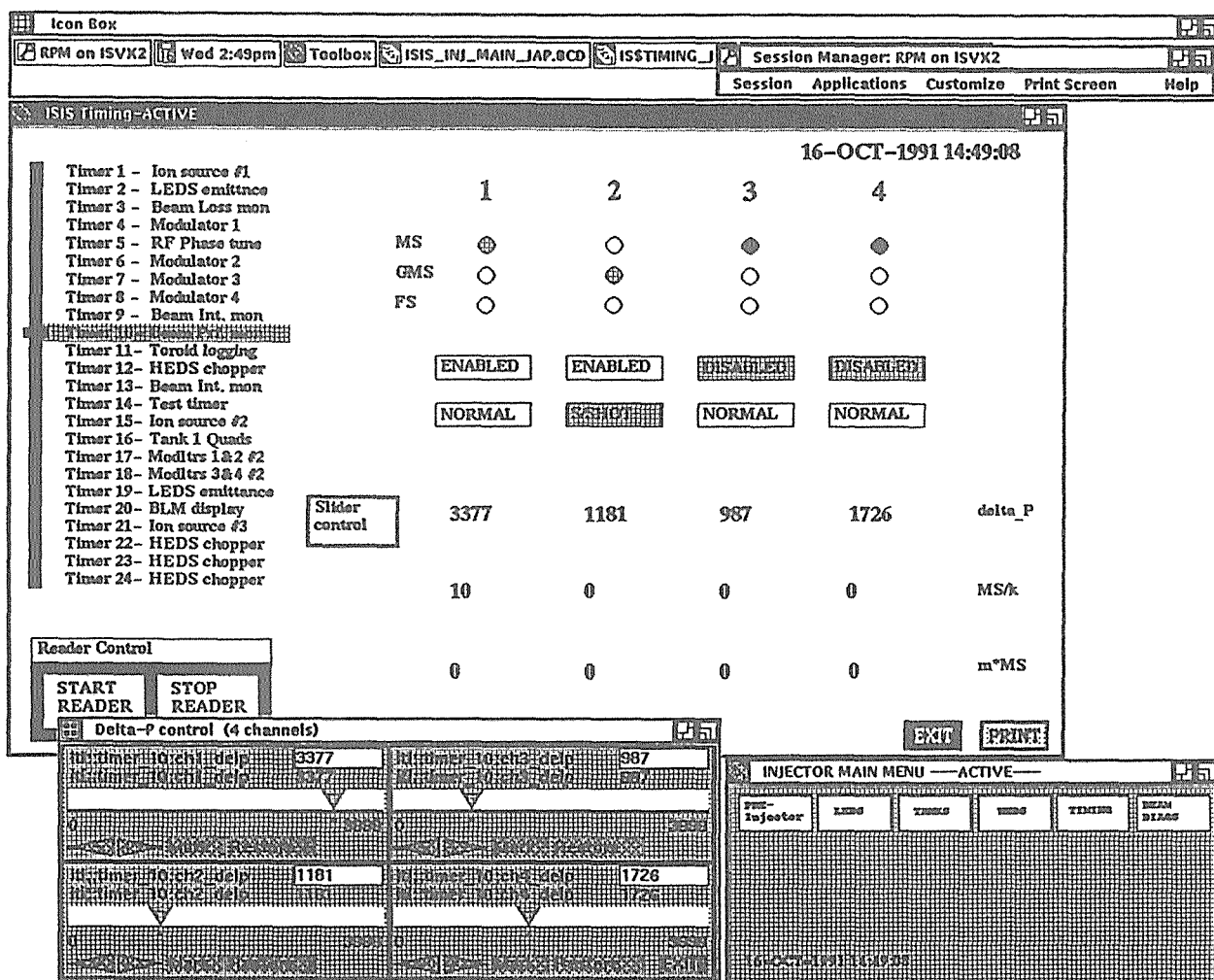Figure 3: Example control screen– ISIS LINAC Low Energy Drift Space triplets.

Figure 4: Example control screen– ISIS LINAC timers.

Upgrading the Control System for the Accelerators at The Svedberg Laboratory

K.J. Gajewski, L. Thuresson, O. Johansson
The Svedberg Laboratory
Uppsala University
P.O. Box 533, 751 21 Uppsala, Sweden

Abstract

Two accelerators at The Svedberg Laboratory in Uppsala, the Gustaf Werner cyclotron and the CELSIUS ring, will get a new control system. At present both the cyclotron and the ring have their own control systems based on S99 and PDP-11 minicomputers respectively. There are also a number of subsystems which are controlled separately from the stand-alone PC based consoles (ECR ion source, electron cooler, vacuum system). The goal of the rejuvenation is to integrate all existing control systems and provide the new system with an uniform operators interface based on workstations. The obsolete S99 microcomputers will be substituted with a VME system and all subsystems will be connected to the Ethernet. The upgrade strategy enabling the transformation of the system without any long shut-down period is discussed. Hardware and software planned for the upgrade is presented together with a discussion of expected problems.

1. INTRODUCTION

The control systems for Gustav Werner cyclotron and the CELSIUS ring have been designed at different times. The cyclotron [1] after a major rebuilding started its operation in 1986 The main cyclotron control system, based on S99 microcomputers, covers the cyclotron and all beam lines. The radiation protection system for the whole laboratory area is also based on S99 microcomputers with an 286/PC as a console. Another 386/PC connected to a Programmable Logic Controller (PLC) controls the external ECR ion source. The CELSIUS control systems is based on the LEAR control system from CERN slightly adapted to our needs. The system is based on a PDP-11/73 minicomputer connected via CAMAC hardware to the accelerator equipment. This system enables to control all function generators and all static parameters but these for the electron cooler. The electron cooler is controlled independently from an IBM PC. Some other subsystems, both on the cyclotron and the storage ring side, are also controlled from the 286/386/PCs. From the very beginning the work on these control systems was completely independent - there were separate teams for each machine (the cyclotron and the ring) and there was no cooperation between them. It all resulted in two distinct control systems with different philosophy and incompatible hardware solutions. After achieving the production stage of operation by the cyclotron and approaching the same stage by the CELSIUS ring a need for a common control system became obvious. Some organizational changes (creating one control group for cyclotron and the ring) should help to achieve this goal. The cyclotron and the ring is each controlled from its own control room with very limited possibilities to control or even only access the information from the other machine. This is highly unsatisfactory and the new control system will enable to control both machines from either of the control rooms. Some other features like beam sharing will be added. In order to improve the situation we plan to connect all substems together via an Ethernet and add a new user interface in the form of UNIX workstations. Some of the old S99 microcomputers will be substituted by a VME system running VxWorks. Such approach will enable us to achieve our objectives with minimal changes in the front end level and without any long shut down periods of the accelerators.

2. EXISTING CONTROL SYSTEM ARCHITECTURE

A. Cyclotron

The cyclotron control system is based on Texas Instrument (TI) microcomputers. At the top level there are three TI S99 microcomputers. The control system data base resides in one of the S99 systems, while the other two are mainly used as operator interface to the system. At the hardware level most of the equipment is connected to the control system by an in-house-built microcontroller called General Interface (GI). A single euro-board includes a TMS9995 microprocessor, analog and digital I/O channels, a terminal port for local control and a custom-designed serial communication bus. Up to 31 devices can be connected to this bus. Special interface boards (also in-house-built) are used to connect the bus to the S99 systems. There are about 150 GI's in the system. They are mainly used in the beam transport- and RF-system. Apart from the GI-controllers two industrial PLC units (Modicon 484 and 684) are used in the vacuum- and RF-system, fig. 1.
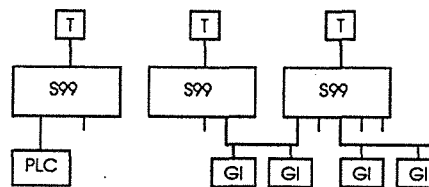
Figure 1. The main control system of the cyclotron.
T - terminal
PLC - Programmable Logic Controller
GI - General Interface (front-end controller)

The S99 systems runs TX990, a multitasking operating system from TI. Today both hardware and software support for S99 is very limited. Most of the control system software is written in assembly language. All this makes maintenance difficult and time consuming.

The radiation protection system also uses two S99 microcomputers. One S99 system handles the interlock and access control to the various experimental areas. The other monitors the radiation level by means of some 50 radiation detectors distributed in the laboratory. A 286/PC is used as

operator interface. The radiation levels are continually logged to the PC disc. No distributed I/O (GI's) are used in this system, thus some 600 digital I/O signals and 50 counters are connected directly to the S99-systems, fig. 2.
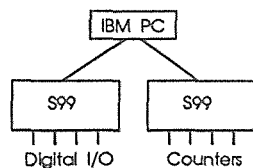
Figure 2. The radiation protection subsystem.

In 1989 an external ECR ion source [2] was added to the cyclotron. A Siemens S5 -115U [3] PLC is used as front-end. For operator communication a 386/PC is connected to the PLC system by means of a Siemens CP524 [3] communication processor, using an RS 232 channel to the PC. Most of the hardware is connected to the PLC but a few devices are connected directly to the PC using RS232, fig. 3.
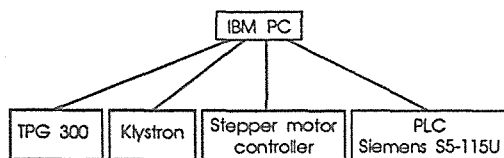
Figure 3. The ECR control subsystem.

MS-DOS was chosen as operating system for the PC, mainly because of the large amount of inexpensive software available for this environment and because the staff was already familiar with this operating system. Another alternative would have been to use a real-time multitasking operating system. This would probably have resulted in a more efficient system but at a higher cost. The present solution was considered adequate since the ion source control handles a relatively small number of parameters (<100).

The control program is written in Microsoft C. To simplify programming, a library of process control routines was written. This library allows creation of multiple threads of execution within a program. Threads can communicate through shared variables or message queues implemented in the library. The threads are scheduled using a nonpreemptive round robin algorithm. This eliminates the need for mutual exclusion mechanisms when accessing shared data. Scheduling only occurs when a thread makes a blocking call to the library (for example reading from an empty message queue). Driver routines for the RS 232 channels are also included in the library. This enables the driver to block a thread waiting for an I/O operation to complete.

B. Storage ring

The control system of the CELSIUS ring covers all function generators (25) and static power supplies (control and/or acquisition) in the ring (about 100), a vacuum system of the ring [4] (about 350 digital inputs/outputs plus 16 analog inputs) and the timing system. The whole system

consists of three parts: main control system (fig. 4), electron cooler control (fig. 5) and vacuum control.
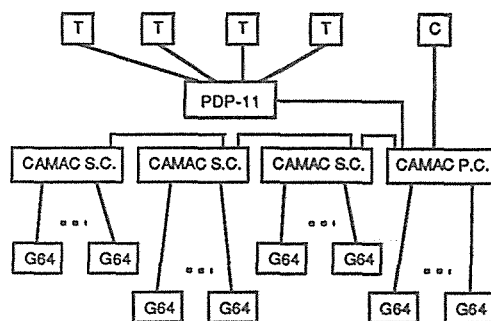
Figure 4. Main control system of the ring.
T - terminal, C - console,
CAMAC PC, CAMAC S.C - parallel and serial crate
G64 - front-end controllers in G64 standard
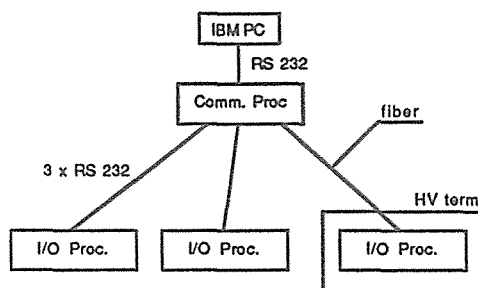
Figure 5. Electron cooler control.

The front-end level in both the main control system and the cooler control system is based on the G64 standard. The vacuum control is executed by a programmable logic controller (PLC). The PC in this case, connected to the PLC via RS 232 serial line is used only as an operator console. The PDP-11 is running under RSX11-M V4.2 operating system and DECnet. The kernel of the control system and many applications are written in MACRO-11 assembler. Other application programs are written in FORTRAN and Pascal. The PCs are running under MS DOS and the control programs are coded in Turbo Pascal. There are several data bases and data base management programs. In the main control system there is a database for static parameters, for dynamic parameters (vector tables) and for the timing system. The cooler control system has its own data base. All these data bases have different structure and contains various fields needed for the control tasks. The operator interface is also different for each system. The main control system have a console with a touch screen, numerical keypad, knobs and a semigraphical color display. The touch screen menus have a tree structure with 16 buttons on one page. The buttons can be used to pick up the parameter for control, to start an application program or to choose another page. A number of application programs have a text oriented user interface and are run directly from the terminal. The user interface of the electron cooler is based entirely on the PC working in the text mode. The operator can display up to 20 parameters on the screen. The absolute and incremental control is possible from the keyboard. Commands to the control programs are invoked by the function keys. The vacuum control system has a graphical user interface. The status of the vacuum system is

presented on the screen with dynamic symbols (changing color or/and shape) on top of the static picture. The operator communicates with the system with use of dialogue boxes, buttons, text input fields etc.

3. UPGRADED CONTROL SYSTEM

A. Hardware

The hardware configuration of the upgraded control system is presented in figure 6. All control computers will be connected to a segment of the Ethernet which is isolated by a LAN bridge from the university network. A repeater must be used because of segment length limit. The front-end level of the control system will not be changed with an exception of a limited number of low level controllers which should enable switching between several control values synchronously with the ring acceleration cycle. This modification is going to support beam sharing between the storage ring and other cyclotron users as well as switching the settings of the electron cooler between injection energy and flat top energy. The switching will be triggered by external pulses from the timing system.

The changes in the hardware of the next level - local process computers, will be more significant. The S99 systems in the cyclotron control will be replaced by a VME system. A Force CPU-30 [5] will be used as local controller. Due to the large installation base of the GI-controllers they will be kept in the upgraded system. A new version of the GI-communication bus interface, adapted to the VME-bus, have to be manufactured in-house. RS 232 and Ethernet interface are included on the CPU-board. All PCs will be supported with the Ehternet controllers. An upgrade of some PCs based on the 80286 processor to 80386 with at least 2 MB RAM is necessary.

The radiation protection system will keep the S99 system as hardware interface since extensive rewiring will otherwise be required.
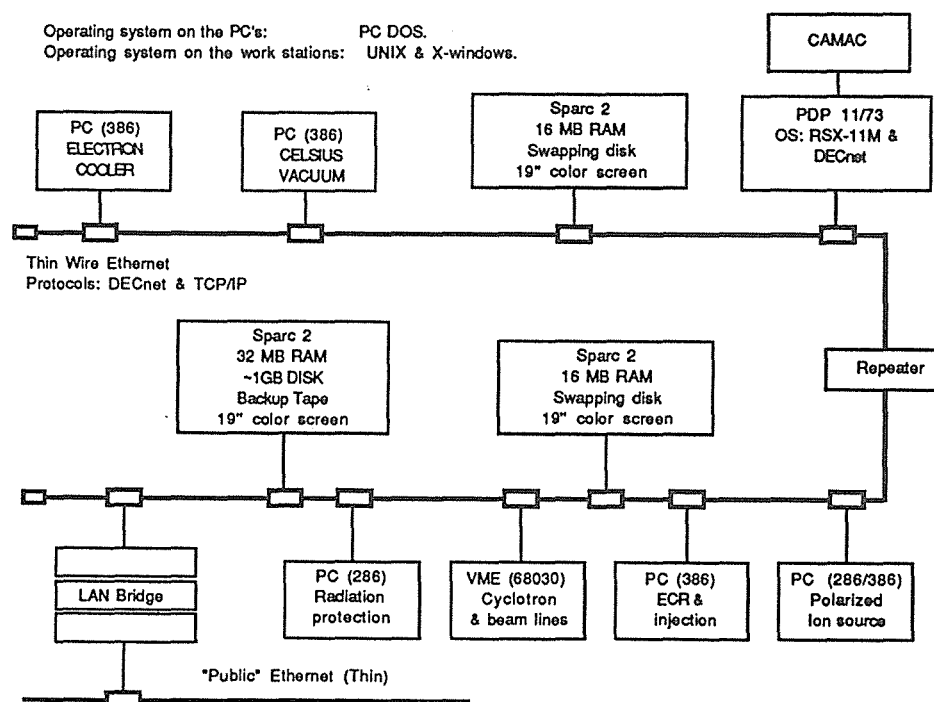
Figure 6. The layout of the upgraded control system.

An additional external polarized ion source [6] will be installed in the beginning of 1992. The ion source is bought as a "turn key" system including control system from Balzers/Pfeiffer. It uses a Siemens S5-135U [7] PLC as front-end and a 286/PC for operator communication.

The top level (user interface) will be based on Sun Sparc 2 workstations equipped with 19 inch color screens. We assume that the mouse and the keyboard will support satisfactory operator interface.

B. Software

The main effort in upgrading the control system lies in the software. Our strategy is to preserve as much of existing investment in the control software as possible. However, a considerable part of the programs will have to be rewritten or moved to another platform. Most of the programs on the workstation must be written from scratch. We have chosen TCP/IP as a communication protocol between the workstations, PCs and VME system. For communication with the PDP-11 we will use DECnet.

The VME system will run the VxWorks[1] real time operating system. The data base will be similar to the ECR ion source data base structure, with the addition of support for multiple control values to a parameter to enable "beam sharing". VxWorks is a "UNIX like" operating system. This means that program modules can be written and roughly debugged on the UNIX workstation, then recompiled and loaded in the target system.

To minimize the use of the poor software development environment in the S99 system, the radiation protection control program will be moved from S99 to PC level. The S99 will only be used for transfering the system I/O image between the PC and the actual hardware. The I/O image mapping server in the S99 will be unaware of the PC application program structure , thus enabling changes in the application without modifying the S99 software.

The programs running on the PC will be modified in such a way that they will be able to get requests from the network and send replies to the callers. The socket interface will be used. The network software we use (PC/TCP for DOS) can support up to about 20 open TCP connections. Most of the connections will be used by application programs and one is reserved for alarm and warning channel to the alarm server on the workstation. The existing programs should still perform all functions as without the network interface. This should allow to gradually move control from the PC to the workstations when the relevant programs on the workstation will be ready and thoroughly tested. We plan to use the same communication procedures and algorithms on all PCs. The programs written in Turbo Pascal will be rewritten to MS Pascal to enable easy linking to the PC/TCP network library.

The polarized ion source uses Wizcon [8], a commercial control system software package. This product have built in support for remote control nodes, using the Netbios communication protocol. This will present a problem since the control system network uses only TCP/IP and DECnet.

The data-base in the workstation will contain only information needed by the application program to access the parameter. All low level internal characteristics will be hidden to the application. There will be no global "life" data base with all parameters continually updated on the workstation. The application programs will get this information from the lower levels on request. The alarm server will listen to the information coming from the different nodes and inform the user of possible problems. This task should also check if all nodes are alive.

We will use SL-GMS [9] as a tool for creating the user interface. The GMS is a software tool kit for building dynamic graphical 2D man-machine interfaces. It was chosen mainly because its flexibility and good compatibility with the X environment.

4. SUMMARY

The work on upgrading the control system has begun about 6 month ago. During this time the requirements for the new hardware and software were specified. We stressed the necessity of using commercial, wide spread and well supported hardware and software. Most of the hardware is already in place or ordered. All computers have been connected to the Ethernet and some tests with the network communication programs have been performed. At the moment we work on the structure of the high level data base and communication protocols. The first program which we plan to write is a synoptic program which will show the status of any part of the accelerators and enable to control selected parameters. Next the application programs from the PDP-11 running on the console will be moved to the workstation. At the same time the work on moving the main cyclotron control to the VME standard will start. The programming and testing will be done on a development system not connected to the cyclotron. After positive tests the system will be installed on the target VME system. The alarm server, the archive program must also be written as soon as possible. As a last step we consider to move most of the application programs running on the PDP to the workstation. Most of them are terminal oriented, so in the transition period we can run them in the terminal emulation window.

The main problem we have to face in upgrading the control system is the incompatibility of existing data bases. This we plan to overcome by creating the new, high level data base for the programs running on the workstation, common for all parameters independently of their physical location. Another problem we meet is the limited pool size in the RSX-11 operating system. This limits the number of tasks which can run on the PDP and special attention must be paid to programming network servers, allowing only one instance of the task. We expect some problems with the RAM size limit imposed by the MS DOS, especially if we would like to keep all the functionality of the existing programs running on the PCs.

5. REFERENCES

[1] S. Holm, Proc. Int. Conf. on Cyclotrons and their Applications, West Berlin, May 8-12, 1989

[2] C. Ekström et al. "Status of the ECR ion source at the Uppsala cyclotron", 4th Int. Conf. on Ion Sources, Bensheim, Germany, 1991. (To be published in Rev. Sci. Instr.)

[3] S5-115U Programmable Controllers, Siemens Catalogue ST 52.3.

[4] K. Gajewski, "The Vacuum Interlock System for the CELSIUS Ring", Nucl. Instr. and Meth. in Phys. Res. A293, 1990, pp. 183-186

[5] CPU-30, Force computer hardware data book 1991.

[6] C. Ekström et al. "External Ion Sources at the Uppsala cyclotron", Rev. Sci. Instr. 61 (1990), pp.628-630

[7] S5-135U Programmable Controllers, Siemens Catalogue ST 54.1.

[8] Wizcon , PC Soft International, Wizcon User Manual 2.1

[9] D.G. Shelef, "Developing Graphics in Realtime", SunTech Journal, Vol. 3 No 3, pp. 64-74, July/August 1990

---

[1] VxWorks is a trademark of Wind River Systems, Inc.

Upgrading the BEPC Control System

Yang Li-Ping, Wang Li-Zheng, Liu Shi-Yao

Institute of High Energy Physics, Academia Sinica

P.O.Box 918, Beijing

Abstract

The BEPC control system has been put into operation and operated normally since the end of 1987. Three years's experience shows this system can satisfy basically the operation requirements, also exhibits some disadvantages araised from the original centralized system architecture based on the VAX-VCC-CAMAC, such as slow response, bottle neck of VCC, less CPU power for control etc.. This paper describes the method and procedure for upgrading the BEPC control system which will be based on DECnet and DEC-WS, and thus intend to upgrade the control system architecture from the centralized to the distributed and improve the integral system performance.

1. The system status and its milestone

The project of BEPC control system was determined to adopt basically from the SPEAR control system in january of 1985. The prototype control system was constructed during half year begun from september of 1985 at SLAC. In the end of 1987, the control system realized the on-line control and monitoring for the most equipments of BEPC Storage Ring (SR) and its Transport Line (TL), and was put into precommissioning at this time. The beam orbit correction has been brought into commissioning in june of 1989. The RF on-line control (including its ramp) was completed in march of 1990. Thus all the equipments of SR and TL were controlled by the computer, and the VAX-11/750 computer (the central control computer) becames a member of DECnet of IHEP at the same time.

From the operation experiences in the past several years, it seems that the philosophy and criterion for constructing the BEPC control system are available, which resulted that this system can be completed just on time schedule and have a high reliability of system operation.

2. Several problems in system

The BEPC control system architecture follows the old centralized control model of the SPEAR, we had done incessantly some improving work on system level in the past

years, but still there are several problems which can't be solved. These problems are as follow:

1. The unique VAX-11/750 computer is used for the central control computer, its poor CPU power (only the 60 percent performance of VAX-11/780) limits the processing speed of whole control system; Many batch jobs (about 16 control processes) always reside in the memory, morever which heavily increases the load of VAX-11/750 computer system. These causes slow the response and processing speed of the entire system.

2. Due to we adopt the VAX-VCC-CAMAC (VCC, that is, VAX CAMAC Channel) hardware system architecture, one VAX QIO in CXCAMAC program takes 20 ms at least, so the VCC forms the bottle-neck of the control system communication. Especially it is obvious during the power supply ramping, the other jobs can not be serviced quickly.

3. The current human-machine interface is supported by two graphic colour monitors (resolution 512 × 512), two touch-screen with several computerized knobs. Now one monitor is occupied by one picture once a time, if we want to see several pictures of the different requirements of accelerator commissioning simultaneously, it needs to add more graphic display.

4. A fatal failure of the VAX-11/750 computer in the night (no on call service for computer) will break down the collider's operation. So a backup computer is needed to be considered.

3. System upgrade

After we analyzed the present conditions, we decided to reform the control system from the current centralized system to the distributed system and intend to solve those problems for improving the whole system performace.

a. The distributed control system architecture

In order to reduce the heavy load of VAX-11/750 and increase the speed of whole system responses, we divide the current control jobs on VAX-11/750 to several parts and load them into different computer nodes. the upgrade system will be based on the DECnet, its system architec-

ture is shown in figure 1. The new added VAX-II computer is used as a intelligent node dedicated to the power supply control. The current VAX-11/750 also is a intelligent node which mainly is used for other subsystems, such as BPM, RF, VACuum. In the first phase, we will still use the current human-machine interface as the system console. In the second phase, we will add another DEC-WS as upgrade system console which can utilize fully the rich DECwindow's software functions and make the multi-display for improving the environment of human-machine interface. Some of the jobs involved a lot of calculation will be moved to the DEC-WS. These three nodes will be networked into a individual local network which is linked to current IHEP DECnet via a network bridge.
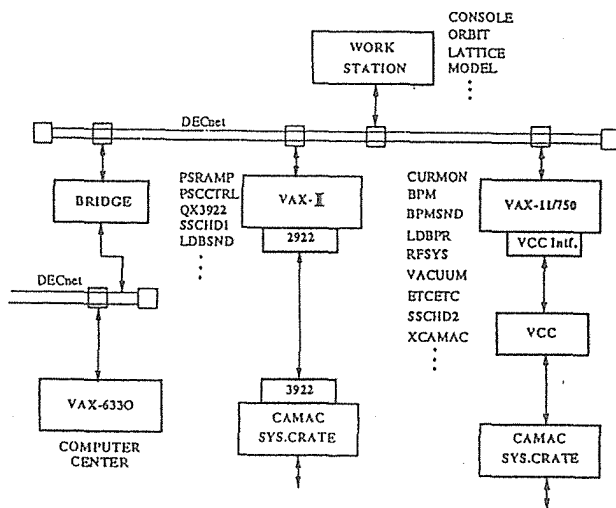
Figure 1. The upgrade of BEPC control system

b. Adopting Q-bus CAMAC adaptor

In the current system, all the datum are passing through the VCC, one QIO of VMS for VCC need about 20 ms at least and obviously reduce the processing speed of whole system. Beside the VCC adaptor is an old SLAC dedicated product, not a commercial module. Therefore, we adopt the Kinetic Systems Corporation's KSC 2922 and 3922 commercial modules as the VAX-II CAMAC adaptor. Under the support of KSC's, a VAX QIO operation only takes 3 ms in 1,000 words under the VMS, it would overcome the bottle-neck of the VCC and also increase the system reliability.

c. The compatiblity on new and old system

For keeping the compatiblity of CAMAC system, we are not going to do any modification on the hardware below the CAMAC system crates as far as possible. Thus after the upgrade is completed, the current control system on the VAX-11/750 can become the backup on Micro VAX-II. Conversely, if VAX-11/750 computer is out of work, the

collider operation can keep continuously due to the power supply still be controlled by the VAX-II machine.

4. The upgrade of BEPC control software system

The upgrade work of the control system is mainly focused on the software side, the block diagram of system software upgrade is shown in figure 2, which can be divided into the following three parts:

1. Creating new control software system for power supply control on the VAX-II.

2. Remoulding the current control software on the VAX-11/750.

3. Coordinating the relations between VAX-II, VAX-11/750 and DEC-WS for integrating whole distributed control system.
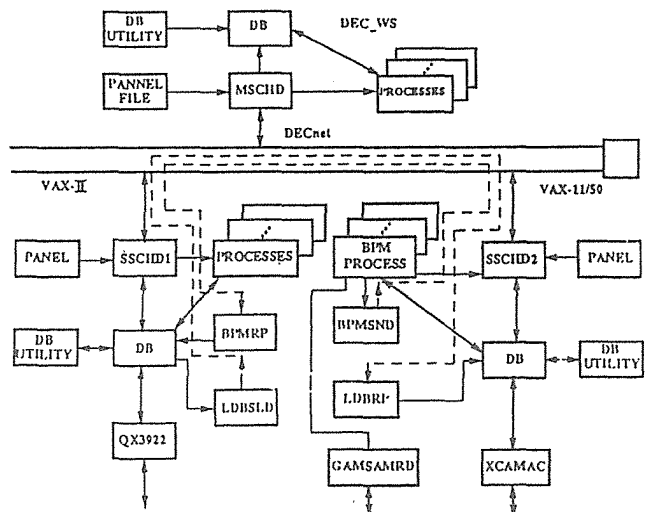
Figure 2. The upgrade of BEPC control software system

a. Upgrading procedures

First, we copy and modify a second set of BEPC control software system and its VMS operating environment on the VAX-II machine, and also modify and create the real time database (DB) on the VAX-II and DEC-WS as like as on VAX-11/750. Under such arrangement, we can reuse same panel files and all of the current application programs, reduce the software work to minimum and keep all the operator procedure as before.

We need to develope some communication programs for exchanging the necessary messages between three nodes on

the DECnet. These programs are: a.) The schedule communicating program —— used for transferring the schedule control and processing message (as position coordinate of touch panel or cursor on DEC-WS later, knob datum etc.) of console node (either VAX-11/750 or the DEC-WS later) to other process nodes. b.) The power supply DB refreshing program —— the real time DB of power supply is resided on VAX-II, this program is used to send and receive all of the power supply DB parameters form VAX-II to the console computer for refreshing the DB on the console node continuously in 1 time/second rate. c.) Other communicating programs —— used for transferring the control and the relative information of BPM, RF and VAC subsystem between VAX-11/750 to VAX-II and DEC-WS.

We prepare to use the DEC GKS graphic packets or the UGS graphic system of SLAC to upgrade the current graphic programs. In order to remain and keep the current operating environment for operator and reuse all of the application programs, we want to save all the previous subroutine names and only change the contents from old graphic program to new upgrading graphic program.

According to our current upgrading police, we still want to keep and save all the current programs on each node. Therefore, we introduce a "Pseudo-Subroutine Method" (PSM) in the programs, i.e., if one process is needed to be activated, only the process resided on the active node is functioned, the same process on other nodes keep quiet without any activity. This PSM can reduce the upgrade work amount and short the upgrading time. Of course, it will introduce some disadvantages, such as, occupy some memory resource etc, which will be considered to improve it at later.

During the second step, when the DEC-WS will be used as the system console, we will transfer some application programs used for modelling-based control, such as Lattice, Modelling and Orbit correction processes to the DEC-WS later. Thus we can construct our minimum distributed system architecture under our tight budget condition and establish a basic distributed system environment for later development.

b. Some progresses at present

So far, we have got some progresses in the upgrade of software on the VAX-II during this summer shutdown of BEPC. In order to run the BEPC control software on the VAX-II, we have modified and changed some parameters and commands procedures concerned with BEPC application system resulting to optimize the environment of the upgrading control software; We have created the real time database and have debugged all of the DB utilities on VAX-II; Now, the process and subtask can be submitted and activated by the schedule (SSCHD1) process through calling the panel files which is same as on the VAX-11/750 while we touch the current touch-screen; When the BPM process on the VAX-11/750 is once activated, the BPM buttons datum can be acquired individully by CAMSAMRD program instead of the XCAMAC. The BPM process also calls BPMSND program to active the remote process BPMRP on VAX-II and send the latest BPM datum to it via the DECnet, thus the BPMRP refreshs the BPM area of DB; The status and parameters communicating process of power supply and the graphic program substitution which is based on the work station also got some progresses.

Acknowledgements

References

[1] Shi-Yao Liu and BEPC control group, BEPC computer Control System, presented at "Proceeding of the 14th International Conference High Energy Accelerators", August 22-26, 1989, Tsukuba, Japan. Page 899.

[2] Shi-Yao Liu, Accelerator Control for Beijing Electron Ring and Its Experience, presented at "The 14th HiSOR Meeting", Hiroshima University, Japan, March 5-6, 1990

[3] S.Howry etc. Portable database driven control system for SPEAR, Report on 1985 particle accelerator conference, Vancouver, B.C., Canada

[4] R. Appalaraju and Robert T. Cleary, VAX-Based Realtime data acquisition for Scientific Measurements application, Report, NSS, Orlando, Florida, Nov 9-11, 1988

The rejuvenation of TRISTAN control system

November 1, 1991

T. Mimashi, J. Urakawa, S. Kurokawa, .T. Kawamoto, S. Takeda,
A. Akiyama,. K. Kudoh,. K. Komada, T. Naitoh

National Laboratory for High Energy Physics (KEK),
Tsukuba, Ibaraki 305, Japan

abstract

The current TRISTAN accelerator control system uses CAMAC as a front end electronics, and they are controlled by twenty five Hitachi minicomputer HIDIC 80's which are linked with an N-to-N token ring network. After five years from now, these computers must be replaced. This is because of the life time of control system and we have to cope with the requirements imposed by our future project such as the KEK B-Factory and the main ring photon factory projects. The rejuvenation of this control has to be done under some constraints such as the lack of manpower, limited time and financing. First we review the problems of current control system, then the philosophy of the new generation control system is presented. Finally it is discussed how to move to the new generation control system from the current TRISTAN control system.

1.Introduction

Eight years have passed since the current control system started the operation of TRISTAN accelerator [1] [2]. We have a few weeks hardware maintenance for every six month to keep our system highly reliable. The Hitachi company takes care of these maintenance; they clean up each computer, check its operation and replace some parts such as fans, filters, etc. The disk of each computer is replaced every five years. Hitachi also supports 24 hour "on call shift" for any computers troubles. Supported by these maintenances, the current system has been working very stable for these eight years.

Recently some of the I/O devices become out of date, and it becomes difficult to repair these devices. In addition to the problem, the current system can not satisfy the increased requirement of accelerator control. The latter is mainly caused by the lack of CPU power and by the fact that the process computers are all 16-bit machine. And the network transfer capability is limited by the lack of CPU power. In order to solve these problems, it is about a time to start thinking about rejuvenation of TRISTAN control system.

2.The TRISTAN complex and its control system

The accelerator complex of TRISTAN consists of three major accelerators. A 400 m electron linac accelerates electrons and positrons up to 2.5 GeV and injects into the accumulation ring. The accumulation ring accelerates them up to 8 GeV and injects them to the main ring. The main ring is an electron-positron collider . Two electron bunches and two positron bunches are circulating in the opposite direction and collide with each other at the mid-points of four straight sections. The current experiment runs at the center of mass energy of 29 GeV [3].

There are two independent control systems for their operation, one of the system takes care of the operation of linear accelerator and the other takes care of that of the accumulation ring and the main ring. This paper discusses about the rejuvenation of the later system [2].

The TRISTAN accelerator control system uses twenty-five minicomputers and two large general-purpose computers (The main frame: Hitachi HITAC and Fujitsu FACOM) [4].

The minicomputers control hardware equipment and serve for man-machine communication. The general purpose computers are used for the calculation of closed orbit distortion correction which overload the minicomputers. The 25 distributed minicomputers, Hitachi HIDIC 80's, are connected by optical fiber cables to form a 10 Mbps N-to-N token-ring network. From each minicomputer, a CAMAC serial highway is extended to the hardware equipment.

The minicomputers are classified into two groups: the system computers and the device-control computers. The system computers control six operational consoles and two alarm-processing. The device-control computes are three minicomputers for RF cavities control, five for magnets control, two for vacuum controls, two for beam monitor, two for beam transportation, one for timing control, one for program development, one for a gate way between the control system and main frame and one for general purpose.
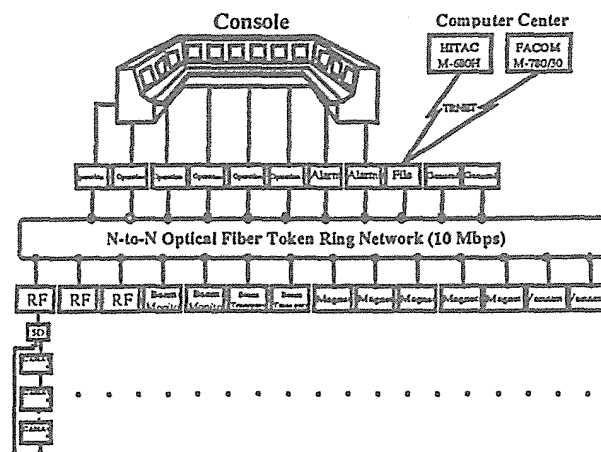
Fig 1: The current TRISTAN accelerator control system

Accelerators are operated through six identical operator's consoles which consists of two high-resolution color graphic CRT terminals and a pair of touch panels and a VT100 terminal used for a program development.

In the TRISTAN control system, the NODAL system was chosen as an application environment of the system. NODAL, a multi-computer language devised at CERN SPS, has been implemented on HIDIC 80's with enhancements such as a fast execution speed, a dynamic linkage scheme for external subroutines, etc. NODAL has multi-computer commands which enable us to transmit NODAL program from an operational computer to a device-control computer through the network and execute it on the device-control computer. All high level application programs are coded in NODAL and low level subroutines called as the data module are written by the PCL language that is the FORTRAN like language on HIDIC.

3.The plans of rejuvenation of the control system

The basic design of current control system was made by KEK and built by the company in the beginning of 1980's. Since there weren't any good standard products in that time, the proprietary system was built using their own network and process computers. And they developed software tools under their special environment. For example, NODAL is developed using PCL language. This is one of the typical case to build the control system in the industry.

department wrote all application programs, and the number of programs has been increased.

Using this kind of relationship to the company, we could build the high quality control system in the short period with small number of staffs. The staff could concentrate on the application program development.

But it results in the single vendor system and the low degree of standardization, for example the system uses special network and special PCL language so that it causes a problem when we try to add the new technology to the system.

The new system should have high reliability, high flexibility, high maintainability, high performance, high extendibility, real time response and at the same time it is important to use multi vendor computers and the standard network protocol. In order to establish this system, the system should be simple and well modularized from the point of view of hardware and software.

The system architecture

The current system consists of two layers. One is the front end layer which is composed of about three thousand of CAMAC modules. And the other is composed of consoles and process computers.

The first step of the system rejuvenation does not touch the front end layer at all, so that only the computing environment will be replaced.

The distributed CPU's and the data base can provide high flexibility and high extendibility.

Fig 2 (A) : The possible new control system for TRISTAN using the real time process computers
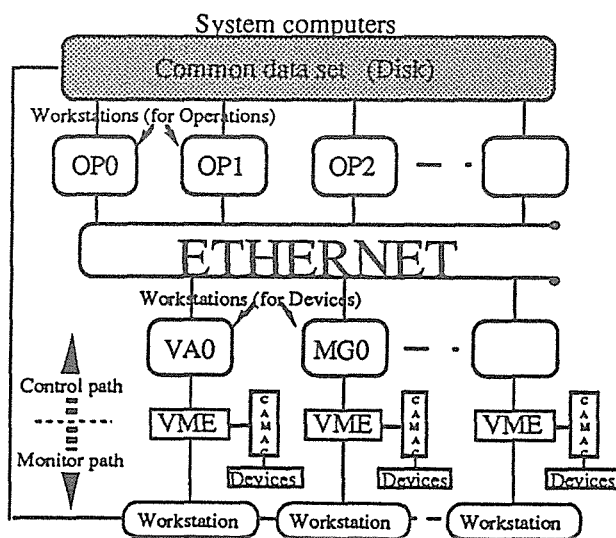
Fig 2 (B) : The possible new control system for TRISTAN using VME modules.

Some of the industries order the other company to make full control system which includes their special application programs. Once their system was established, there is no change in the application program. People in the industry only takes care of ' Turn Key Operation '. The difference between our control system construction and that of industry is that people in the accelerator

The discussion points are how to link these CPU's and how to handle the real time processes. The system architecture should be simple and the status of each CPU can be seen easily from operator consoles. From the point of this view, the number of hierarchy in the depth direction of the multi-computer system has to be minimized.

The real time response

In order to get high performance in the reasonable price, Unix workstations and VME modules are good candidates as components of the system. Since a Unix workstation does not have real time response, we have to prepare the real time system using VME modules or special real time process computers that are not well standardized yet. Two examples of the real time system are described in Fig. 2 (A) and Fig. 2 (B). Fig 2 (B) system uses VME modules connected to the Unix workstation to proceed the real time task. All processes that need real time response should be taken care at the VME level, so that the communication between operational consoles and workstations for the hardware equipment control is not necessary to be a token ring which has a real time response.

On the other hand, Fig 2 (A) uses real time Unix or VMS workstations to control real time processes. Since real time Unix and VMS workstations are vendor dependent, the system that deals with real time processes has to be minimized for the `future update of the system. And these real time computers must be linked with a token type network such as FDDI.

The software development

The environment of software development is one of the key of successful system. The language should be available in the standard workstation. And programs have to be written easily. NODAL and Object oriented programing are good candidates. We should think about the usage of the commercial software package to save the time.

4.The modularization of the system

When we rejuvenate the control system, the key of the rejuvenation is modularization, because they have to be replaced step by step. In this section, the possible modularization with respect to both the hardware and the software is described.

The modularization of the hardware system

There are some possibilities in the hardware modularization.

Hardware Modularization

Monitoring ── Vacuum Monitor / Beam Monitor / Alarm Monitor

Control ── Magnet Control / RF Control / Beam Transport

Timing
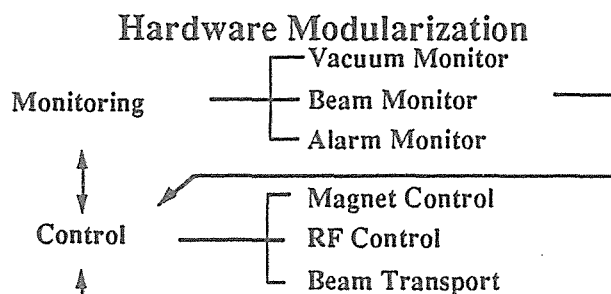
Fig.3 Separation of monitor path and control path

The monitor path can be separated from the path for device control. (Fig 3) The real time response is necessary for only the path of the device control. As same as the current system, the accumulator ring and the main ring can be modularized. It is also possible to modularize the system due to the function of each system, such as vacuum control, RF control, magnet control.

The modularization of the software system

The modularization of software is very important to rejuvenate the system and also important for the object oriented programing. The discussion in this context based on the four levels of control process.

Level 0 : The interface against operator. It must be possible to generalize the action of operators, such as 'select menu', 'input number', 'display data'.

Level 1 : The programs of this level should be written as the words of accelerator operation. For example, 'Measure Closed Orbit Distortion', 'Accelerate the particles','Change tune'.

Level 2 : The programs of this level are discussed as the protocol of the magnet or beam instrumentation operation, etc [5]. The objects of this level are the equipment of an accelerator such as a magnet, a vacuum pump, etc. And there should not depend on the sort of hardware interface, i.e. CAMAC or VME.

Level 3 : For a while, CAMAC has to be used for device control. All CAMAC actions should be defined in this level.   When we want to replace some of the CAMAC modules to VME or other hardware modules, the software modification should be done in this level only. The codes should be put in a object library for each hardware individually so that it can be possible to update each hardware control system independently.

Object

Operator
Level 0 — Menu Items
Level 1 — The words of beam operation / Has to be written as Words of Accelerator
Level 2 — Component of an accelerator
Level 3 — CAMAC / Call CAMAC Functions
Equipment
RF BM VA ··· MG

Fig 4 : The software modularization in the object oriented programing for the accelerator control

In each level, the object can be defined easily in the object oriented programing. Since they are well modularized, if CAMAC is replaced by other front end modules, such as VME, the software rejuvenation can be done easily. There must be no effect in level 0 - 2. And

the part of the programs in level 0 - 2 can be shared in the different accelerators, for example B-factory and the main ring synchrotron radiation project.

5.The transition to the new generation control system

We have just started rejuvenating our control system. Three Unix workstations are installed and we start including them for our accelerator control. One workstation is going to be used to communicate with the linac control system. Since the current control system can not control linac parameters, such as the beam energy, the beam position of the injection line, it will be used to monitor and control some of the linear accelerator parameters and at the same time it sends some informations of TRISTAN ring to their control system. The other workstation will be used to do simulations for an accelerator control. The current control system has used PETROK that is the program for the computation of the closed orbit distortion correction [6]. And recently associating with the implementation of the superconducting Q-magnet, the SAD (Strategic Accelerator Design) which is the set of programs for simulation and optics matching, started to be used very frequently for the TRISTAN accelerator control [7]. The SAD programs are developed for accelerator design in KEK accelerator department. All programs are developed in the main frame: Hitachi HITAC. But the recent drastic improvement of the calculation speed of workstation make it possible to run the simulation in the private workstation. Since the current control system does not support standard network, the project to transfer the data from the current control system to the new installed workstation through CAMAC has just been started.

Because the current process minicomputers are 16 bit machine, the lack of memory causes some troubles.

The graphics has critical problems, so that it will be replaced by the graphics on workstations. In order to remove the current graphic system, the current software co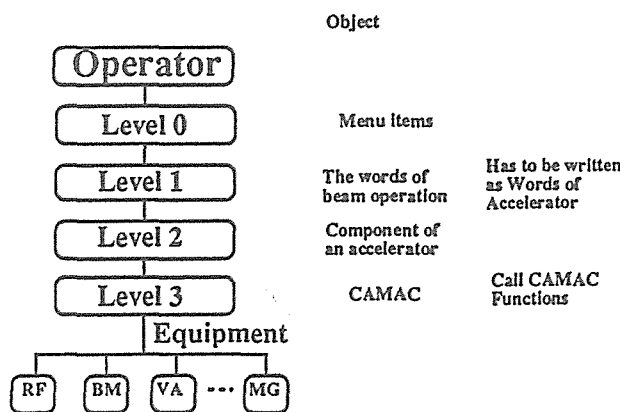des also must be available in the new workstations until all graphic programs are rewritten. The current graphic system "HIDIC 80-RS232-Graphics controller-CRT", will be replaced by "HIDIC 80-RS232-Xwindow in the workstation"

The new functions for the accelerator control, such as the real time tune monitor, is going to be constructed with the graphics in the workstations

The possible system in the transition stages

There are several possibilities of the system rejuvenation. One possibility is to install another crate controller into each CAMAC crate so that two independent system can exist until the new system is completed. In any case, the replacement must be done step by step. The biggest change will be network support, unfortunatly at this point, the step by step replacement is impossible. Since the number of programs for man-machine interface are much more than that of hardware control, one possibility is to replace operational consoles with new network system in the early statge, and connect the new system and the old system through the gate way. For a while, both old and new operational consoles must be available. Then hardware process computers can be replaced one by one.

There are several options of this replacement. For example, 1: replace monitoring path such as a vacuum monitoring system, an alarm system first, then replace control path such as a RF control system, a magnet control system.

Actually there have already existed the independent RF control system composed of VAX workstations for taking care of aging of RF cavities. The system can be available for the system replacement.

Another method is 2: replace process computers for accumulator ring operation first, then replace that of main ring operation.

These hardware replacement has to be done together with the software emulations. But the software replacement is more complicated. One of the approach is to try to borrow NODAL program coded in C language from CERN. But even in such a case, all data module subroutines coded in PCL language have to be rewritten in C.

6.Conclusions

The rejuvenation of the TRISTAN control system has been just started to satisfy the increased requirements from complex operation of the accelerator. The Unix workstations and VME's are good candidates as a component of the new system, and some of them have already been added in the current system. The several R&D 's are underdevelopment in both hardware and software. We are going to choose one of the best solutions, but the basic ideas are

1: Use standards and make the open system. 2: Modularize the system and replace them step by step. 3:Use a commercial products to save a time and concentrate on the application program development.

References
[1] S.Kurokawa et al., Nucl. Instr. and Meth. A247 (1986) 29
[2] H.Ikeda et. al., IEEE Trans. Nucl. Sci. NS-28 No.3 (1981) 2359
[3] S.Kamada et. al., KEK Preprint 91-76 (1991)
[4] E.Kikutani et. al., The 6th Symp. on Accelerator Science and Technology. [1987] 236
[5] R.Bailey et al., Nucl. Instr. and Meth. A293 (1990) 332
[6] PETROK is KEK version of PETROS which is developed at DESY.
[7] K.Oide, Nucl. Instr. and Meth. A276 (1989) 427

Upgrade Plan for the Control System of the KEK e−/e+ Linac

K. Furukawa, N. Kamikubota, K. Nakahara and I. Abe
National Laboratory for High Energy Physics (KEK)
Oho, Tsukuba, Ibaraki, 305 Japan

Abstract

The KEK 2.5-GeV linac has been operating since 1982. However, in order to maintain reliable operation, the control system should be upgraded within a few years. We plan to replace the minicomputers and the main network connecting them. Thus, the architecture of the control software will also be revised. In the new system we should adopt software and hardware standards.

In the next control system we will employ the TCP/IP (DARPA) protocol suite for the main network and Unix workstations to replace the minicomputers. For connections to the local controllers, VME bus (IEEE 1014) will be utilized.

I. INTRODUCTION

The KEK 2.5-GeV linac provides electron and positron beams for both the TRISTAN collider and the Photon Factory storage ring. The linac control system was designed in 1979 and has been successfully operating since its commissioning in 1982. This old system is based on a distributed processor network, comprising eight minicomputers and hundreds of microcomputers, and on a control message exchange. Detailed descriptions have been given elsewhere [1] [2].

We have recently had many requests for increased functionality of the control system not included in the original design. However system resources were inadequate to implement the desired functionality. Furthermore, the minicomputers and the associated main network used in the old system will become unsupported by the computer company in a couple of years. We had thus decided to replace the main part of the control system and have carried out studies regarding the new system.

As a result, we have decided to adapt international and de facto standards as much as possible. The recent advances in electronic technology enable us to use common standards among various fundamental components. We have employed these standards in the new network system and computers. For the main network the TCP/IP (DARPA) protocol suite on the Ethernet (IEEE 802.3) media is utilized. Unix operating systems on a group of workstations and VME-bus-based (IEEE 1014) computers were chosen for the main console stations and subcontrol stations, respectively.

We describe below the design, current status and future of the new linac control system.

II. SYSTEM COMPONENTS

A modern control system must interlink many kinds of objects and facilities. We may link it with advanced technology, such as A.I. systems [3] or CASE tools. We may also link to the control systems of other accelerators or utilities, although some access limitation mechanism is needed. We may further link the new control system at the next upgrade time. Since the accelerators for research projects are always improving, we must combine many kinds of components to the control system. Thus we must make the control system as flexible as possible.

In order to fulfill the requests we cannot rely on only one hardware and software vender. We had better utilize international and de facto standards. If we have much capacity to develop control components, we may define our own standards. However, this would lead some difficulties in future, as we have already experienced. We thus want to utilize currently available standards.

The lifetime of an accelerator control system is relatively short, since electronics technology makes rapid progress and demands for control systems are changeable. We must thus always worry about future changes and upgrades of the control system. Since new technology and future standards are very likely to support existing standards, employing standards is promising.

A desire for standards has grown among both users and vendors recently. If we adopt standards, (a) since there will exist less ambiguity in the rules, we can develop reliable technology based on it, (b) since the required components may be obtained commercially, the development period may be substantially reduced, (c) since the standards in other fields may support each other, it may be easier to construct the entire system. These facts are especially important in a small system, such as in our case where manpower is limited.

A. TCP/IP Network.

Since equipment used for a accelerator is spread over a long distance, we must use some kind of computer networks between controllers. Since our linac is a pulsed high-power RF machine, in the old system we attached importance to noise elimination and have used proprietary fiber optic networks (Loop-1). Since their hardware and software are dedicated to the old system, we cannot utilize them for any other configurations.

Thus, we had to think about introducing a more standardized network, which can be adopted to various configurations. In order to be independent from suppliers TCP/IP protocol suite (DARPA) is most appropriate. We

have decided to use TCP/IP, even on VMS operating systems where the DECnet protocol is available.

We have developed programs which test the basic communication capabilities of TCP/IP, which proved capable of running on more than 15 different computer/operating system architectures without any modifications to the source codes. Performance measurements were also carried out with these programs, and a part of the results is shown in [4]. The round-trip response time of the typical workstations over the Ethernet is about a millisecond, which is appropriate for a control network.

For the network media we utilize Ethernet (IEEE 802.3) since its components are commercially available and cheap. We know that the CSMA/CD mechanism of Ethernet is less effective, compared with token passing. However, we can make use of it up to 5 M bit/s, which is adequate for our current needs. It is possible to replace the main network part in the future, without any modifications to our software, with FDDI network system, which provides a band width of 100 M bit/s and token ring network access mechanism.

B. Unix Operating System

Unix is not presently a real-time computer operating system. However, except for full priority scheduling, most real-time features have already been implemented into many Unix operating systems. At least the current Unix systems can have a better real-time response than the real-time computers existing 10 years ago. Synchronization between processes can be possible in a millisecond if the processes are locked on memory; it took about 10 milliseconds to switch between processes on the old minicomputers. It should be noted that the POSIX (IEEE P1003) for Unix standardization tries to include real-time capabilities.

The system call incompatibility between Unix systems from many venders, especially between BSD-based and System-V-based Unix systems, could cause some troubles. However, most of our software can be easily ported, since we tried not to use any unusual system calls. We currently develop software mostly on DecStation 5000's in the control system and SparcStation 1 on the laboratory network. We also use the Mitsubishi's MX3000II for the gateway between the old and new network systems, as described below.

C. VME Front End

In the old system one subcontrol station comprises a minicomputer and a CAMAC crate. Since we have only a few kinds of CAMAC modules and the VME bus standard (IEEE 1014) is widely accepted, we will replace the subcontrol stations with VME-bus-based systems.

The OS9 operating systems and MC 68030 processors drive the VME systems. We have already finished examining the TCP/IP communication capability. We made the same network software to run on Unix and VME-based systems. The results satisfied our purpose.

In order to communicate with local device controllers we continue to utilize fiber optic in-house local networks (Loop-2,3) for some time, since there exists about 300 microcomputer-based device controllers [5]. The VME modules for the Loop-2,3 network system and its associated software are under development. Newly designed device controllers may use VME systems and communicate directly to the main network.

III. ARCHITECTURE

A. General Network Service

We are using some general network services, which includes: (a) time synchronization, (b) file sharing, (c) printer sharing, (d) computer and network status monitoring, (e) relational database access, (f) logging message distribution, and (g) name services. These mechanisms are achieved without any effort by using standard network protocols.

Currently, ntp (network time protocol) is employed on the Unix operating systems and VMS systems for time synchronization [6]. Modifications to the source codes were needed on the System-V-based Unix systems. With ntp, the time difference between computers is regulated using a statistical method; under normal operation it remains at about several milliseconds. This is very important under a distributed environment, where common resources are shared between many processes on the network, in order to keep consistent information.

The NFS service provides a homogeneous disk file system access over the network. The components of operating systems are shared as well as the control system databases and system development environments. Recently, computer equipment has become very reliable. However, hard disks can easily fail and data cannot be saved after crashing. It is thus important to reduce the number of hard disks as well as to save the contents frequently. We intend to run most VME-based machines and some workstations using NFS without local disks.

B. Message Exchange Controls

Since the old system runs based on control message exchange [2], in the new control system it must be supported in order to keep the old software running. The old control messages depend on the physical structure of the network connection and the characteristics of the associated equipment or service. In order to adopt control messages to the new control system we have also developed a suite of new control messages with more realistic symbolic messages and object-oriented concepts [4]. We have already developed a message-conversion process between old and new formats.

C. Servers

In the old system although many processes are loaded onto the main console station., the number of processes running on

3rd Int. Conf. Accel. Large Exp. Phys. Control Syst.   ICALEPCS1991, Tsukuba, Japan (partial, pages 194 of 611)   JACoW Publishing

ISBN: 978-3-95450-254-7   ISSN: 2226-0358   doi:10.18429/JACoW-ICALEPCS1991-S02SRU09

the station is limited. It is thus difficult to expand the control system functionality. On the new system we should make the control software independent of the number of control workstations.

The response time measurement mentioned above suggests that the processes on a main console station can be distributed among many workstations, since the response time between the processes on the new network system is much faster than that on the same old main station.

In order to distribute the control processes, we should develop several processing servers which would function as an operating system on the control network. These servers should include: (a) a static database server, which would provide characteristics of equipment and services, and (b) a dynamic database server, which would distribute live accelerator status. These services need some caching mechanism on each computer. We should also provide: (c) a locking server, which could carry out resource management, (d) an alarm server, and (e) a communication server, which could manage communication to other accelerators or facilities. With these server processes the network system would act as one computer.

D. Surveillance and Diagnosis

We have realized that under normal accelerator operation the status and fault surveillance system is important for maintaining stable operation [7]. The purpose of the control system is to operate the objects based on some physical models. These models should be performed in the control processes. However, it is always possible that the equipment or software may fail. Surveillance processes should detect any failure and alarm the operators. The database information servers and alarm servers described above would help these processes. Preferably, fault recovery should follow fault detection with a diagnostic system.

Currently, most surveillance tasks depends on human operators. We thus need some simple method to describe the conditions. Otherwise, we must put complicated surveillance description into the software. Object-oriented software techniques will aid the in development to some extent. We also intend to utilize rule-based real-time expert systems.

E. Remote Procedure Calls

Although we are now developing control software based on message exchanges, we are also considering using the mechanism of remote procedure calls (RPC) in order to improve the efficiency of the system and application software production. We implemented the NC/RPC system of CERN's [8] and RPC of Sun Micro System's into DecStations and a SparcStation, which gives good results. With some improvements for software development environment and asynchronous RPC, we will use the RPC in near future.

IV. SCHEDULE

A. Condition

The operation of the accelerator doesn't allow us long shutdown periods of the control system. We must therefore replace the system components gradually.

The original system comprised 6 components as already described: (a) two minicomputers as the main console stations, (b) six minicomputers as the subcontrol stations, (c) a fiber-optic proprietary network between minicomputers, (d) hundreds of microcomputers for local device controllers, (e) about twenty fiber-optic in-house local networks, and (f) an operator's console subsystem of personal computers [9]. We will replace the first three components during the first stage. Then others will be improved.

B. Gateway Between Old and New Systems

Since it's risky to completely replace the system components all at once, we have built a control message and status information gateway between the old and new systems [2]. The gateway comprises software on the old and new computers which transfer information through parallel interconnections. Messages can be transmitted from the new network system to the old system, and vice versa, without worrying about gateway software.

Several application programs, such as equipment status surveillance which was limited in the old system, work fine over the gateway. The message-transfer scheme was also a basic concept in the old system; thus many old applications can be moved to the new system.

We are currently developing software for the new system using this gateway. Although the response time on the old network side is large, the new software should well simulate the new system

We have no control computer networks to the TRISTAN control system. However, we need to communicate with each other for better operation. These systems will be connected through the TCP/IP network during this year. Using TCP/IP protocols, message-packet routing, as well as its limitation, is possible without using any additional software.

C. Main Control Workstations, Subcontrol Stations and Main Network

Since the original main network is a proprietary for old minicomputers, we cannot install these three separately. However it is not too difficult since the gateway system described above is already installed. We will first change the route of messages to the Loop-3 networks into new system. After that, the route to the Loop-2 networks will be switched. The physical connection and message flow will switch as shown in figure 1. The replacement is scheduled to take place over a period of two years.
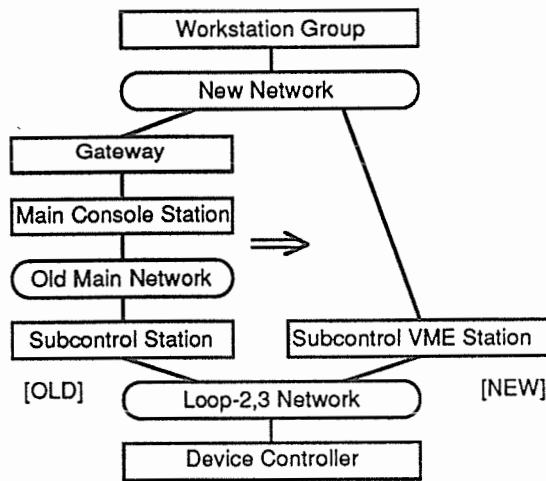
Figure 1. Transition of the system connection and the control message flow from the old system to the new system.

Later, the network traffic will increase since we share it also for software development. We will thus separate the network into two parts: a network for subcontrol stations and a network for main control stations. It is also possible to replace the latter with an FDDI network, which is more efficient, employing token ring network access mechanism. The total system is shown in figure 2.
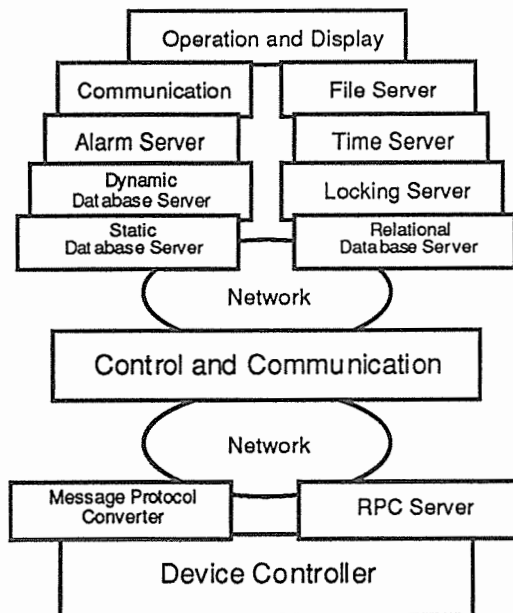
Figure 2. Logical structure of the new linac control system.

D. Further Improvements

Although we continue to use the old microprocessor-based local device controllers, for the new kinds of equipment we will utilize VME-based local controllers. During the next year we will obtain several pieces of new accelerator equipment,

that may be the first case to have new local controllers which are connected directly to the main network.

For the operator's console, a personal-computer-based system which enables frequent modifications, is utilized. However, the connection to its gateway personal computer is a serial link, and the new functionality on the new control system cannot be utilized from that system. We are thus presently developing some console functions on X-Window. The application development environment has improved much using various window toolkits and widgets. We will utilize Motif widgets for application programs. Some simple applications has been developed and an X display will be installed into the operator's console room during the next year.

V. CONCLUSION

An upgrade scheme of the linac control system was investigated and we have obtained good results so far. Using international and de facto standards, the upgrade has become much simple and we need not rely on any single company. This is important in small accelerator control systems, as in our case.

VI. ACKNOWLEDGEMENT

VII. REFERENCES

[1] K. Nakahara, I. Abe, A. Enomoto, Y. Otake and T. Urano, Nucl. Instr. and Meth. **A247** (1986) 153.
[2] K. Furukawa, N. Kamikubota, K. Nakahara and I. Abe, Nucl. Instr. and Meth. **A293** (1990) 16.
[3] I. Abe,, M. Kitamura and K. Nakahara, "Network communication libraries for the next control system of the KEK e⁻/e⁺ linac", this conference.
[4] N. Kamikubota, K. Furukawa, K. Nakahara and I. Abe, "A diagnostic expert system in PF linac control", this conference.
[5] For example, T. Shidara, R. P. Bissonnette, K. Nakahara and S. Asami, Nucl. Instr. and Meth. **A245** (1986) 500 .
[6] D. L. Mills, DARPA Network Working Group Report RFC-1129 (1989).
[7] K. Furukawa, N. Kamikubota, T. Shidara, K. Nakahara and G. Horikoshi, Part. Accel., **29** (1990) 245.
[8] P. S. Anderssen, V. Frammery and R. Wilcke, Nucl. Instr. and Meth. **A293** (1990) 225.
    K. Kostro, private communication.
[9] K. Nakahara, I. Abe, K. Furukawa and N. Kamikubota, Nucl. Instr. and Meth. **A293** (1990) 446.

The New Control System for TARN-2

S. Watanabe, M.Yoshizawa, J.Yoshizawa and T.Katayama
Institute for Nuclear Study, University of Tokyo, Tanashi, Tokyo 188, Japan

K.Aoki and K.Ohnishi
Sumitomo Heavy Industry, Tanashi, Tokyo 188, Japan

The new control system for the cooler-synchrotron, TARN-2, is described. The new control system consists of OPU's (work stations) and EXU (control computer) linked with the local area network. The text message is used to transfer the control commands and their results. The control program CSA90 at EXU decodes the text message and executes it with the aid of the interface and periodic control subroutines. Both subroutines use common sharable image composed of the status, values, parameters and so on. The CAMAC, GPIB and RS232C are standard interface at EXU.

Introduction

A heavy-ion synchrotron-cooler ring, TARN2 has been constructed at the Institute for Nuclear Study, University of Tokyo. It has a maximum magnetic rigidity of 6.2 Tm, corresponding to 1.1 GeV for protons and 0.37 GeV/u for ions of charge-to-mass ratio 1/2. Now the injector is a sector-focusing cyclotron with K=68. The aims of TARN2 are to study the acceleration, cooling and extraction of heavy ions and so on. Recently study of the weak beam monitoring and experiment of atomic physics is being continued. So the TARN2 control system must gives reliable control functions and monitoring required to direct the up-to-date complex operations associated with on site experiment.
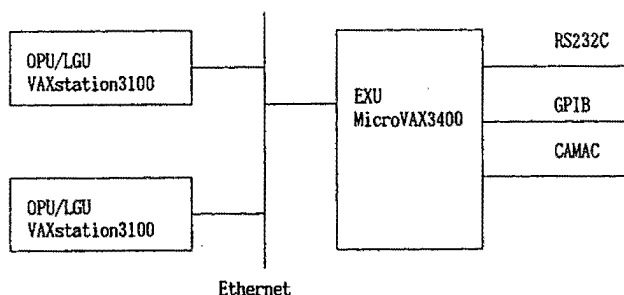
Fig.1 The new computer control system of TARN2

The present control system consists of the serial CAMAC, GPIB and microcomputers[1]. The serial CAMAC system covers static control system such as beam transport[2], injection, electron cooling[3] and a beam extraction system. On the other hand, dynamic control of both the RF acceleration and ring magnet system[4], is also performed with the dedicated microcomputers with external memory modules[5]. These systems well regulate TARN2 and now used without some troubles.

In 1990, basic design of the new computer control system was started and was authorized to combine the present I/O controllers. The present paper describes the new control system of TARN2 during the development phase.

Basic Architecture

In the new control system, the actual device control and man-machine communication is made by the separate units. The new system comprises OPU's (workstations) and EXU (control computer) linked with the local area network. The new computer control system of TARN2 is shown in Fig. 1.

The several kinds of workstations and PC's are used as OPU's. One is a VAXstation 3100. Another one is a microcomputer NEC-PC9801 with Ethernet board and DECnet-DOS. The uVAX 3400 is used as EXU and CAMAC, GPIB and RS232c are also available in the EXU.

The OPU and EXU execute task to task communications between them. Before the message transmission, OPU is linked with EXU. Then OPU and EXU transfer the text message to each other. The format of text message is formalized as shown in Fig.2.

At EXU, the received text message is stored into a buffer area by the interface process of control program CSA90[6]. Subsequently the buffered text message is taken by interface subroutine and then executed by a periodic subroutine after decoding the text message. Both subroutines use a common sharable image composed of the status, values, parameters and so on. Both subroutines are registered in the EXU and selectively called by the OPU using the subroutine number. The text message consists of this interface subroutine number and a command message associated with a control procedure of the target devices. So the central job at OPU is to generate the text message according to the associated control procedures. The interval of a periodic subroutine can be controlled by the associated control program CSA90. The minimum interval time is 50 msec and can be increased by 50 msec integrals.

The communication network 'INS-Ethernet' is a standard network in our institute, whereas other network system is used in our institute. The new control system

is linked with the local area network which is in the TARN2 and cyclotron areas. Our network is also linked with 'INS-Ethernet'. On the other hand, the large scale computer system INS-M780 is also available through the associated local area network if we need any large computing power. In this case, TCP/IP protocol is used and terminal server with LAT-TCP/IP protocol is available in the network system.

The log-in by the off-site users will be expected through the network. To reject such a problem, physical cut off of the network will be expected during the actual operation phase, but general protection method is also employed.

function	input ·	output
get device information number	0:D<devnam>	0:n
get interface subroutine number	0:R<subrnam>	0:n
get periodic control subroutine number	0:K<subrnam>	0:n
get device information name	0:n	0:<devnam>
examine information	0:@Cn	0:<string>
	0:@In(,m)	0:x
	0:@En(,m)	0:x
modify information	0:@Cn=<string>	0:
	0:@In(,m)=x	0:
	0:@En(,m)=x	0:

```
() = optional
i = subroutine number
n = item number of device information
m = element number of array
x = data
```

Fig.2 Format of Text Message of OPU/EXU

Operator Console

The distributed operator consoles provide many kinds of operating functions. Up to 10 operator consoles could be joined with EXU to execute task to task communications. The main aim of task to task communication is to transfer the text message as described before. On the other hand, operator consoles as workstations provides us with the highest quality of visualization, man-machine communication and calculation speed. we have been considering several types of man-machine communications such as a touch panel, rotary encoder, mouse and so on.

The NEC-PC9801 with the resistive touch panel sheet and rotary encoder is chosen. The mouse system is also used in this OPU. The ON/OFF control and value setting are the main job of this system. The operating system of MS-DOS 3.3B and MS-C, MASM and turbo-C have been used to develop the application control program. DECnet-DOS

is used in this system as the communication interface to EXU.

The VAXstation 3100 is used as OPU with VAX resources. The beam simulation program is available in this system. For instance, COD correction can be performed by the combination of OPU and EXU. The EXU sends measurement result of a beam orbit to OPU and receives back the command to change the correction current. The change of current is carried out by EXU.

Monitoring of the operating condition is an important facility. To discover the operating status, EXU periodically collects the operating status through the CAMAC system. The collected data is saved into a file and periodically refreshed by a timer process. The OPU assigned as a warning monitor refers the file in any time and executes appropriate fail-safe procedures.

Control Computer

The control program, CSA90, has been developed to execute the interface and periodic control process of EXU. The logical image of CSA90 is shown in Fig. 3.
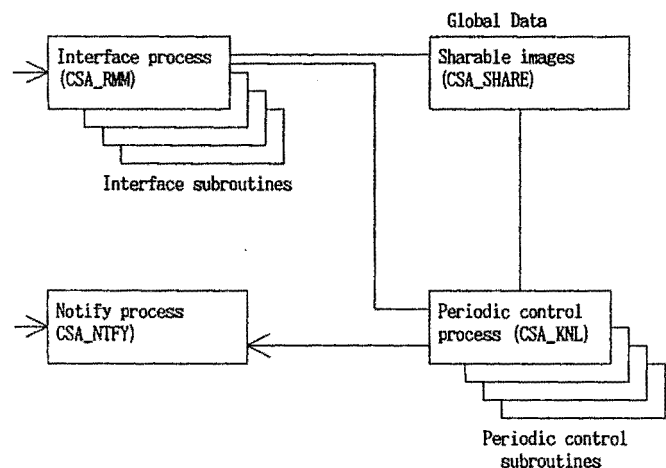
Fig.3 Logical image of control program CSA90

The CSA90 provides us with a flexible and expandable control system. This program has a basic program, device control programs, configuration definition files and a data generator. The device control programs are a set of subroutines and have no direct relation to each other. The basic program executes those subroutines according to a configuration program and requests from OPU. To make the configuration definition file easily, a data generation program is also provided.

By using CSA90, addition of new functions or modification of the control procedures are easily made by addition or modification of subroutines. Then the control sequence or configuration of the control system could be changed easily in according to the purpose of

accelerator studies and so on.

The I/O interfaces are comprised of CAMAC, GPIB and RS232c. Presently the main CAMAC crate is alternatively controlled by either the Kinetic 3922 and EXU or Kinetic 3920 and an old existing microcomputers. The serial CAMAC driver housed in the main CAMAC crate is used commonly by either EXU or existing microcomputers through the above mentioned crate controller. Several kinds of local control devices with GPIB are located near the RF acceleration system, electron cooling and elsewhere. They are controlled by EXU through the fiber cable. The CAMAC control system also supports GPIB through the local CAMAC-GPIB converter.

Local intelligent Control Devices

The distributed control devices with an intelligent function have been used in the TARN2 control system. For instance, pattern generator to regulate the magnetic field of the ring has been used in the synchrotron magnet power supply system. Present pattern generator is composed of the microcomputers and is directory controlled by the terminal at the control room. To include such an individual intelligent controllers into the network system, we have been considered an up-grade of existing intelligent one to new one.

Though the add-on board of Ethernet system can be used in the present intelligent local control system, it is difficult to construct the network communication system because it is depend on the operating system of the central processor of local control system. For more any

Fig.4 Proposed 32bit CAMAC board computer

unknown conditions, we have already considered that distributed local intelligent devices should be tied with the local area network. For instance, above mentioned pattern generators will be changed to a VME system with pattern memory system. The local CAMAC computer housed in the CAMAC crate is also considered as the candidate of distributed intelligent local control device. We have already been developed 8bit CAMAC module computer with a function of pattern signal generator, parallel bit signal input and output for any purpose. In this system, control program is stored as the ROM image and the RAM image. In Fig.4 the block diagram of proposed 32 bit CAMAC board computer is shown.

Conversion from present system to new one

The present microcomputer systems is written using an interpreted language such as INSBASIC with CP/M-86. To continue this approach with the present I/O control and the new control system, especially for OPU, a program converter has been developed from INSBASIC to QuickBASIC, which is used as the programming language for OPU. So the new control system will provide us with the user functions and subroutines associated with the present INSBASIC. Thus transparent programming can be performed in the new control system, especially for PC9801 as OPU.

For the FORTRAN system, such as the minicomputer U-400 or HP1000, the programs developed for CAMAC system will be implemented on the OPU and EXU. For instance, an existing program indicating the beam envelope of transport line will be reconstructed with MAGIC code in VAXstation 3100 and CAMAC handler in uvax 3400.

References

[1] S.Watanabe, The Computer Control System of TARN-II. Nuclear Instruments and Methods in Physics Research A293(1990)50-53.

[2] F.Soga, K.Noda, K.Chida, T.Hattori, T.Honma, T.Katayama, A.Mizobuchi, M.Nakai, A.Noda, F.Sakuta, M.Sekiguchi, N.Ueda, S.Watanabe and M.Yoshizawa, Beam Transport and Injection System from the SF Cyclotron to the TARN-II Synchrotron Cooler Ring, INS-T-494, February 1990.

[3] T.Tanabe, K.Chida, T.Honma, T.Hattori, M.Kanazawa, A.Mizobuchi, M.Nakai, A.Noda, K.Noda, M.Sekiguchi, F.Soga, N.Ueda, S.Watanabe, T.Watanabe, M.Yoshizawa and A.Ando, Initial Electron Cooling Studies in the TARN II. Proc. of the 7th Symposium on Accelerator Science and Technology, Dec. 1989, Osaka, Japan, p201.

[4] S.Watanabe, T.Kohno, S.Yamada and T.Katayama, Power Supply Control System for TARN-II Main Ring Magnet System. Proc. of the 6th Symposium on Accelerator Science and Technology, Oct. 1987, Tokyo, Japan, p242.

[5] S.Watanabe, A.Noda, T.Watanabe and T.Kohno, Computer Control of TARN-II Main Ring Power Supply. Proc. of the 7th Symposium on Accelerator Science and Technology, Dec.1989, Osaka, Japan, p249.

[6] K. Aoki, S. Watanabe, The Design of the New Control Program (CSA90) for the TARN2. INS-T-505, 1991.
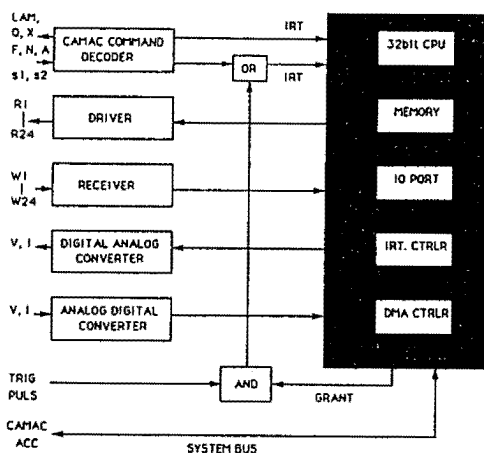
A New Architecture for Fermilab's Cryogenic Control System

J. Smolucha, A. Franck, K. Seino and S. Lackey

Fermi National Accelerator Laboratory*
Box 500, Batavia, Illinois, 60510

Abstract

In order to achieve design energy in the Tevatron, the magnet system will be operated at lower temperatures.[1] The increased requirements of operating the Tevatron at lower temperatures necessitated a major upgrade to the both the hardware and software components of the cryogenic control system. The new architecture is based on a distributed topology which couples Fermilab designed I/O subsystems to high performance, 80386 execution processors via a variety of networks including: Arcnet, iPSB, and token ring.

Introduction

The addition of "cold compressors" to the Tevatron's satellite refrigeration system, as well as the desire to dynamically balance the site's distributed compressor load, introduced additional performance requirements on the cryogenic control system. The required signal processing capabilities basically doubled, raising the number of I/O terminations to approximately 2,000 per satellite.

To effect global optimization the system's software also required significant enhancements. These included: support for global communications between refrigeration processors, an improved user interface to the finite state machines, both code and parameter down-loading capabilities, and substantially improved diagnostic support for the system's hardware.

Together, the performance criteria exceeded the capabilities of the original, Z80 based control system[2]. Therefore, a complete upgrade was warranted.

The new architecture

To achieve the desired performance with a minimum of hardware, we adopted a distributed architecture centered around Intel's 32 bit, 80386 microprocessor. Multibus II was selected as the hosting platform. We prefer it not only for the architectural advantages it offers, but also because of the existing support services which have already been developed for our front end computers[3]. The substantial amount of hardware and software that we were able to inherit allowed us to concentrate the majority of our efforts on developing smaller, more efficient I/O subsystems.

We partitioned the control system into six sectors following the physical distribution of satellite compressors around the Tevatron. Each sector consists of four satellite refrigerators and their associated compressor. The sectors are connected by a token ring network to effect global communications.

Each satellite unit is controlled by an Intel, iSBC 386/120 single board computer. The five CPU modules reside in a common Multibus II chassis which is located at the compressor installation. The loosely coupled architecture of Multibus II is ideally suited to this application. It supports protected, independent execution for each processor module, while simultaneously providing fair access to centralized system services. These services include: a Tevatron clock distribution processor, a "machine data" (MDAT) I/O module, global shared memory, a DOS based 386PC/AT for system initialization and diagnostics, and a token ring processor for global communications.

Inter-processor communications within the sector are accomplished over the parallel system bus (iPSB) which, in conjunction with the message passing protocol of Mutibus II, functions as a 40 megabyte per second local area network. The crate configuration is illustrated in Figure 1.
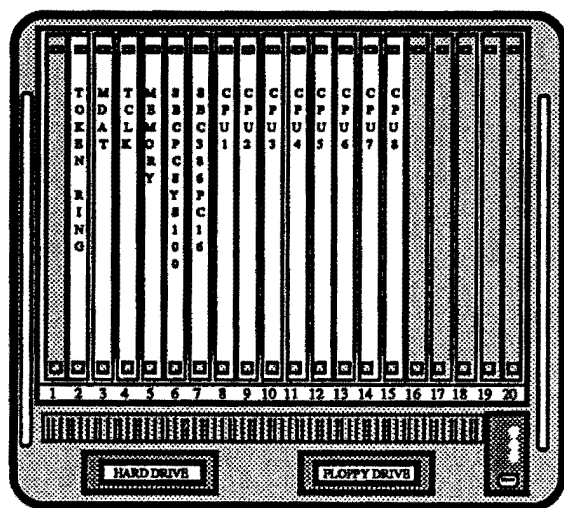
MULTIBUS II CRATE

FIGURE 1

Subsystem communications

Control for each of the satellite refrigerators is further divided into two I/O subsystems: Cryogenic Thermometry and Device I/O. (Compressors do not require thermometry.) The subsystems are located at the source of their respective signals and are connected to the execution processors by Arcnet, which functions as a small area network.

Arcnet is a low cost, highly reliable network that supports up to 255 nodes and can be implemented over a variety of media including coax, twisted pair and fiber[4]. It utilizes a deterministic "modified token passing" access scheme with a data rate of 2.5 megabits/second and a packet size of 512 bytes. It has a very simple protocol and all of the network services are provided at the silicon level by a variety of commercially available controller chips. The controller interfaces to a system through a page of dual ported RAM, and a single command/status register. To initiate a network transmission the host processor simply loads a message packet into the RAM buffer and issues a "transmit request". Message packets consist of a source ID, a destination ID, the message byte count, and up to 508 bytes of user defined data. All messages are accompanied by a 16 bit CRC character which the receiving node uses to verify the integrity of the transmission. Additionally, all error free transmissions are positively acknowledged.

There are no hardware imposed restrictions on communications within a sector. Each node is permitted equal access to all the other nodes on the network, thereby allowing a variety of logical topologies. Currently, a Fermilab designed protocol layer called "Virtual I/O Bus" (VIOB)[5], manages communications between the execution processors and the I/O subsystems. VIOB enables the subsystems to appear as extensions of the execution processor's local buses. The current topology for a single sector is illustrated in Figure 2.
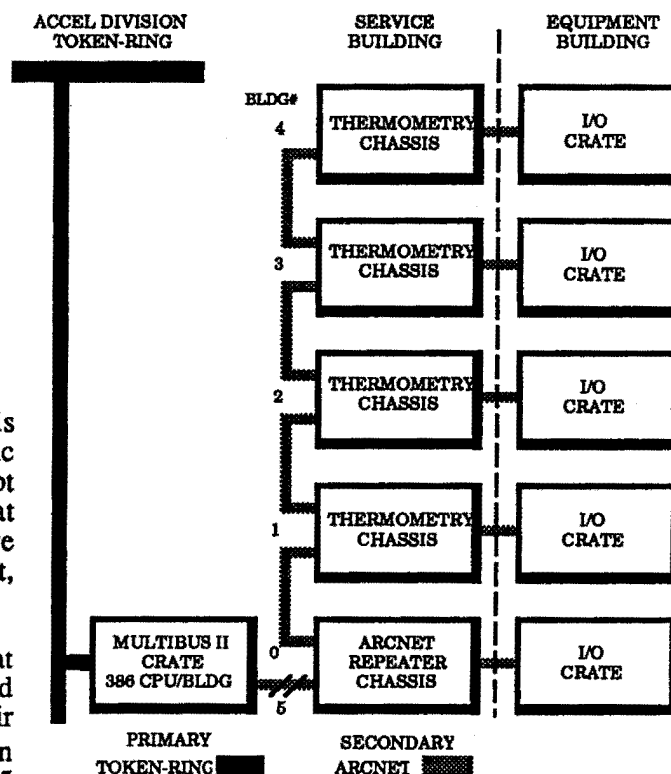
COMMUNICATIONS TOPOLOGY

FIGURE 2

The I/O subsystems

The basis for each of our subsystems is a 16 Mhz Intel 80C186EB functioning as an I/O processor. The "EB" is a low power, CMOS version of Intel's 80186 embedded microprocessor. It is based on a modular CPU core that integrates most essential system services onto a single 80 pin package. The object code of the 186 is directly compatible with the "real mode" of the 80386 microprocessor. Hence, a subset of the tasks developed for the execution processors could, in the future, be ported to the I/O processors as a way to further improve software performance.

The Device I/O subsystem

The Device I/O subsystem includes support for 128 transducer inputs, 38 actuator valves, 32 helium relief valves, 12 vacuum gauges and 6 "wet" or "dry" expansion engines. The subsystem is housed in a Euro-chassis located within the cryogenic equipment building. The current configuration is illustrated in Figure 3.
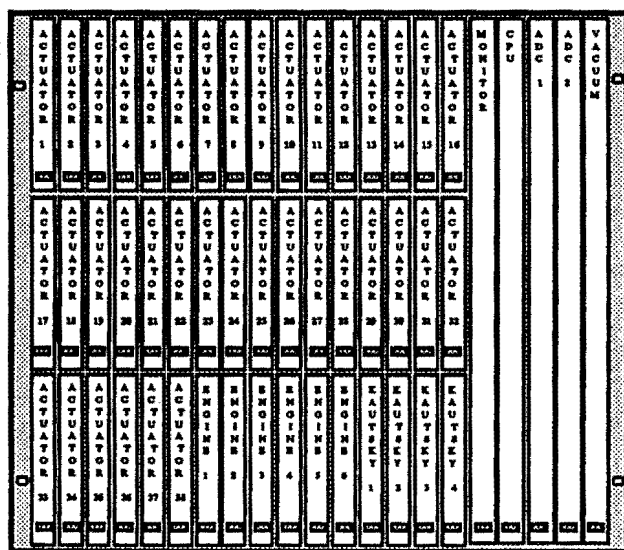
EQUIPMENT BULDING I/O CRATE

FIGURE 3

Activity within the subsystem is orchestrated by a 9U size processor module which hosts the 80186EB. In addition to those system services provided by the 80186, the processor module also provides fast data logging facilities, Tevatron clock decoding, the Arcnet interface, and distribution of various clock signals.

The I/O processor interfaces to the rest of the system via three identical backplane buses referred to as the "F2" bus. The F2 bus was developed in house to overcome the difficulties associated with cabling an average number of process I/O signals to a 3U size Eurocard while simultaneously supporting a high performance 16 bit bus interface on a single 96 pin DIN connector. It provides 32 pins of user defined I/O at each slot along with addressing space for up to 512 sixteen bit memory mapped registers. The individual slots are mapped to the I/O processor's memory and are individually selected in a fashion similar to the CAMAC standard.

Transducer interfacing is accommodated by two, 9U size A/D converter modules. Each module supports 64 differential input channels with a 10 volt range. The analog front end consists of two stages of input multiplexing, an instrumentation amplifier, and a 12 bit, self calibrating ADC. To facilitate fast data logging all 64 channels are digitized within a one millisecond period. The results are stored in a double banked, dual ported RAM. The arrangement allows the I/O processor fast access to the digitized data with a minimum of arbitration conflicts.

Actuator Control cards provide a comprehensive interface to the Barber–Coleman electromechanical valve actuators used throughout the cryogenic system. Each 3U size card incorporates all of the functionality necessary to interface a single device including: 24 volt DC drive control, Linear Variable Differential Transformer instrumentation, and A/D conversion circuitry. A local front panel control mode is also provided.

Wet engines, dry engines and cold compressors[1] all interface to the control system via Engine Control cards.[6] Each 3U card provides momentary relay contacts to implement "start", "stop" and "reset" commands. Also included are two independent channels of analog I/O (one for engine speed and one spare), and 16 optically coupled status bits. A front panel display reflects the state of both command and status bits.

The variety of solenoid valves used throughout the cryogenic system are interfaced by Kautzky[7] Control cards. Also implemented in the 3U format, each card provides control and status readback for up to eight solenoids. A front panel display reflects both the actual state and the requested state of each valve.

Pirani (vacuum) gauges are interfaced by a 9U size Vacuum card. The card supports 12 transducer circuits. A free running, multiplexed A/D converter continuously scans the transducer voltages, storing the digitized results in a dual ported RAM buffer.

Miscellaneous status bits are monitored by a Digital Input card. The 3U size card supports 30 bits of optically coupled status with input voltages ranging from 5 to 24 volts D.C.

The thermometry subsystem

The thermometry subsystem supports 96 channels of pulsed current, resistance thermometry. Unlike the Device I/O subsystem, it is implemented as an

application specific single board computer. It resides in a NEMA–12 enclosure mounted along the back wall of the Tevatron Service building.

The free running measurement scheme uses a precision current source (along with six stages of multiplexing) to deliver a 50 microsecond current pulse to each resistor module. The voltages developed by the resistors are scaled by a programmable gain instrumentation amplifier before being digitized by a 12 bit, self calibrating A/D converter. Here too, a dual ported RAM is arrangement is used to store the digitized results.

The subsystem also maintains a hard-wired 16 bit bi-directional data link with the Tevatron Quench Protection system (QPM). In the event of a magnet quench, the QPM provides the refrigeration control system with specific cell information to assist automatic recovery schemes. The refrigeration system provides the QPM with a permit allowing the magnets to be turned on.

Reliability and system diagnostics

One of the notable benefits of using the Multibus II architecture is the emphasis that the platform places on confidence testing and system diagnostics. Every module in the system supports a standardized set of "built in" self tests (BIST). During the initialization phase following a power–on reset, each module executes a subset of these routines to test its own functionality. The results are posted in "interconnect space" and are globally available to the rest of the system. Upon the coordinated completion of a reset, one module (in our case the PC16) assumes a temporary role as the systems' "boot master". The master can subsequently invoke, or download additional tests for each subsystem on an individual basis. This can be accomplished locally via an RS232 port or from a remote facility with the use of a modem.

Upon completion of confidence testing the operating system and refrigeration specific application software are down-loaded via the iPSB. At that time the execution processors are started and the PC16 relinquishes control of the system. The PC16 is now free to be utilized as a local information data base or console emulator by technical personnel. System schematics, diagnostic flowcharts and related maintenance histories can be displayed, as well as any other useful DOS based application.

Conclusion

The use of a hierarchical, networked architecture is ideal for this application. The higher cost, 32 bit microprocessors have been de-coupled from the cryogenic system, thereby protecting the investment from premature obsolescence. The distributed subsystems can be incrementally upgraded – or even replaced – with a minimum of impact to the rest of the system. Additionally, the maximum limit of 255 nodes per sector permits almost unlimited expansion of the system.

Acknowledgements

[1] Fuerst,J.D. "Trial Operation of Cold Compressors in Fermilab Satellite Refrigerators", Advances in Cryogenic Engineering, Vol. 35B, Plenum Press, New York (1990) .

[2] Zagel,J. et al., "Tevatron Satellite Refrigeration Control Subsystem",IEEE Transactions on Nuclear Science, Vol NS28 , p 2153-2154.

[3] Glass,M., et al., "The Upgraded Tevatron Front End"' NIM A, 1990, Vol 293, p.87-92.

[4] Glass,Brett; "The Return of Arcnet", BYTE, February 1991, p119.

[5] Chapman,L. et al., "Microcomputer Software Architecture for Cryogenic Controls", (This conference).

[6] Rode,C.H., "Tevatron Cryogenic System", 12th International Conference on High Energy Accelerators, 1983. p 529-535.

[7] "The Kautzky Valve", Fermilab Reports, June, 1982.

Controls for the CERN Large Hadron Collider (LHC)

K.H. Kissler, F. Perriollat, M. Rabany and G. Shering

CERN, CH-1211 Geneva 23

Abstract

CERN's planned large superconducting collider project presents several new challenges to the Control System. These are discussed along with current thinking as to how they can be met. The high field superconducting magnets are subject to "persistent currents" which will require real time measurements and control using a mathematical model on a 2-10 second time interval. This may be realised using direct links, multiplexed using TDM, between the field equipment and central servers. Quench control and avoidance will make new demands on speed of response, reliability and surveillance. The integration of large quantities of industrially controlled equipment will be important. Much of the controls will be in common with LEP so a seamless integration of LHC and LEP controls will be sought. A very large amount of new high-tech equipment will have to be tested, assembled and installed in the LEP tunnel in a short time. The manpower and cost constraints will be much tighter than previously. New approaches will have to be found to solve many of these problems, with the additional constraint of integrating them into an existing framework.

I. LHC REQUIREMENTS

The Large Hadron Collider (LHC) is the major project planned by CERN[1], and will be its largest and most expensive ever. It will present control problems much greater than those experienced in earlier accelerators.

LHC is a superconducting twin beam hadron (proton initially) collider providing 7.7 TeV per beam at 10 Tesla bending field. The novel twin bore magnets in their cryostats will be installed in the same 27 kilometre tunnel as the LEP machine. The **scale** of the control problem can be gauged in part from the 1792 dipole and 392 quadrupole cryostats filling most of the circumference, in part from the number, about 2000, of insertion and corrector magnets and appropriate beam instrumentation. The **difficulty** of the control problem will come from the sensitivity of the superconducting magnets to quenching under beam loss from the 4725 bunches of 10^{11} protons at 400.8MHz, making 851mA. This problem is exacerbated by the time varying persistent currents and the need for strong collision insertions to achieve the targeted luminosity of over 10^{34}. These requirements will strain dynamic aperture and magnetic field control to the very limit.

As LHC will be built in the same tunnel as LEP, a lot of equipment and controls will be common to the two machines. A major objective will therefore be a seamless integration of LHC and LEP controls. This will not be easy, in part due to the much more difficult control problems of LHC, in part due to the wide separation in time between the construction of the two systems compared to the speed of evolution of controls technology. A challenge for LHC will therefore be to permit the use of the latest and cheapest controls technology in such a way that it integrates with existing technology, allows experience and algorithms to be maintained, and does not demand a difficult and costly upgrade of existing systems. Another objective to be borne in mind is the aim of having a single control centre for the whole of CERN on the LHC time scale.

II. MAGNETIC FIELD CONTROL

This will be the most difficult control problem for LHC. The magnetic field is determined not only by the voltages and currents from the 1400 power supplies, but also by "persistent currents" in the superconductor which vary with time depending on the history of the magnetic cycle. Fortunately HERA experience has shown these effects to be reproducible, hence eventually calculable and correctable. Corrections will be derived from magnet measurements during the construction and from on-line measurements from reference magnets. The final trimming will have to be done using single pilot bunches of first 10^9 then 10^{11} protons. After full beam injection, continuous feedback control will be required, especially immediately after injection and during beam squeezing.

A. Modelling Server

The solution envisaged is to use a modelling server which will re-calculate the power supply settings in real-time, using measurements from the reference magnets, beam measurements, past history of the magnetic cycle, and the magnet characteristic data-base. The update time will be between 2 and 10 seconds. Tests on an Apollo DN10'000 in the LEP control system indicate that the computational load will not be beyond the sort of on-line computer we can expect in the LHC time scale. Each power supply will have a microprocessor capable of interpolating the required voltages and currents between modelling server updates.

B. Fast Communications

A new communications system is being studied for LHC, in conjunction with SSC, in order to acquire the beam data and set the power supplies at the required rate[2]. This will use TDM communications technology and reflective memory com-

puter cards to provide parallel transmission of data with no software protocol overhead. The power converters and beam monitors will use the latest digital signal processors and microprocessors, so no problem is envisaged at this level provided they can all operate in parallel.

C. Pilot Procedures.

It will be necessary to automate the procedures of machine preparation, pilot injection tests, and full beam injection and acceleration, in part to keep the persistent current effects under control, in part to avoid human error. Much work of this kind was done for the Antiproton Accumulator after initial commissioning, but for LHC this will have to be done beforehand, so placing an early load on the applications programming effort. Of course the modelling programs in the servers will also have to be ready and tested before injection tests can start. High quality work in these areas will greatly reduce the time needed to get the machine into proper working order.

III. QUENCH PROTECTION

The first line of quench protection will be assured by 24 quench protection stations distributed around the ring. They will constantly analyse the voltage transducers connected at each coil access point in order to determine if a quench has taken place. The quench station will then take a series of time critical actions including firing the beam dump and firing the magnet heaters to spread the quench evenly over the magnet. This forces the current to flow through the cold diode protection shunts to avoid local damage to the coil at the quench position. Helium pressure relief valves are then opened to avoid reaching emergency over pressure. A 5 second pre-quench record will be held for "post-mortem" analysis.

The magnet control computer will then take over, switching in series resistors in appropriate places to run the current down as quickly and safely as possible. Constant surveillance of the quench protection stations will of course be necessary as no malfunction can be permitted. Also the state of each magnet will have to be monitored for abnormalities in temperatures and voltage drops to detect deviations which might indicate variations in performance from the model, or incipient trouble. Before each injection extensive automatic checks will have to be done on all magnets, cryogenics and quench protection stations to ensure a fully normal situation.

All of this software will have to be ready and well tested before any significant number of magnets can be put into operation. For this reason the controls effort will liase closely with the magnet test strings and test stands right from the beginning.

IV. BEAM MONITORING

The two main subsystems are the orbit measurement and the beam loss monitors. The monitors will be wired to 24 concentrators round the ring for local processing and connection to the network. Each local unit will have a direct connection to the beam dump which will be activated if conditions which will lead to a quench are detected. In the case of some critical beam loss monitors, like those near the collimators, the dump will have to be activated in 1 or 2 revolutions, ie. 90 to 180 microseconds.

The closed orbit measurement will be systematically acquired by the modelling server to aid in its 2-10 second update of the power supply currents. In addition to helping compensate for the persistent currents, this will provide a continuous on-line correction of the orbit.

The orbit and beam loss monitors will also be acquired on a regular basis by the central alarm server to help in the detection and avoidance of quench provoking situations.

Other instrumentation is foreseen to measure tune, chromaticity, profile, and dynamic aperture and will also be controlled by local sub-systems. These measurements will be used along with the orbit and loss monitors by the programs which set up the machine for injection, acceleration, and collision. A particularly important role for all the instrumentation will be the pilot injection tests first with a bunch of 10^9 protons, then with a 10^{11} bunch to prepare for the full batch.

The beam monitoring stations will be connected by the fast acquisition system to a beam data server and hence to the modelling server. A first trial of the fast acquisition is planned in the near future using some of the LEP beam instrumentation as a prototype.

V. CRYOGENICS

The number of pieces of cryogenic equipment to be controlled is very large. However, the equipment involved is well known to industry, and an industrial control system will be purchased for this purpose. This must communicate effectively with the main control system, as any magnet whose cryogenic state or pre-history is inadequate may be prone to a quench before full field is reached. Thus the state of each magnet must be checked by the control system before each cycle and continuously monitored thereafter. Also access to the cryogenics systems should be the same for normal operator use as the access to any other system.

VI. OTHER SYSTEMS

At present it is assumed that most other LHC subsystems will be controlled by extensions of the LEP control system. If after further study it is found that some other system, for example the collimators, needs special fast control, they can

be dealt with using the fast techniques used for the beam monitors and power supplies.

At first LEP and LHC will not function at the same time and therefore much of the standard LEP controls may be re-used for LHC. In any upgrade of LEP this will be borne in mind and spare capacity will be provided where appropriate.

VII. LHC DATABASE

A database (ORACLE) will be used to store the characteristics of the LHC machine and to help with the planning and installation. Such a facility proved invaluable for LEP. For LHC there will be even more high technology equipment to be installed in the tunnel in a shorter time and with limited staff. A maximum amount of easy to use informatics must be provided to ease and control this installation.

All controls information will also be stored in a database. The use of an on-line database was pioneered at the PS on the original IBM 1800, followed by the data-module concept at the SPS. The usefulness of having cabling and installation data on-line was illustrated in the '70's with the SPS experimental area system[3]. For LEP it was necessary to copy the data to local files for use in the control system. We hope to avoid this in LHC and use direct on-line data-base access. Not only will this save work and additional software, but it will reduce the possibilities of errors and inconsistencies. Two approaches are envisaged. For interactive use the database can be interrogated as required by SQL commands during the course of the applications programs. Tests at the PS have shown that response times consistent with human operator interaction can now be achieved. Alternatively, applications programs which must run fast or which need a lot of data, can interrogate the database at start up time and store the data in RAM structures for immediate and fast use, as is done at Isolde[4]. This also protects the programs from inconsistent updates. For any updates to become effective, the program has to be stopped and restarted.

VIII. ARCHITECTURE

Controls technology sees major revolutions on something like a five year time cycle. As LHC completion is more than 5 years away it is premature to make final decisions on the architecture, even more premature for the detailed components to be used. Nevertheless, since this is a controls conference, an architecture is presented for discussion which illustrates much of the current thinking[7]. This is shown in figure 1.

A. Equipment Connection Policy

A major feature is the equipment connection. An important change in equipment connection policy and technology took place in LEP. The responsibility of the equipment layer of the controls was transferred to the groups responsible for the equipment, and a "thin" connection made to the control system. This policy will be continued and reinforced for LHC . The controls group will provide the hardware connection for data and timing, and together with the equipment group it will define a control protocol which will define the characteristics of the equipment to the control system. This policy has several impor-
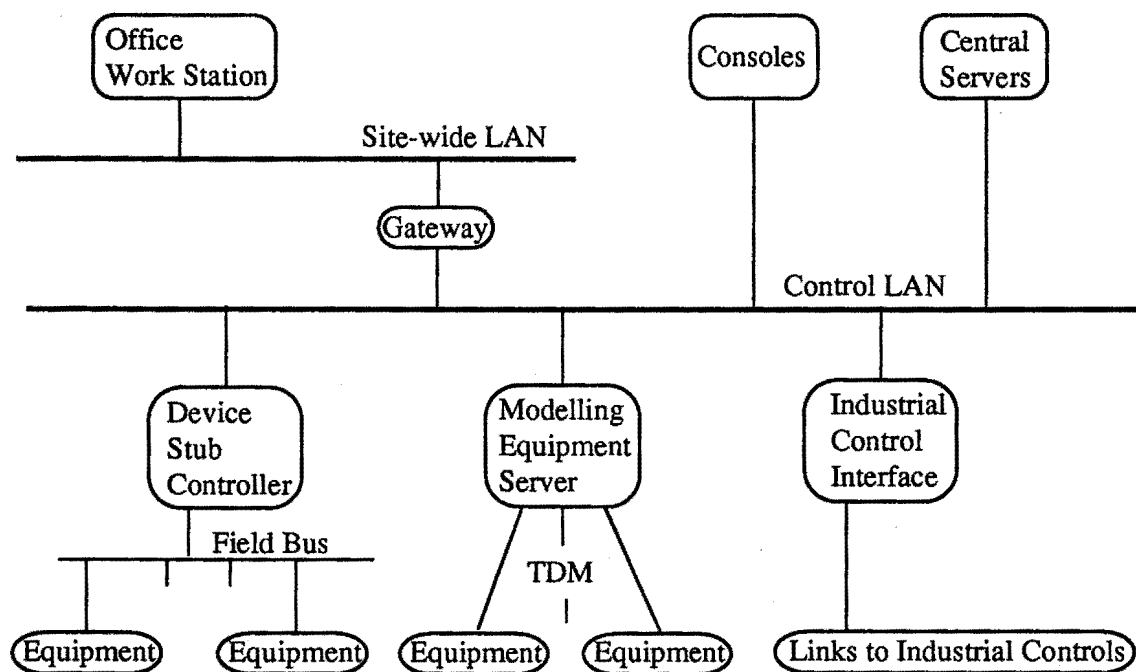

Figure 1. Schematic diagram of architecture

tant advantages. Firstly it clearly defines the budget separation between the control and equipment groups. Secondly it clearly delineates the responsibility for design, performance and maintenance of the control equipment. Thirdly it decouples the evolution of the control system from the evolution of the specific equipment controls. This is becoming an ever more important consideration as the park of control equipment increases and we have not the resources to change everything at the same time.

B. LEP Equipment Connection

The bottom left of Figure 1 shows the LEP equipment connection[5]. The specific equipment is connected to a field bus, MIL 1553, which is connected to a "Device Stub Controller" consisting of an Olivetti PC running the SCO UNIX operating system (replacement by LynxOS, a real time POSIX compliant operating system is planned in the near future. This operating system can also run in a VME crate). These PCs are located in the field, and are linked to the rest of the control system by the control LAN. Much of the LHC equipment will be in common with that of LEP. This includes, of course, all the tunnel management services. Some of the new equipment of LHC may also be connected to the MIL 1553 to save money and development cost. As LEP and LHC will not operate at the same time, economies can be made in sharing equipment and resources.

C. Fast Equipment Connection

The equipment connection shown in the centre bottom of figure 1 is a new development for LHC, being done in conjunction with the SSC[2]. The driving force for this is a fast connection to allow the rapid measurements and power converter updates required for control of the magnetic field. Other advantages of low cost and simplicity are also claimed for this new method, however, which may make it a contender for applications which do not need its speed. The basis of this method is to use Time Division Multiplexing (TDM) techniques to provide an individual channel from each equipment to a central server, all connections working in parallel with no software protocol overhead. This TDM technology is in commercial use for telephone and higher speed data-link requirements by the communications industry, mostly running on optical fibre. It will also be used in LHC for dedicated connections, telephone and video links, interlocks, and many other applications which are traditionally implemented using hard wired copper. Between the equipment and server, the TDM channel will link "reflective memory" cards in which part of the equipment memory containing the control protocol is reflected into the central equipment modelling server. Thus the modelling server can read its input, do its calculations, and write its output with no degradation in performance due to software protocol overheads and long transmission delays. Several servers may be used, eg separate ones for beam monitors, power converters and magnet monitoring. The TDM may be used to link these servers among themselves using reflective memory, or to link the equipment to several servers.

D. Industrial Equipment Connection

The third mode of equipment connection, shown diagrammatically on the bottom right of figure 1, is connection through industrial control equipment[6]. For the cryogenics equipment, and perhaps others, it makes sense to buy the controls from the equipment manufacturer or an industrial process control manufacturer as he has solved the problems, has all the components in house, and can take charge of the installation and maintenance. The controls thus purchased may stop at the level of the Programmable Logic Controller (PLC) for the equipment concerned, or even a complete system may be bought with the manufacturer's network system and consoles so that he can take full responsibility. In both cases an interface to this manufacturer's equipment has to be provided to achieve two way communication and control, and to integrate the industrial controls into the protocols and methods of the rest of the controls.

E. The Central Servers

In addition to the modelling server connected to the power supplies, there will be other central equipment servers connected by TDM to subsystems requiring fast response such as beam monitors, quench protection stations, collimator subsystem controls, dump, etc. Other central utility servers, as shown on the top right of figure 1 will be required for alarms, a file server, an on-line database engine, etc.

F. The Consoles

These will be standard large screen workstations. The PS and SPS console facilities are being upgraded to match LEP[7]. A considerable effort is being put into this and we can assume that the resulting facilities will be entirely adequate for LHC. There is a strong desire to have a single control centre with standardized facilities for the whole of CERN on the LHC time scale.

G. The Office Workstations

LHC will involve a large fraction of CERN's accelerator community over the construction and commissioning period. These people will be spread over a large geographical area. To permit them to work without too much travel to the central control room, access will be provided from the office workstations connected to the site wide LAN. A gateway will be used to authenticate access permissions, and to cut off access at critical times.

H. The Control LAN

A single LAN is shown diagrammatically in figure 1. The present system uses both Ethernet and Token Ring in a bridged configuration. It is reasonable to expect the use of FDDI in the LHC time scale.

S03SRD01

IX. PROTECTION

LHC will have severe requirements concerning the authorization of actions through the computer system. In situations where a quench is possible, equipment changes must be restricted not only to authorized persons but also to authorized programs which have been vetted to perform adequate checks for quench avoidance.

There will be a large number of access points to the system for testing, maintenance and commissioning including, as mentioned above, the office workstations. These numbers will aggravate the protection problem. Network management will have to provide access permission or denial facilities, as now being added to LEP, and the alarm and surveillance system will have to detect and locate abnormal access attempts. The protection will have to be closely linked to the machine operational state. Widespread access will be needed to speed installation, testing and repairs, moving swiftly to a closely controlled situation when quenching becomes possible.

X. TIMING

In addition to data transfer, many types of equipment need special timing signals and events. General purpose timing systems have been developed for PS and SPS from different historical backgrounds. Currently work is progressing on new designs which will cover both areas and which will take into account LHC requirements. Thus the timing distribution for LHC should be part of a new overall system for all CERN accelerators. There is a hope that the timing information may even be combined with the data stream so reducing the number of cables, connectors, and chips, so reducing cost and increasing reliability.

XI. APPLICATIONS PROGRAMS

The maxim "a control system is only as good as its application software" is as true now as ever it was. In the '60s the term "software barrier" was coined to express the difficulties in achieving good applications. In the '70s a determined effort using NODAL in the SPS, PS, PETRA and TRISTAN achieved a good measure of success. In the '80s the problem re-appeared with more complex systems and higher demands. For LHC a serious effort will be made to put together a team of machine physicists and programmers who will ensure that the applications programs will be available, not only to run the machine but also to make a positive contribution to a fast, smooth and safe commissioning.

For LHC, applications software will be required on two levels, at the server level, and at the console level. In the servers sophisticated software will be required for the modelling and real time closed loop control of the high intensity beam in the superconducting magnets. High reliability will be required in this software, and in the quench protection and magnet surveillance programs. All changes and machine settings will have to be verified through a model before being applied so as to avoid quenches.

In the consoles a wide range of software will be required. Automated injection procedures will be needed for speed and reliability. These programs and the server software will have to be ready before commissioning, unlike the cut and try approach which could be used with earlier less sensitive accelerators. Extensive software for beam measurements and diagnostics will be required.

Surveillance and alarm programs will be particularly important to avoid quenches and warn of quench provoking situations. A variant of this software will be required to check out the machine before injection, or before any action which might result in a quench.

Some of this software will have to be professionally written, installed and tested, especially in the servers. Other parts will be better written by the machine physicists and hardware specialists themselves. All will have to be carefully specified beforehand. Tools will include professional development facilities for compiled programs, the old tried and tested NODAL[8], mass market packages such as spreadsheets linked to the machine variables as used at Berkely[9] and in Isolde[4], and other commercial synoptic packages for control as used in LEP.

XII. REFERENCES

[1] Design Study of the Large Hadron Collider (LHC), CERN 91-03.

[2] C.R.C.B. Parker, M.E. Allen, H.C. Lue and C.G. Saltmarsh, A New Data Communications Strategy for Accelerator Control Systems, Nuclear Instruments and Methods in Physics Research, 1990 (Proceedings of Vancouver Controls Conference, 1989).

[3] H.W. Atherton, M. Rabany and R. Saban, A database as a bridge between hardware and software, Proceedings of the 5th International IASTED Symposium, Tunis 1982

[4] R. Billinge, I. Deloose, A. Pace and G. Shering, A PC Based Control system for the CERN ISOLDE Separators, Proceedings of this conference.

[5] P.G. Innocenti, The LEP Control System, Nuclear Instruments and Methods in Physics Research, 1990 (Proceedings of Vancouver Controls Conference, 1989).

[6] M. Rabany, Interfacing Industrial Process Control Systems to LEP/LHC, Proceedings of this conference

[7] R. Rausch and C. Serre, Common Control System for the CERN PS and SPS accelerators, Proceedings of this conference.

[8] M.C. Crowley-Milling and G. Shering, The NODAL system for the SPS, CERN 78-07.

[9] S. Magyary, M. Chin, C. Cork, M. Fahmie, H. Lancaster, P. Molinari, A. Ritchie, A. Robb, C. Timossi and J. Young, The Advanced Light Source, Nuclear Instruments and Methods in Physics Research, A293, 1990.

A Performance Requirements Analysis of the SSC Control System

S.M. Hunt, K. Low
SSC Laboratory *
2550 Beckleymeade Ave.
Dallas, Texas 75237

Abstract

This paper presents the results of analysis of the performance requirements of the Superconducting Super Collider Control System. We quantify the performance requirements of the system in terms of response time, throughput and reliability. We then examine the effect of distance and traffic patterns on control system performance and examine how these factors influence the implementation of the control network architecture and compare the proposed system against those criteria.

I. INTRODUCTION

The Superconducting Super Collider Laboratory (SSCL) is a complex of accelerators being built in the area of Waxahachie, Texas. It will be fully operational at the end of the decade. The SSCL consists of six accelerators: a 1 Gev Linac, three booster synchrotrons (the 12 Gev low energy booster, a 200 Gev medium energy booster, a 2 Tev high energy booster) and two intersecting, contra-rotating 20 Tev synchrotrons that make up the Collider itself. The complex will occupy approximately 112 km of underground tunnels. There are estimated to be about 150,000 control points requiring remote control and interrogation in order to operate the accelerator and diagnose its condition.

II. PERFORMANCE REQUIREMENTS

A. Response time

The control system's response to operator requests should be such that response delays be unnoticeable to the operators. The minimum response time of the control system should be, in the absence of any other constraining factors, 20Hz.
In addition, it will be necessary to provide for higher rates, up to 1 Khz for some essential services like the quench protection monitors (QPM).

B. Throughput

In the Collider tunnel there are 5 Superconducting dipole magnets per half-cell and 968 half-cells per ring. Every 450m is an equipment niche (alcove) which controls 5 half-cells (200 niches). The HEB has 280 half-cells controlled by 24 Niches. The MEB consists of 200 half-cells controlled from 8 surface buildings and the LEB has 108 half-cells controlled from 6 surface buildings. Throughput

requirements vary widely [Table 1]. The Linac is not considered here. The environmental (ENV) figures include niche temperature, power, smoke alarms, oxygen and water. The value in the column marked Locations indicates the number of Niches or equipment buildings for a particular machine. The value in the column marked Bytes indicates the number of bytes of raw data being generated at that location for each time interval indicated in the column marked Rate, which is the number of time intervals per second. The value in the column marked Bandwidth is the total number of bytes per second generated for that equipment type and is the product of the Locations, Bytes and Rate values. For the LEB BPM data rates have been set in this table at one tenth of the raw rate in order to conserve bandwidth.

The total amount of data generated site-wide by the SSCL is in excess of 250 Mbytes per second (2 Giga bits per second)[1,2].

C. Reliability

Total allowable unscheduled downtime for the control system is 30 hours in 4505 hours of operation per year. The control system will consist of 205 equipment locations consisting of 162 Collider niches, 24 HEB niches, 11 MEB buildings, 6 LEB buildings, the Linac and the control room complex. Each of these locations will have one communications element (Hub Gateway or multiplexor, depending on the communications architecture chosen) and up to 9 equipment crates.
If each of these 2050 elements (205 locations x 10 elements) is a "critical" system, then to achieve 30 hours of unscheduled downtime with a mean time to repair of 1.5 hours (20 incidents per year) each element would have to achieve a mean time between failure of 54 years [3]. It is therefore clear that other measures, such as the use of redundant systems, will be necessary in order to achieve the necessary reliability figures.

D. Capacity

The installed system should have a capacity at least 50% greater than the requirements stated above. It should furthermore be capable of being expanded by 400% without incurring any additional civil engineering costs or replacement of existing components, only expansion costs.

III. OPERATIONAL REQUIREMENTS

A. Data Accessibility

Table 1. Throughput and response time requirements

System	Locations	Bytes	Rate	Bandwidth
LEB				
BPM	6	108	50,000	32,400,000
RF	1	2,125	20	42,500
RAMPS	6	432	720	1,866,240
VACUUM	6	288	20	34,560
ENV	6	22	20	2640
MEB				
BPM	8	150	6,600	79,200,000
RF	1	2,125	20	42,500
RAMPS	8	600	720	3,456,000
VACUUM	8	400	20	64,000
ENV	8	22	20	3,520
HEB				
BPM	24	70	24,000	40,320,000
RF	1	2,125	20	42,500
QPM	24	60	720	1,036,800
CRYO	24	326	20	156,480
RAMPS	24	280	720	4,838,400
VACUUM	24	187	20	89,760
ENV	24	22	20	10,560
COLLIDER				
BPM	200	58	3000	34,800,000
QPM	200	60	720	8,640,000
CRYO	200	135	20	540,000
RF	1	1062	20	21,240
RAMPS	200	332	720	47,808,000
VACUUM	200	155	20	620,000
ENV	200	22	20	88,000
TOTAL				256,123,700

Control loops will be implemented at the Local (Niche), Regional (Sector) and Global (Control room) levels[4]. However, in order to debug the systems it will be necessary to open some of the local and regional loops from the control room, and move some of the loops from local to regional to control room and vice versa. For instance a control room algorithm might be tested at the global level and then installed as a local loop for reasons of security, because it may need to continue to operate when the global system is in a maintenance mode.

This leads to a requirement that all raw data that might be needed at any level of the control system, even local loops, must be available in the control room. Furthermore this has the important advantage that application programs will have access to all of the data associated with the sensors that might otherwise be hidden. For example if a beam monitor system provided only the result of a calculation, for instance the beam tune, it would not be a simple matter to add another capability, for instance a calculation of the beam lifetime. More importantly new algorithms could not be tested without affecting the operation of existing systems. In addition, as new capabilities are added to systems, the control system should not limit the raw data that is acquired to some arbitrary fraction of the total. All data should be available at all levels.

B. Traffic Patterns

As has been stated earlier, data generated at equipment locations (niches and buildings) should be available simultaneously at more than one location, for instance at the regional level and at the control room to be consumed by console applications. It is not anticipated that there will be significant Niche to Niche communications. This must not however be excluded. Although a number of computers must be able to read the data from equipment locations, only one should be able to write commands to field equipment at any given time. An arbitration mechanism to give permissions to access equipment for commands will be necessary. When command messages are sent to an equipment location it may be necessary to queue the messages for processing. This can be achieved by use of a high level communications protocol such as TCP/IP, but this introduces a large overhead for data transmission of up to 50%. It might be more efficient to queue the requests to the arbitration mechanism that allows commands to be sent to the equipment.

C. Determinism

Application programs typically periodically request current readings and settings from field sensors. It will be necessary to quantify the rate of such requests and guarantee the timely transport of data through the communications systems. The performance of the system must be predictable, that is deterministic. Furthermore because the system will be designed to worst case scenarios, no advantage would be gained by trying to achieve best case performance better than worst case. Thus data transport should be load-independent.

D. System Software

This predictability in the control system should extend to computer operating systems as well as communications equipment. This may mandate the use of real-time operating systems. This would be true for embeded systems using, perhaps, real-time kernels and also for the computers in which run application programs. For these therefore operating systems such as UNIX would be discarded in favour of, for instance, a real-time UNIX.

E. Data Stores

It will be necessary to provide a mechanism to store any data that is acquired to equipment locations or applications programs. This data storage might be for temporary use in shared memory, in a database or a system. As some of this data may be archived for many years it is important

that it be stored in a format that does not become obsolete with new versions of the storage utilities. There should be one coherent set of library routines and utilities that can handle these functions. These utilities and library routines must be able to access the structure of the data, not just handle the data set as a whole. The ability to access the data should not depend on the availability of header files that were written at the time with the original application program, and may now be out of date. And they must not need prior knowledge of the structure of the data. The description of the data should therefore be in a standard format, embedded in the data or in a database.

IV. POSSIBLE COMMUNICATIONS ARCHITECTURES

A. SERIAL CAMAC HIGHWAY

Serial Camac meets many of the requirements of a controls communications architecture, all the data is available from a global level, with no data-hiding, but it is difficult to control access to the equipment. Response times and throughput values using present systems are below the level of performance that we require.

B. CSMA/CD

IEEE Standard 802.3 is a "listen while transmitting" LAN access technique, commonly referred to as CSMA CD (carrier sense multiple access collision detection). It closely resembles Ethernet with some minor changes in packet structure together with an expanded set of physical layer options. Bit rates supported are between 1 Mbps and 20 Mbps. Collision detection means that collisions i.e. simultaneous bids for access to the medium can be detected early in the transmission period and aborted, thus saving channel time and improving overall channel utilization. When a collision is detected, the user backs off and continues to attempt access until a maximum number of unsuccessful attempts has been reached before generating an error. A contention-based protocol like Standard 802.3 is unsuitable for a number of reasons. It is not deterministic. It can be totally blocked with no critical data able to get through from a critical system. It is not easy, using standard protocols, to arrange that many stations be able to receive the data, and the data producer has to know the address of the recipient. The access protocol works only for short segments of network, requiring bridges between physical networks. Typically, networks which are lightly loaded with random traffic requests are especially suited for CSMA/CD schemes.

C. Token Rings

The token ring protocol specified by IEEE Standard 802.5 is a polling-based controlled LAN access technique. A station gains the right to transmit on the ring when it detects and subsequently captures the circulating token. This station continues to transmit until it either exhausts all transmission frames or the token times out. The station relinquishes monopoly of the ring by generating a new token which other stations may then acquire. The timing-out mechanism ensures that other stations on the ring have a chance to transmit.

Token passing protocols can be made to be deterministic, but many of the higher level protocols do not take advantage of this feature. Token rings transmission rates can only go as high as 4 or 16 Mbps, meaning that many rings would be needed to achieve the bandwidth requirements of the SSCL. Also, response time is slow due to delays introduced by each station. It is more suited to larger transfers such as file transfers.

D. FDDI

The fiber distributed data interface has the advantages of the token passing protocol and is much faster (100 Mbps). It can operate over the distances covered by the SSCL.

The delay introduced per station is much lower than for token ring as it does not capture all of a packet before retransmitting it, but captures the token while at the same time retransmitting it to the next station on the ring. If the station is the intended destination, determined by examining the first bytes of the token, the outgoing transmission is aborted and this invalid packet is stripped off of the ring by the station that originated it.

However the use of only a single token on the ring at any time means that for large rings such as at the SSCL, when transmitting small amounts of data, most of the time is wasted transmitting the tokens.

E. TDM

Time Division Multiplexing (TDM) is widely used in the Telecommunications industry. It is the method used for passing voice and data channels over common copper and fiber optic media. The system consists of a network in which a channel that is a multiple of 64 Kbps or 1.544 Mbps is assigned between two geographical locations [5]. There may be many channels between these two locations, each channel carrying specific information with all the channels sharing the same fiber media.

The TDM system with its established industry standards of supported transmission rates will be able to address our requirements as outlined below.

V. TDM PERFORMANCE

A. Response Time

The equipment employed has very low overhead, typically 10 μs per node which is less than the speed of light delay imposed by the distance around the Collider. Since it is a point to point network and not a ring, the time to transmit a message is halved as the message does not have to return to its source.

B. Throughput

Throughput is determined by the number of channels assigned to any link. Standards exist for TDM equipment at a number of data rates. Low speed systems use asynchronous transport at (for example) 1.544 Mbps (called T1 or DS1) and 45 Mbps(called DS3).

At higher rates, synchronous systems often based on fiber optic technology are available. These are defined in the Synchronous Optical NETwork (Sonet) standard. This standard specifies rates that are defined at 55 Mbps (called OC1) which can transport 28 DS1 signals and multiples of that rate. The rates are not exact multiples as some overhead information passed for network management which uses the same number of bits regardless of the link speed. OC3 for example can handle 84 T1 signals and runs at 155 Mbps. Commercial equipment available now is capable of transporting 2.5 Gbps (OC48) over a single fiber optic cable.

Work is continuing with multiplexors at OC192 that should be commercially available in a few years, but additional fibers can always be utilized for increased bandwidth. Standard fiber optic cables consist of up to 250 fibers in a 3/4" diameter cable, giving present total capacity of 625 Gbps.

C. Reliability

This type of equipment is used throughout the commercial telephone system where an interruption of service could be disastrous. As this equipment is (as in our case) installed in remote locations, many network management and diagnostic features are built in.

Redundancy is an important feature in these systems. Dual redundant optics and power supplies, and the possibility of building redundant ring networks, are standard features. The use of redundant rings affords network integrity. Data is automatically rerouted the opposite way round the network should the fiber break or otherwise fail.

D. Data Accessibility

As each data channel is independent of all other data channels, there can be no contention in the network. Data from embedded systems or regional computers can be made globally available as the network will have the capacity to handle all data.

VI. TDM Implementation

A. Interface to front end equipment

In each of the equipment locations, for instance Collider niches [Fig. 2], a number of different systems have to be interfaced to the TDM communications system. These include beam monitors, quench protection, ramp generation, vacuum and cryogenics.

These systems will use VXI, VME, STD or CAMAC bus standards or may in some cases be acquisition and control interface cards directly connected to the TDM network. Each of these systems would normally have a 64 Kbps interface to the TDM network. Where a higher interface rate

is needed, it would be a multiple of 64 Kbps or a full T1 interface at 1.544 Mbps (24 individual 64 Kbps channels).
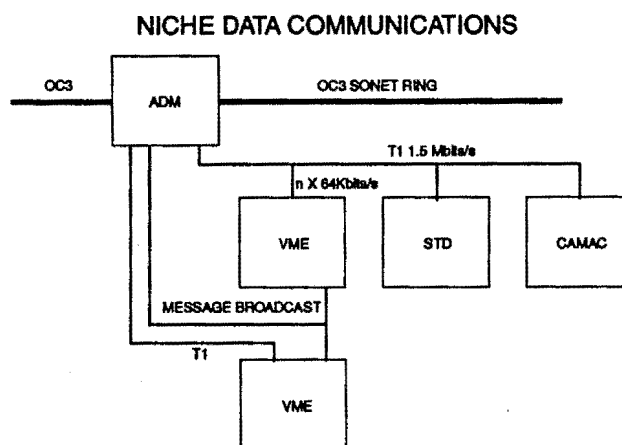
Figure 2: Collider niche data communications

B. Message broadcast system

Some systems also have an interface to a message broadcast system[6]. This uses a T1 interface to distribute, from the control room, medium speed synchronization signals (called events). These signals are characterized by being repetitive such as the 720 Hz clock event or signals that are needed in many locations such as an injection warning event.

C. Long distance links

At each equipment location, the low speed T1 signals are multiplexed using an add-drop multiplexor (ADM) onto a high speed OC3 Sonet link [Fig. 1]. This link will connect all equipment locations in a region. A region would be a Collider or HEB sector (the Collider has 10 sectors, the HEB 2), the Linac, LEB or the MEB. At one location in each region the OC3 link will interface to a global OC48 (2.5 Gbps) Sonet link. This will be a Sonet ring that will connect together all the regions and the control room.

D. Interface to regional computers

The regional computer would also be interfaced to the regional OC3 link. This is to allow it to control regional control loops if necessary. Data arriving from equipment locations would be available to the regional computer as well as transported to the central control room via the OC48 link.

E. Interface to Functional computers

In the control room functional computers running accelerator applications programs will need to have access to data arriving from the equipment locations over the Sonet links. The physical attachment to the TDM networks will be from VME-based Sonet interfaces running

SSC CONTROLS ARCHITECTURE
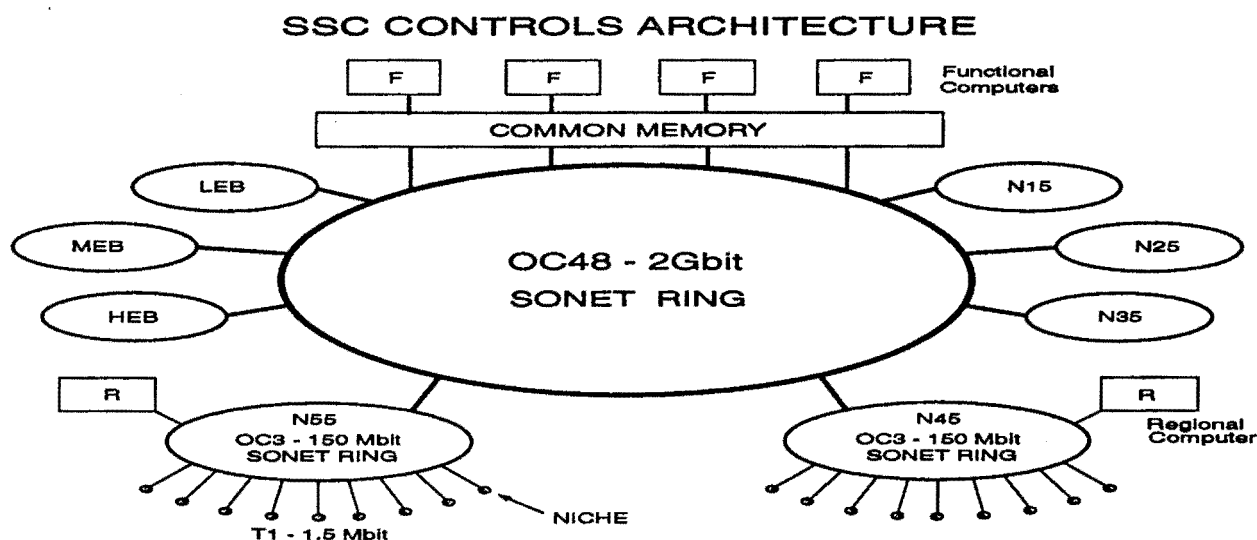
Figure 1: SSC Controls Architecture

at OC3. These OC3 signals will be obtained from an add-drop multiplexor on the OC48 ring. Each functional computer will have access to the data, not only from the OC3 to which it is directly connected but also from all other functional computers. The data arriving will be memory mapped into the virtual address space of all of the functional computers.

VII. CONCLUSIONS

The SSCL performance requirements appear to be attainable with today's technology. Furthermore, the communications network will be largely commercial, thus meeting the reliability and inevitable future capacity upgrades. TDM technology is well-understood and well-established and would not become obsolete during the lifetime of the project.

VIII. ACKNOWLEDGMENTS

The choice of implementation using parallel communication and memory mapping was influenced by the control system of the Advanced Light Source at Lawrence Berkeley Laboratory[7]. Many of the original concepts were proposed by C. Saltmarsh (SSCL) and C.R.C.B. Parker (CERN) who is pursuing some of these ideas for use in the LHC control system at CERN.

References

[1] Site Specific Conceptual Design, SSCL-1056

[2] M.E. Allen, C.G. Saltmarsh, S.G. Peggs, D.A. Dohan, and V.E. Paxson, "Demands Placed on the SSC Laboratory Control System by the Engineering and Operational Requirements of the Accelerators," International Conference on Accelerator and Large Experimental Physics Controls Systems, 1989, Vancouver, Canada.

[3] K. T. Dixon and J. Franciscovich, "SSC Accelerator Availability Allocation", 1991 International Industrial Symposium on the Super Collider, Atlanta, Georgia.

[4] M. Allen, S.M. Hunt, H. Lue and C.G. Saltmarsh, "A High Performance Architecture for Accelerator Controls" , 1991 International Industrial Symposium on the Super Collider, Atlanta, Georgia.

[5] ANSI T1.102,"DS1, DS1C, DS2 and DS3 Levels of Digital Hierachy."

[6] K. Low and R. Skegg, "The Prototype Message Broadcast System for the SSC", IEEE Transactions on Nuclear Science vol. 38, No. 2, pp325-328, April 1991.

[7] S. Magyary et al, "The Advanced Light Source Control System ", Nuclear Instruments and Methods in Physics Research, A293(1990) 36-43, North Holland.

The Computer Control System for the CESR B Factory

C. R. Strohman, D. H. Rice and S. B. Peck
Laboratory of Nuclear Studies
Cornell University
Ithaca, New York 14853[*]

Abstract

B factories present unique requirements for controls and instrumentation systems. High reliability is critical to achieving the integrated luminosity goals. The CESR-B upgrade at Cornell University will have a control system based on the architecture of the successful CESR control system, which uses a centralized database/message routing system in a multi-ported memory, and VAXstations for all high-level control functions. The implementation of this architecture will address the deficiencies in the current implementation while providing the required performance and reliability.

I. INTRODUCTION

CESR-B is an upgrade to the existing CESR facility.[1] The major part of the upgrade is the addition of a second storage ring in the existing tunnel. The two rings will operate with asymmetric energies (3.5 GeV and 8 GeV) and will intersect within the CLEO detector. The design luminosity is $3\times10^{33}cm^{-2}s^{-1}$ which will be achieved with 230 bunches in each ring.

The control system for CESR-B is also an upgrade of the existing control system.[2] The architecture is shown in figure 1. The MPM (Multi-Port Memory) contains the database and is accessible by the high-level computers and the BCCs (Bus Control Computers). The high-level computers are used to develop and run programs which interface with the operators and physicists to control and monitor the experiment. The BCCs manipulate and move data between the database and the accelerator hardware. Both the MPM and the interface hardware are mapped into the memory space of the BCCs. They only transfer data when requested to by the high-level computers.

It is important to remember the difference between the architecture of a system and how it is implemented. Within a well defined architecture, one can make hardware or software changes to improve some aspect of performance without affecting systems which are outside of the boundary of the control system.

II. DESIGN PROCEDURE

Having decided that the current CESR control system is a suitable model for the B factory, we proceeded to analyze the strengths and weaknesses of our system and the different needs of the new system. Part of this process was defining the scope of the control system.

A. Boundaries of the Control System

For a large design project, well defined boundaries are essential. At the boundaries, the needs of other people must be taken into consideration, and the design process requires communication between the designers and the users of the control system. Within the boundaries, the control system designers can do whatever is needed. We have defined the scope of the control system by defining interfaces for application programmers, instrumentation designers, and operators.

Application programmers must be provided a complete, well documented, set of functions which meet their needs. Programmers are not allowed to bypass these functions by using calls to lower-level routines. CESR uses approximately 35 functions.

Designers of instrumentation hardware are provided with a specification for constructing interfaces to the control system. This includes mechanical, electrical, and protocol details. Recommendations that simplify the control system are included, but not required. This encompasses things like avoiding write-only registers and not having read operations change the state of the system.

The actual implementation of the operator interface is a combination of efforts by both the application programmers and the instrumentation designers. However, it is essential to know the needs of the operators when designing the control system.

B. Special Requirements of the B Factory

We need to know what makes CESR-B different from CESR and how these differences affect the architecture and implementation of the control system.

The first question is how much larger will the new system be? At this early date, we do not have all of the details from the various design groups (eg. vacuum systems, magnet systems), but we do have general numbers. Combining this information with the fact that the amount of equipment in the tunnel will approximately double, we determined that the new control system will have roughly twice as many output control points as the current system.
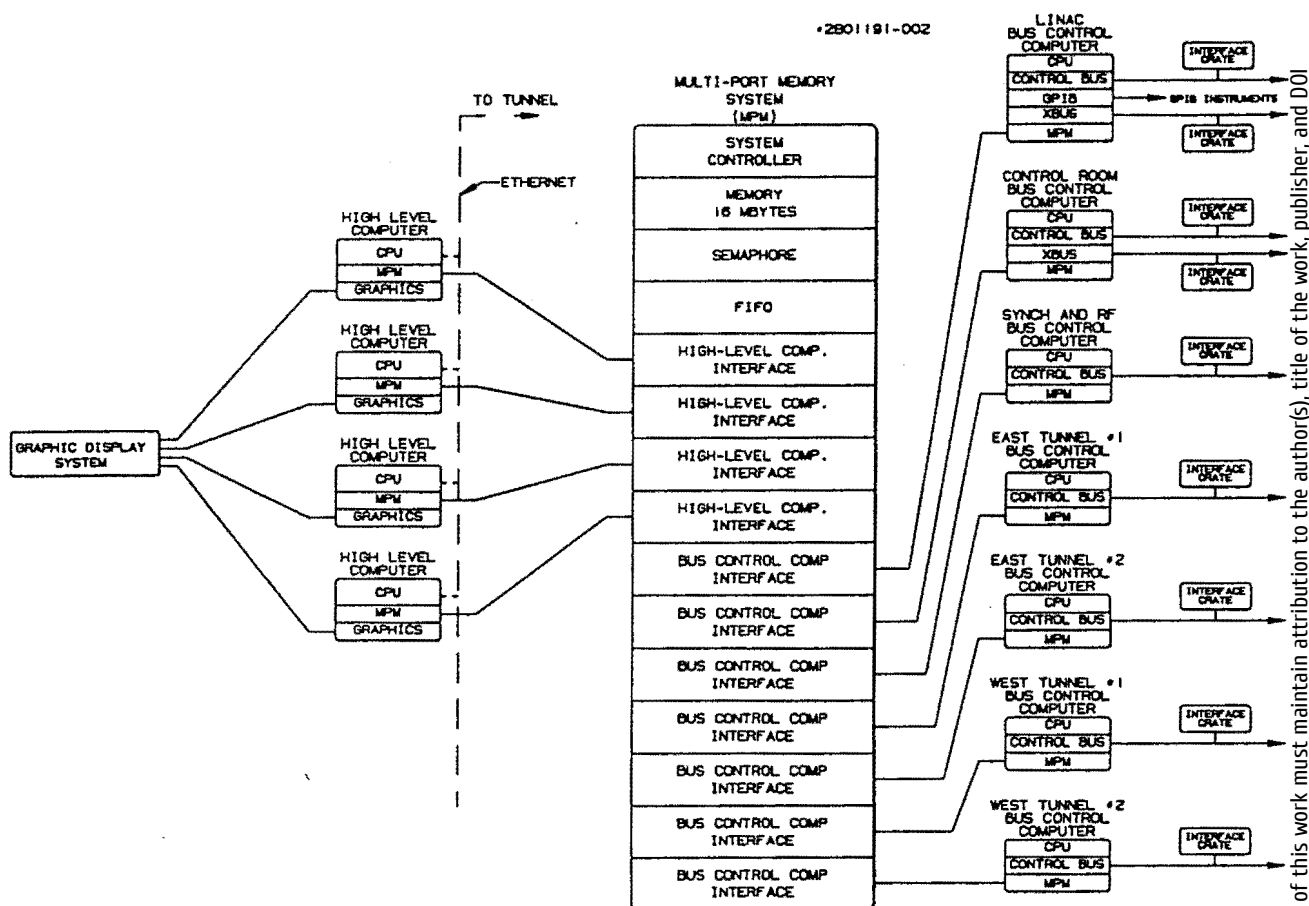
Figure 1. Control System Schematic

We are planning to monitor a great deal more than we currently do. This will help minimize downtime either by indicating when a problem is developing or pointing to the correct area when a malfunction occurs. With the increased monitoring and the additional equipment, we are assuming about five times as many input control points.

There will be several instrumentations systems which require local processors. These systems will need a control system interface for passing processed data. They will also need a connection for downloading and debugging.

C. Strengths of the Existing CESR Control System

Performance is a critical issue. When the operator turns a knob, there should not be a noticeable lag in the response. We run the CONSOLE program at 10 Hz. This is the program that accepts input from the operators and displays results to them. Each time this program runs, it scans the operator input devices, updates the controlled device, and updates the operator's display. It takes about 7 milli-seconds for each pass through this program.

The CESR control system makes very efficient use of the high-level computers. We currently use two VAXstation 3200 computers. The normal application load, consisting of 14

programs, uses 50% of one computer and 30% of the other. This allows special applications (orbit measurements, energy changes) to run in a timely fashion. The efficiency is achieved by minimizing the layers of function calls required to communicate with the hardware and by having processes hibernate when they have requested that the BCCs move a lot of data.

The system is simple and easily extended. We make minimal use of operating system and network functions. This allows us to avoid 'black boxes'; pieces of software over which we have no control. When we added a SUN computer for beam dynamics studies, it was trivial to move the subroutines that are provided to the application programmers.

There is a single database. This allows us to avoid the programs and overhead which would be needed to maintain the consistency of a distributed database.

The database and the interface hardware are memory mapped into the address space of the BCCs. Simple 'move' instructions are used to access the database and the hardware. The control bus can complete a data transfer to the farthest interface crate in less than 15 μsec. Database accesses typically take less than 1 μsec.

Several BCCs can work in parallel on a given transfer request. For example, when reading magnet currents two computers are moving data, one for the east half of the ring

and the other for the west half. Three computers control the operator interface.

The graphics display system provides fast and efficient access to many graphics screens, both color and monochrome. It is accessable to all of the high-level computers. Data is sent to it through a FIFO, so the high-level computers simply send data when they have it available.

Geographical addressing is used in the interface crates. Technicians do not need to set address switches on each card. We also insert and remove cards with the power on.

In the tunnel, the entire control system is contained with a metallic exoskeleton for shielding. Feedthroughs are used for connections to the controlled equipment. We are expecting the electrical environment to be more severe in CESR-B.

D. Limitations of the Existing System

The CESR control system has worked extremely well for over a decade. However, it does have limitations, most of which stem from design decisions which were appropriate in the late 1970s.

The address space on the control bus, which connects the interface crates in the tunnel to the bus control computers, is too small. Each control bus has an address space of 65 kwords. This is divided between 16 interface crates, with 16 slots per crate, yielding only 256 addresses per card slot.

We do not have a simple local extension of the control bus. We need to allow designers to build control system interfaces into their equipment and have an easy way to connect to the control bus. Our current system requires two bus operations with a delay of 20 µsec between each operation. A protocol similar to MIL-STD-1553 running in the 5-10 Mhz range would be useful. The length of these extensions would be less than 15 meters.

The interface crates were designed and built by us in the late 1970s. The backplane uses a byte-wide multiplexed protocol on which we transfer one address byte and two data bytes. Since it is a non-standard bus, there is no way to use commercial circuit boards. Producing more crates is very expensive and labor intensive.

There are no facilities for connecting a terminal or a computer in the tunnel. This makes testing and troubleshooting very difficult. We either walk back and forth to the nearest terminal or use the building public address system to communicate with someone at a computer.

The interface from the VAX computers to the MPM is relatively slow. This is due to the fact that the VAX does not have the ability to directly map the entire address space of the MPM. The system uses a set of address and data registers which are located in the Qbus I/O space. To access the MPM, one must first load the desired address into the address register, then transfer the data by reading from or writing to the data register. Since the Qbus has only a 16 bit data path, four Qbus operations are required for each MPM operation. An MPM access requires 12 µsec.

III. HARDWARE

A. High-Level Computers

CESR-B will probably use VAX computers for the major high-level functions. The features of the VAX are that they provide a reasonable development environment, they support priority scheduling of processes, and we are very familiar with them. We have shown that the control system operates comfortably on a VAX, and we expect that the heavier demands of CESR-B can be met by the newer generation VAXes.

As VAX performance improves, the inability to make a memory-mapped interface to the MPM becomes more of a bottle-neck. There is a two-stage plan to address this. First, we will make a 32 bit interface so that setting up the address register and moving the data becomes two operations, instead of four. If even more performance is needed, we will copy the read-only portion of the database into the VAX memory. The VAXes will be clustered to facilitate disk sharing.

There may be other types of high-level computers for special functions. Any computer that we can plug circuit boards into can be interfaced to the control system.

B. Multi-Port Memory System

The multi-port memory (MPM) will most likely be a VMEbus based system, although the Futurebus and any other contenders will be investigated. The MPM contains the RAM used for the database and for message passing. It also contains special hardware which is used to enhance the multi-processor aspects of the system. It can support up to 16 interfaces to high-level or bus control computers.

The system controller in the MPM is supposed to guarantee that under normal conditions no processor has to wait more than 4 µsec for a transfer to complete. This is to satisfy the bus-timeout requirements of the high-level computers, but it means that if the full complement of 16 processors are connected to the MPM, each bus operation must finish in 250 nsec. Several things are done to achieve this. No processor can own the bus for more than one operation. There is neither read-modify-write nor block-mode capability. The system controller contains a special 16-way round-robin arbiter. This uses wiring added to the backplane which provides individual bus request and bus grant lines for each processor interface. The timeout circuit is adjusted according to the needs of the slowest slave device, which is the memory, and must account for the actual access time plus any dead-time from error correction or memory refresh. In CESR, it is set for a 2 µsec period.

CESR uses about 3 Mbytes of a 4 Mbyte RAM board. It is error-correcting memory with a cycle time of 400 nsec. We discovered that the cycle time is more important than the access time in a multi-processor system. One memory board that was supposed to be fast malfunctioned if consecutive accesses occurred too close together. We expect to use

between 16 to 32 Mbytes of RAM for CESR-B. The performance of most modern RAM boards will be adequate.

Semaphores are provided for controlling access to shared data areas. The semaphore is a hardware test-and-set register. If the semaphore is not owned, a read operation returns a zero and sets the semaphore to one. If the semaphore is owned, a read operation returns a one. Writing resets the semaphore to zero. There is one semaphore for each longword (4 bytes) of RAM. This simplifies assigning semaphores. One just adds a fixed offset to a RAM address to access the semaphore associated with that address. The semaphores are implemented with a PAL and static RAMs. A 1 Mbit SRAM can make enough semaphores to cover 4 Mbytes of RAM, so it is easy to provide a sufficient number. Semaphore cycle time is 100 nsec.

The FIFO board is provided for passing messages between processors. A single operation can send a message to any number of processors. The FIFO also provides a queue to manage multiple messages to the same processor. The message passing protocol is defined so that the FIFO cannot overflow. Special wiring is provided so that an interrupt can be generated when a processor's FIFO contains a message. A processor can also poll a status register to determine if there is a message. FIFO cycle time is 150 nsec.

C. Bus Control Computers

The BCCs transfer data between the MPM and the accelerator hardware over the control bus, performing a variety of operations on the way. They are supposed to complete each transfer request as quickly as possible, then wait for another request. The computers contain interfaces to the MPM, the control bus, and, in some cases, the CESR Xbus. CESR uses 68020 CPUs and CESR-B will probably use the same family. Each computer runs a single common program; there is not even an operating system on them. The program is written on the VAX in 'C', compiled by a cross-compiler, and downloaded into the MPM. Bootstrap code in the BCCs moves the code from the MPM to local RAM and starts execution.

Under normal load, the CESR bus control computers are idle 85% of the time, which means requests are usually handled immediately. CESR-B will add two more computers to handle the additional equipment, for a total of seven. Boards with 68040 processors should be able to provide the CESR-B control system with enhanced performance appropriate to the heavier load.

The MPM interface passes memory references that fall within a particular address range on to the MPM. The BCCs are in the same physical location as the MPM. In CESR, we use a 32 bit, multiplexed, single ended connection between the BCCs and the MPM. It has an overhead of 300 nsec. For CESR-B, we will provide the same functionality, probably with the same type of interface.

The Xbus interface drives the control bus used in CESR. There are some places where CESR-B might use the same hardware as CESR, for instance in the LINAC or the Control Room, so an Xbus interface is required.

The control bus interface communicates with the interface crates around the lab. The details of this interface will depend on the design of the control bus itself.

D. Interface Crates

Interface crates will be distributed throughout the tunnel and in the control room. In the tunnel, there will be 4 control busses, each with 16 crates. The crates will be configured so as to maximize the effect of the parallel operation of the BCCs. This will involve two busses in each half of the tunnel, with crates alternating between the two busses.

The interface crates will use commercial VMEbus backplanes running the standard VMEbus protocol. At a minimum, they will support short (16 bit) and standard (24 bit) addressing with 1 Mbyte of address space per crate. Data width will be 16 bits. We may choose to support long (32 bit) addressing and 32 bit data.

The interface crates contain I/O devices which we would like to map into the address space of the bus control computers. The crate controller, which interfaces to the control bus, should be a simple design. We do not plan to have a general purpose processor board in each crate.

By using a commercial bus we will be able to buy circuit boards for many functions. However, we will most likely design and build our own interfaces for the more common functions. This will allow us to get exactly what we need, without too few or too many features. Maintenance will be simplified since we will use a consistent design for the bus interface logic.

We want to support geographical addressing, but this requires an extension to the VMEbus specification. Our plan is to divide the short address space between 16 backplane slots, yielding 4 kBytes of address space per slot. We will use four pins on the VMEbus P2 connector. On the boards that we design, these pins will be used to match address bits [A15..A12]. Commercial boards and designs that require more than 4 Kbytes of address space will use the standard (24 bit) addressing mode, with switches or jumpers on the board.

We will also investigate the issues involved with live insertion.

E. Control Bus

The control bus is a data highway between the interface crates and the bus control computers. We have not decided how to implement this connection. The speed and performance should be as good as or better than the Xbus used in CESR. This bus has about 4 μsec of protocol overhead plus a round-trip propagation delay of about 3.5 nsec per foot. It uses differential data transmission for noise control and parity for error detection. At a minimum, we need to transfer 24 bits of address, 16 bits of data, plus some control signals. The three options under consideration are a fully parallel system, an address/data multiplexed system, and a serial system.

The parallel system is logically the simplest. We would just need to buffer the address and data lines of the BCC.

Protocol time could be less than a µsecond and throughput would be limited by propagation delays. The drawbacks are the amount of cabling and the number of connectors and bus receivers. Forty wire pairs are required just for the address and data signals.

The multiplexed system, where the address and data signals share the cable, is almost as simple as the parallel one. The receiving boards would need an address latch, but fewer receivers. Another µsecond would be added to the protocol time for the address transmission phase. It still needs in excess of 30 wire pairs, so the cabling is not trivial, but it is a one-time process. There are commercial products that could satisfy our needs.

A serial system requires only a single conductor. It could be a fibre which would provide noise immunity. Cabling would be simple. The disadvantages are speed and electronics complexity. Moving more than forty bits of data in 4 µsec indicates that the data rate would need to be in excess of 10 Mhz. Serial-to-parallel conversion would be needed at both ends. We are looking at some FDDI chipsets which would simplify this type design.

F. Control Bus Extension

The control bus extension allows designers to build a control system interface into their systems and eliminates the need to bring many analog and digital signals to the interface crate. The extension will provide up to a 4 kbyte address space, which may be shared between several remote devices.

The choices for a bus extension are the same as for the main control bus. Minimizing congestion in the tunnel by minimizing the number and size of the cables and connectors makes a serial protocol highly desirable. We are looking at implementations using MIL-STD-1553, 16 Mbit per second token ring, high-speed UARTS, and TAXI chips.

G. Graphic Display System

The graphics display system will provide each high-level computer with direct access to video displays. There will be at least 4 color display channels and 16 monochrome channels. The channels will be distributed throughout the lab through our video distribution system.

There is a large FIFO connected to each computer. When a program needs to update a display, it allocates the FIFO and sends its data. Aside from the appearance of the graphics, there is no acknowledgement that the data has been transferred. The high-level computer doesn't have to wait.

We will be investigating X-terminals. In particular, we will look at their response time (can we turn a knob and have a reading on the screen track the changes?) and the amount of computer resources that they use.

H. Special Function Hardware

Some applications require a dedicated processor. These systems either handle large amounts of data or need higher update rates (>10Hz) than can be handled by the bus control computers. These include beam position monitoring, beam lifetime monitoring, the collision assurance system, and feedback control equipment. These systems interface to the control system via a shared memory and only transfer data that is needed by the high-level computers. An ethernet will be provided in the tunnel for maintenance functions, such as downloading and debugging.

The GPIB is supported by an interface in one of the BCCs. We do not have any CAMAC plans, but if needed, it too can be driven by a bus control computer.

IV. DATABASE

The database contains all of the information required to define the accelerator hardware and to communicate with it.

The heart of the database is the Name Table. It contains an entry for each node in the control system, where a node is a grouping of related hardware or software entities. The name table entry for each node contains a 12 character mnemonic name, information about how many elements and properties the node has, and pointers into the data area for each property. Properties include control bus addresses, scale factors, and raw and processed data. There is also information about which BCCs are used for a given node and the type of processing that is performed on the data.

Additional data structures are the hash table, the link table, the request packet area, the request packet address table, and the data area. These structures will be described by way of looking at a typical operation.

V. TYPICAL OPERATION

An example of a common operation is reading the quadrupole magnet currents. The application program requests data by making the subroutine call:

call vxgetn('CSR QUAD CUR',num1,num2,readout_vec)

where 'CSR QUAD CUR' is the mnemonic name for the CESR quadrupole magnet current node, 'num1' and 'num2' are the first and last elements of this node that the user wants to read out, and 'readout_vec' is an array where the data will be returned. This subroutine, like most of the control system routines, will not make any subroutine calls. It will directly communicate with the MPM. We will go through the sequence of operations performed by this subroutine and show how the hardware, software, and database function together. Performance measurements based on CESR will be provided.

A. Initialization

The VAX computers cannot directly map the MPM, but instead use interface registers. The VAX to MPM interface provides 32 sets of registers. When the VAX is booted, the VMS program 'SYSGEN' is used to create 32 dummy devices,

one for each register set. With this technique, the operating system handles allocation and cleanup. A program allocates a set of registers and owns them for the duration of its execution.

The SUN computer can directly address the entire MPM address space, so its programs simply use pointers. The mapping registers must be initialized at boot time.

The program also needs to allocate a Request Packet. It does this by reading the semaphores associated with the Request Packet Address Table. When an unowned semaphore is encountered, the program records the number and address of the Request Packet.

B. Search the Name Table

The mnemonic name is hashed by exclusive ORing the three longwords of the name and dividing the results by the size of the Hash Table. This produces an index into the Hash Table. The Hash Table entry, which is a pointer into the Name Table, is retrieved. The requested name is compared with the name found in the name table. If they match, then the search has been successful. If they do not match, then the Link Table, at the same offset, has the index of a new Hash Table entry. The name comparison is repeated.

In CESR, with over 900 nodes in the Name Table, the names will match on the first try 90% of the time. No lookups require more than three tries. A name table lookup requires 100 µsec.

C. Set Up and Deliver Request

The Request Packet owned by this process is filled in. The program inserts the Name Table pointer, the number of the first and last elements desired, and the type of operation that the BCC should perform. It also inserts 'BCCs used' bit field, which identifies which BCCs may be involved. This piece of data is inserted in four entries of the Request Packet; the 'used', 'start', 'done', and 'error' entries.

The 'BCCs used' bit field is combined (ORed) with the number of the Request Packet. The result is written to the FIFO board, which signals the appropriate BCCs that there is work for them. The logic on the FIFO board causes the number of the request packet to be written into all of the FIFOs which have a bit set in the bit field.

At this point, the program waits for the bus control computers to move the data. Programs can either go into a wait loop or they can hibernate. The choice depends upon the programmer and how many elements are involved. Most programs hibernate, which contributes to the efficiency of the high-level computers.

D. Bus Control Computer Operation

The BCCs are normally executing in a loop, checking their FIFO to see if there is a message. When there is one, the BCC reads the Request Packet number from the FIFO. In the Request Packet, the computer clears the bit which identifies it

in the 'start' entry. Since there may be several bits set, a semaphore must be used to guarantee that only one computer at a time is changing a bit.

From the Request Packet, the computer gets the name table pointer, the element numbers, and the operation mode. It verifies that all database pointers required for the operation are valid. It uses an entry in the database which specifies the first and last elements that this computer handles to modify the element numbers from the request Packet. This keeps the computer from spending time trying to transfer data related to elements controlled by another computer.

Once all of the checking is complete, data movement begins. The control bus address of the next element is retrieved from the database. If the address indicates that the element is controlled by this computer, the raw value of current is read over the control bus from the magnet and stored in the database. If the element is controlled by another BCC, then this computer goes on to the next element.

Using parameters from the database, the raw value is then scaled and offset. The final result is written to the database. The time and status of the operation are also stored. Error information, if any, is saved. The process continues until the last element is done. Notice that other BCCs may be working on this request at the same time.

After all of the elements have been processed, the computer clears its bit in the 'done' entry of the Request Packet and in the 'error' entry (if there were no errors). Again, semaphores are used when changing bits.

E. Retrieve Status and Data

The high-level computer reads the 'done' entry in the request packet. When all of the bits are zero, then all of the bus control computers have finished. If the 'error' entry is all zeroes, then there were no errors. The data is read from the database and moved into the user's array.

In CESR, elements 1 through 49 of the 'CSR QUAD CUR' node are on one control bus and elements 50 through 98 are on another. The execution time of the 'vxgetn' subroutine to read the current from one magnet takes 700 µsec. Reading 49 currents requires 4780 µsec, or 80 µsec per element. The bus control computers use 50 µsec and the high-level computer uses 30 µsec. When reading all 98 elements, the advantage of the parallel operation of the BCCs becomes obvious. It takes 6200 µsec. The extra 1420 µsec is just the time that the high-level computer needs to move the data for the additional elements into the user's array.

VI. REFERENCES

[1]   CESR/CLEO Staff,"Conceptual Design for a B Factory Based on CESR," CLNS 91-1050, 1991.

[2]   C. R. Strohman and S. B. Peck, "Architecture and Performance of the New CESR Control System," in the *Proc. of the 1989 IEEE Particle Accelerator Conference,* Chicago, IL, March 1989. pp. 1687-1689.

Standards and the Design of the Advanced Photon Source Control System[*]

William P. McDowell, Martin J. Knott, Frank R. Lenkszus, Martin R. Kraimer, Robert T. Daly, Ned D. Arnold, Mark D. Anderson, Janet B. Anderson, Robert C. Zieman, Ben–Chin K. Cha, Fred C. Vong, Gregory J. Nawrocki, Gary R. Gunderson, Nicholas T. Karonis, and John R. Winans

Argonne National Laboratory
Advanced Photon Source
9700 South Cass Avenue
Argonne, Illinois 60439

I. INTRODUCTION

The Advanced Photon Source (APS), now under construction at Argonne National Laboratory (ANL), is a 7 GeV positron storage ring dedicated to research facilities using synchrotron radiation. This ring, along with its injection accelerators is to be controlled and monitored with a single, flexible, and expandable control system. In the conceptual stage the control system design group faced the challenges that face all control system designers: (1) to force the machine designers to quantify and codify the system requirements, (2) to protect the investment in hardware and software from rapid obsolescence, and (3) to find methods of quickly incorporating new generations of equipment and replace obsolete equipment without disrupting the existing system. To solve these and related problems, the APS control system group made an early resolution to use standards in the design of the system. This paper will cover the present status of the APS control system as well as discuss the design decisions which led us to use industrial standards and collaborations with other laboratories whenever possible to develop a control system. It will explain the APS control system and illustrate how the use of standards has allowed APS to design a control system whose implementation addresses these issues. The system will use high performance graphic workstations using an X–Windows Graphical User Interface (GUI) at the operator interface level. It connects to VME–based microprocessors at the field level using TCP/IP protocols over high performance networks. This strategy assures the flexibility and expansibility of the control system. A defined interface between the system components will allow the system to evolve with the direct addition of future, improved equipment and new capabilities. Several equipment test stands employing this control system have been built at ANL to test accelerator subsystems and software for the control and monitoring functions.

II. STANDARDS AND THE APS CONTROL SYSTEM

The APS control system must be capable of (1) operating the APS storage ring alone and in conjunction with its injector linacs, positron accumulator, and injector synchrotron for filling,

High performance processor
Program development
Log/archive
Bridge to other networks
Some applications
Clients for X–terminals

Console Computer
RISC Processor
Most Applications
Program Development
Most Console Functions
4 at Central Control
1 at injector area

Injector Input/Output Controllers (IOC)
M680x0 Processor(s)
Most interface functions
Alarm Checking
Closed loops
7 at linac
5 at positron accumulator
7 at synchrotron
1 at high energy transfer

Storage Ring IOCs
M680x0 Processor(s)
1 at each sector–pair
2 at specal diagnostics
2 at rf building
1 for ring magnets

Remote Console
Provides all console functions
All central console functions
Anyware on network
Functions as local console

Equipment Processors
Status details
Preprocess data
PS: 1/section (5 x 40 = 200)
Raw PS: 1/sector pair (20)
Diagnostics 1/sector pair (20)

Figure 1. APS Control System

and (2) operating both storage ring and injection facilities as machines with separate missions. The control system design is based on the precepts of high–performance workstations as the operator consoles, distributed microprocessors to control equipment interfacing and preprocess data, and an interconnecting network. In a paper presented at the 1985 Particle Accelerator Conference [1] we outlined our initial approach to the APS control system. In this paper we predicted

that the control system would use workstations for the operator interface, single–board computers to control the front–end electronics, and a network consisting of either Ethernet or Token–Ring. The APS control system today is remarkably close to the initial design concepts due to rapid performance gains in computing workstations, low cost network connections, both Ethernet and Fiber Distributed Data Interface (FDDI), and availability of real–time operating systems for the front–end computers.

Figure 1 shows in schematic form all major components and their relationships. The current design includes about 45 distributed microprocessors and five console systems, which may consist of one or more workstations. An additional 70 Input–Output Controllers (IOC's) will be used to control the insertion devices, front ends, and beam lines.

The operator interface (OPI) is implemented with a high performance graphic workstation and uses an X–Windows based GUI. Standards play a large role in the selection of the OPI since the hardware, operating system GUI, and network must be compatible. The ideal control system design would be vendor independent in all of these areas. To make the APS control system vendor independent we chose to use standards when selecting these components.

A. Standards

The definition of the word "standard" used for the purposes of this paper is as follows: "Something established by authority, custom or general consent as a model or example" [2]. Past practice at large laboratories has been to invent almost everything that was needed to build a control system. Accelerator control system groups have built computer systems and designed networking schemes. Of course there were good reasons for this – the laboratories were often pushing the leading edge of electronic and computer technology and the required devices and techniques were not available on the open market. This picture has greatly changed. Computer technology has now spread into every corner of our lives. There are literally tens of thousands of companies inventing new uses for computers and pushing the limits of technology. This has had a very positive effect on control system design as the effort required to build a control system can now be redirected towards control and accelerator details rather than details associated with building a computer or computer network. In the Proceedings of the Second International Workshop on Accelerator Control Systems [3] held in October of 1985, no discernible trend can be observed in control system design. This contrasts with the sense one receives from reading the titles of the the papers presented at the 1991 Particle Accelerator Conference. These titles show a ground swell towards what could be called a generic control system. The generic system consists of workstations running UNIX, a network, and front end processors running a real time operating system. We now find standards being followed at all levels of control system design.

B. Operating Systems/Workstations

At the Europhysics Conference on Control Systems for Experimental Physics [4] in October of 1987, discussion panels ran late into the night with the "religious" arguments for the choice of UNIX or VMS as the operating system of choice for control systems. There where convincing arguments presented by advocates of both sides of the discussion. Four years later the argument has been settled, not because either of the opposing sides was won over by a technical argument but because of market forces. The development of the the Reduced Instruction Set Computer (RISC) processor has resulted in UNIX dominating the workstation market.

RISC is a recent innovation in computer architecture (although some people claim that the PDP–8 was a RISC machine). The study of computer instructions and their relative frequency of usage revealed that most of a computer's time is spent in the execution of a small subset of its repertoire. RISC architecture takes advantage of this fact by streamlining the execution of this subset and by implementing the less used and more complex instructions with combinations of the (now fast) small set of instructions. Since there is now a small set of simple instructions, parallel "pipelining" can be used to increase execution speed. In this method more than one instruction can be executed simultaneously by staggering in time the various suboperations. Some processors can even average more than one instruction per clock cycle.

The converse to RISC architecture is Complex Instruction Set Computer (CISC) architecture. Most computer architectures developed prior to 1980 are of the the CISC type, a typical example being the VAX. Today there is a five–to–one advantage in raw MIPS (millions of instructions per second) for RISC devices. This should be discounted to some degree since RISC requires more instructions to perform some types of operations, but an advantage of even two–to–one on reasonable benchmarks is obtainable.

The UNIX operating system itself was originally developed by Bell Telephone Laboratories as a word processing tool, but it was soon modified to support software development tools and finally grew into a full–featured operating system. UNIX was written in the "C" language, also developed at Bell Telephone Laboratories. The keys to UNIX's success are that it is extensible and it is written in a portable language. These attributes allow the user to make enhancements, remove features, and tailor UNIX to specific needs. One indication of this is the fact that the UNIX operating system is available for microprocessors as well as supercomputers. Thus, if a start–up company chose UNIX as its operating system and made the changes necessary to support its chosen computer architecture, any existing software that ran under UNIX could be recompiled to run on their computer. In this way new computer architectures can be introduced with ready–made operating system software and trained users.

Because of the development of RISC processors and the existence of UNIX, nearly all computer manufacturers are developing and marketing RISC–based computers and workstations which use the UNIX operating system. Competition is driving performance up while keeping prices low and this trend is likely to continue. There is still a market for CISC architecture computers and operating systems such as VAX/VMS, principally due to the installed base of application software and the steady improvements made to the hardware by vendors.

These reasons seem to make it obvious that the operating system of choice for any control system to be delivered in the mid 1990s will be UNIX. The bottom line for APS is the fact that the UNIX operating system provides the control system a large measure of vendor independence. We have the OPI software running on SUN 4 and Digital Equipment DS5000 workstations and expect to port the system to other vendors' workstations.

The GUI "wars" now being fought in the press and on workstations provide a very good reason to conform to standards when writing the OPI software. APS is developing applications using the Open Software Foundation's Motif toolkit. We are extensively testing the software against the two major window managers Motif and Openlook.

C. Front End Systems

The IOC, or front–end electronics, is implemented with single–board computers of the Motorola 680X0 family, packaged along with signal interface cards in VME and VXI form factor crates. Motorola 68020 processors are used in initial configurations with 68040 processors planned for most future configurations. A real–time operating system, VxWorks from Wind River Systems Inc., is used to provide multitasking, high performance front–end software. More than fifteen VME input–output modules are currently supported. These modules include binary input and output, analog input and output, motor drivers, counter timers, and subnet controllers. More modules will be supported as they become available and are required. Most information preprocessing is performed at this level with only engineering units sent to the OPI for display. Signal monitoring can be set up to communicate only on signal change or limit–breaching or at some preselected rate. Local sequential and control–loop operations can also be performed. In this way, maximum benefit is gained from the many IOC processors operating in parallel. This is one area where APS is vulnerable to complications which would arise if the vendor of the real–time software failed. When the posix standard for real–time systems becomes a reality and most real–time vendors conform to the standard, our estimate is that it would take about two months of a very knowledgeable programmer's effort to change real–time kernels.

In addition to local IOC I/O, subnets are utilized to interface remote points where an IOC may not be present to a distant IOC. There are currently three supported subnets in the APS control

system: Allen Bradley, GPIB, and BITBUS. The Allen Bradley I/O subnet uses the 1771 series I/O modules to provide basic binary and analog I/O support for the APS control system. Allen Bradley is a inexpensive and rugged standard for industrial control systems. Copper and fiber optic based multidrop subnets are available for this equipment.

Laboratory test and measurement equipment often use the GPIB standard as an interface to an external control system. This multidrop standard presents some serious challenges and potential problems to the system designer. GPIB has a distance limitation of 20m which requires the instruments connected to the bus to be in close proximity to the IOC. In addition, the signals within the GPIB cable are not balanced and thus susceptible to EMI/RFI noise and ground spikes. Signal transfer and isolation techniques are not part of the GPIB standard and although commercial equipment is available to extend the distance of a GPIB interface, it is prohibitively expensive.

BITBUS provides a method of high speed transfer of short control messages over a multidrop network. The BITBUS subnet can be used as a method for remote, single point I/O as well as a gateway for remote GPIB and RS232 signals [5]. A differential, opto–isolated, wired subnet is the BITBUS standard; however, a multidrop fiber optic network has been developed for BITBUS at the APS.

D. Networks and Protocols

Argonne uses Ethernet as the intra–laboratory network. There are backbone cables in the individual buildings with communication between buildings presently done via the Lanmark PBX system. Intra–building FDDI will be available within 6 months. The control system development computers are presently sharing the APS Ethernet backbone with all other APS computing needs (55 Sun Workstations, a VAX Cluster with six members, 18 terminal servers, 40 PC's using Pathworks, and 40 Macintosh systems). Two test stands and six development IOC's are running in this environment without experiencing network–induced problems. A Network General Sniffer is on line at all times should the need arise to diagnose an apparent problem. In the APS facility, however, we plan to use FDDI with Ethernet branches as performance needs dictate.

The central feature of both the OPI and IOC software designs is the protocol for connections between software modules for the purpose of exchanging information. This protocol is called channel access [6] and is built on the TCP/IP Standard. TCP/IP is an integral part of every UNIX–based workstation as well as being built in to VxWorks, the real–time operating system. When an OPI application program needs to connect to a process variable located in an IOC, it issues a broadcast over the network and the IOC in whose database the requested process variable resides provides a response. A socket–to–socket connection is established and thereafter efficient two–way communication takes place. IOC–to–IOC channel access can take place to exchange inter–IOC information. It should be noted that the OPI

needs no knowledge of the location of the desired process variable, only its name. Figure 2 shows the relationship of the channel access software in both the OPI and IOC systems. It also illustrates how a mouse and screen "slider" are used to communicate, through channel access software at both ends, with a D–A convertor. Similarly, A–D convertor output finds its way to a screen.

The database which defines all IOC channel connections and properties is distributed over the many IOC's and downloaded at IOC boot time along with the operating system and the particular device driver software modules required by each IOC. The entire database is centrally maintained and configured with a UNIX workstation which, of course, can be any OPI. Figure 2 shows the downloaded location of the IOC database in the overall data flow.

E. X–Windows

In the X–Windows client–server paradigm, an application program is divided into the "client" (which provides the computation and logic of the program) and the "server" (which provides the interaction facilities for the human operator or user). In the APS control system, both client and server are implemented in processors at the OPI level. The client and server need not reside in the same processor so that, for example, a specialized parallel processor may provide client services for a more common workstation server or X–Window terminal. In this way, the OPI's will be able to have windows open to clients operating both locally and on other processors on the network.

F. Application Software

Application software comprise those programs which the operator or physicist invokes to provide a feature or service not provided by the equipment operation level of the control system. An example would be the software required to provide a local bump in the orbit. These programs can be of two general forms. The first is a control panel which is created during a session with a display editor (see Figure 2, upper left). Graphic tools such as buttons, sliders, indicator lights and meters, and graph paper are selected and located on the panel. Static entities which can be used to depict the physical system, such as piping diagrams, are added where appropriate. Connections to IOC channels are specified at this time and the proper drawing list and action code are automatically generated. When complete, the panel is called up for execution, the channel access calls are made, and the control panel is now "live." No actual code is written or compilation made, aside from that originally involved in the tools themselves. The software provides calculation records and allows cascading of physical inputs and outputs with these calculations. This allows very complex operations to be designed. The second form of application program is that of employing classic in–line code generation. In this case standard entry points are provided to the same graphic and channel access tools. Using this approach, an existing code can be adapted to our
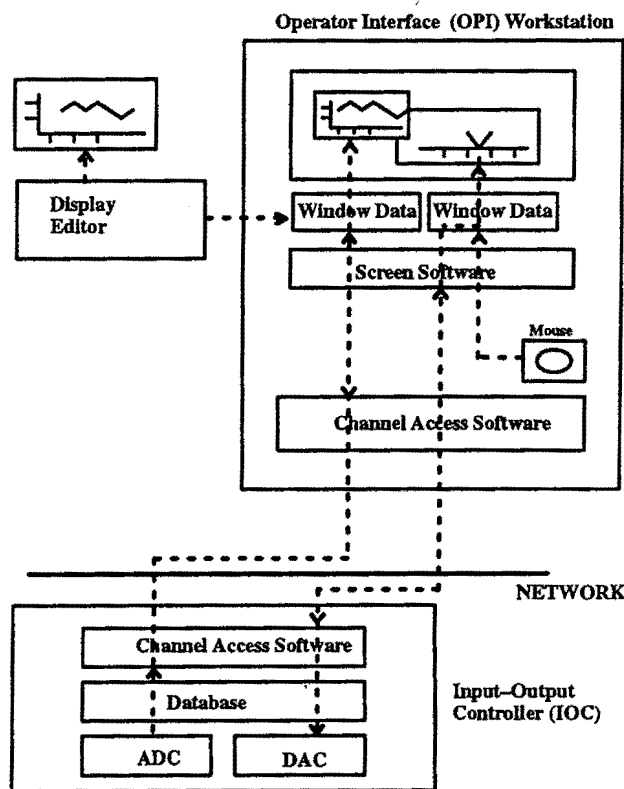
Figure 2. APS Control Environment

system by calling the channel access code and displaying the results using either traditional line–by–line or graphical output.

All software is being developed under UNIX, including that for the IOC's. In this way, windows can be opened simultaneously at an OPI for software development, actual run–time applications, database configuration, electronic mail, etc. This streamlines software development, database servicing, and system troubleshooting.

III. INTERLABORATORY COOPERATION

At the Accelerator Control Toolkit Workshop [7] in 1988 a group of people responsible for accelerator control systems at laboratories throughout the world spent a week discussing various aspects of control systems. One of the topics discussed was the development of "tools" which could be used at more than one laboratory. Subsequent to this meeting we decided that the APS would pursue the idea of looking at existing control systems with the aim of determining if they could be used at APS. After much discussion we decided to pursue collaboration with Los Alamos National Laboratory (LANL). Discussions were held with the developers from LANL and it was decided that APS would send a representative to LANL who would use the system to develop an application which would be useful to LANL. One of us (Kraimer) spent a summer at LANL developing the software to control a magnet measuring facility. Upon his return he imparted his positive impressions of the system. We then

decided to complete an in–depth study of the software. He sequestered himself in his office with the software listings. As he gained understanding of the system he gave tutorials to the APS controls group staff on the internal design of the software. Further group discussions led to the decision to try to form a cooperative development team with LANL. M. Knott (ANL) and M. Thout (LANL) proceeded to develop an agreement that has led to the co–development of the Experimental Physics and Industrial Control System (EPICS). The paper entitled " EPICS Architecture" [8] by L. R. Dalesio, et al. presented at this conference provides a detailed look at the features of EPICS.

IV. CONSTRUCTION STATUS OF APS AND THE APS CONTROL SYSTEM

Construction is proceeding rapidly on the physical structure of the APS. As of this date (late October 1991) the linac and injector buildings have steel erected, the concrete for the linac tunnel and positron accumulator ring vault is in place, the control center has reached the first floor level, and the foundations are in for the first section of the storage ring building which will be used as an early assembly area and magnet measuring facility. Barring unforeseen construction delays, the linac and control center are scheduled for occupancy in April of 1992.

The APS control system is now actively supporting two test stands, rf and linac. Work on these test stands started in 1989. In their first implementation they used a predecessor version of the OPI running on a VAXStation under the VMS operating system and a predecessor of EPICS called GTACS (Ground Test Accelerator Control System) for the IOC. As work on a UNIX–based OPI and EPICS progressed, both test stands converted to the UNIX OPI software and EPICS. The APS rf test stand was reported at the Real Time '91 Conference [9]. Two IOC's are being used to implement the linac functions: one for beam diagnostics and the other for control. The test stands have proved to be highly beneficial to both the controls group staff and the linac and rf systems development team members. The controls group has gained experience in using the control system as well as received suggestions for changes and improvements. The test stand staff has been able to concentrate on linac and rf design details without developing their own control system. The only way provided to remotely run the test stands is via the control system.

Progress in the development of EPICS software is continuing. An alarm handler [10] has been developed and is being optimized. We are continuing to add device and driver support for new hardware modules as well as develop new record types such as pulseTrain, pulseDelay, etc. A graphical database link display tool is being developed as a way to document databases and requirements are being developed for a system–wide database and a system–wide error handler to accept

and process IOC–generated errors. We are developing low cost IOC's based on single–height VME modules as well as Gespac G64 modules. VXI crates using the standard VME processors and network boards are currently operating. On the OPI side we are running on both the SUN 4 and DEC 5000 platforms and we will soon port the system to Hewlett Packard 700 series workstations.

V. REFERENCES

[1] M. Knott, G. R. Gunderson, F. R. Lenskzus, W. P. McDowell, "Control System Features of the Argonne 6GeV Synchrotron Light Source," IEEE Transactions on Nuclear Science, Vol. NS–32, No. 5, October, 1985.

[2] Webster's New Collegiate Dictionary, Merriam, 1977.

[3] Accelerator Control Systems, Proceedings of the Second International Workshop on Accelerator Control Systems, Nuclear Instruments and Methods, Vol. A247, No. 1, 1985.

[4] Proceedings, Europhysics Conference on Control Systems for Experimental Physics, CERN Report 90–08, 1990.

[5] N. D. Arnold, G. J. Nawrocki, R. T. Daly, M. R. Kraimer, W. P. McDowell, "I/O Subnets for the APS Control System," to be published in the Proceedings of the 1991 Particle Accelerator Conference, San Francisco, California, May 6–9, 1991.

[6] J. O. Hill, "Channel Access: A Software Bus for the LAACS," in Accelerator and Large Experimental Physics Control Systems, D. P. Gurd and M. Crowley–Milling, Eds., ICLAEPCS, Vancouver, British Columbia, Canada, pp. 288–291, 1989.

[7] Accelerator Automation Application Toolkit Workshop, Sponsored by AT Division and MP Division, Los Alamos National Laboratory and CERN, Geneva, 31 October – 4 November, 1988.

[8] L. R. Dalesio, M. R. Kraimer, A. J. Kozubal, "EPICS Architecture," to be presented at ICALEPCS '91, Tsukuba–shi, Ibaraki, Japan, November, 1991.

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

[10] M. R. Kraimer, B. K. Cha, M. D. Anderson, "Alarm Handler for the Advanced Photon Source Control System," to be published in the Proceedings of the 1991 Particle Accelerator Conference, San Francisco, California, May 6–9, 1991.

The ESRF Control System; Status and Highlights

W.-D.Klotz
European Synchrotron Radiation Facility
BP 220
38043 Grenoble Cedex, France

1 Introduction

The European Synchrotron Radiation Facility[1][1] will operate a $6GeV$ e^-/e^+ storage ring of 850m circumference to deliver to date unprecedented high brilliance X-rays to the European research community. The ESRF is the first member of a *new generation* of Synchrotron Radiation Sources, in which the *brilliance* of the beam and the utilization of *insertion devices* are pushed to their present limits.

Commissioning of the facility's storage ring will start in spring 1992. A full energy injector, consisting of a 200MeV linear preinjector and a $6GeV$ fast cycling synchrotron ($10Hz$) of 350m circumference have been successfully commissioned during the last months.

The machine control system for this facility, which is under construction since 1988, is still under development, but its initial on-site operation this year has clearly made easier the commissioning of the preinjector plant.

A description of the current system is given and application software for start-up is briefly described.

2 Architecture

The ESRF control system is based on a multi-level architecture of distributed hard- and software processing units[2]. Logically the system is structured into four levels. From top to bottom we call them:

- Console Level (Presentation);
- Process Level (Applications);
- Group Level (Device Servers);
- Field Level (Equipments, Embedded Controlers).

On the lowest level, all equipments are interfaced; either by intelligent controlers, as they are delivered from the manufacturer, or by dumb interfaces. Equipments are logically grouped together on the group level. Grouping of equipments is done by similar functionality. The group level is responsible for hardware specific and real time I/O-operations. *Device servers* perform the task of hiding hardware specifics to the upper level. The process level

[1](ESRF)

represents that level of the control system where practically all higher level control tasks take place and where physics applications are processed. Powerful multitasking capabilities and fast processing is mandatory on this level. The presentation level presents the interface between the operators and the system. Within this level data entering from the lower level are presented graphically or are formatted to readable reports. Commands entered by means of interactive devices are decoded into events and finally passed as internal messages to the lower levels.

Physically the system is split into 2 levels. All nodes of the presentation and process level consist of UNIX based workstations and file/compute servers interconnected by Ethernet. The group level nodes are realised by VMEbus crates, equipped with 68030 CPU boards. These systems run the OS9 multitasking real time kernel/operating system. Every process level server connects to a private Ethernet segment onto which group level nodes it is in charge of are connected.

The physical boder line between group level and field level is fuzzy. In our system some dumb devices are directly interfaced to VME I/O-boards that are plugged into group level crates, but most dumb devices are interfaced by means of G64 crates. Groups of G64 crates, that interface classes of similar devices, are connected to multidrop highways that are mastered by group level crates. This multidrop highway[2] was developed at ESRF. However, intelligent devices with embedded controlers are, in the majority of cases, directly connected to VME (group level) crates[3].

3 Networks

Modern control systems are distributed[3, 4]. The larger the accelerator is, the more important is its network infrastructure. The ESRF control system is fully distributed and relies strongly on a high speed computer network.

Figure 1 gives an overview of the logical and physical implementation of the control system and its network.

Apart from the home-made multidrop highway all computer connections are based on the Ethernet(IEEE 802.3)

---

[2]we named it FBUS
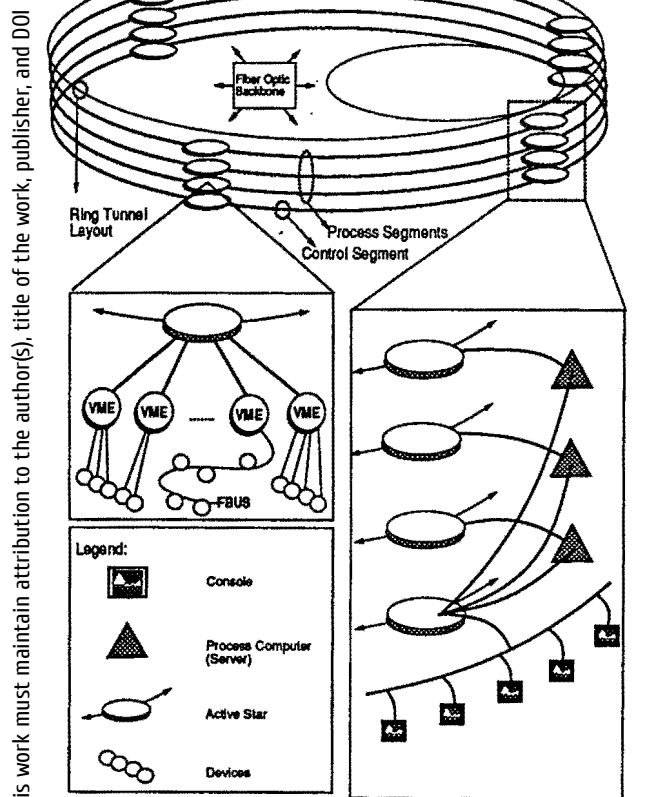[3]in the majority by RS422 or RS232 asynchronous serial links

Figure 1: Overall Layout of Control System and Networks

LAN standard and the TCP/IP protocol suite. The network is constructed using $50/125\mu m$ multimode graded index optic fibres[4] for cabling, that will allow a later migration to FDDI.

Four networking centers are located around the storage ring tunnel, a center comprising a "NODE" and a wiring "HUB". A NODE is the location that acts as the convergent point for IEEE 802.3 compliant active devices e.g. an *active star*. To the NODE are attached "LOBES" which are configured on a star wired topology. A LOBE is the remote system connected to a NODE, and in this case it is a remote "transceiver" attached by a standard AUI[5] cable to a VME crate, process computer, graphics workstation or fanout unit.

For the active stars at the NODES, the "Lannet Multinet II" system was chosen for its ability to operate in a single chassis and support up to four independent backbone bus's. In addition all backbone fibre optic links are run in a *synchronous* Ethernet mode. By using this mode it is possible to install more than the "four" repeaters that restricts standard asynchronous Ethernet systems, with the restraining factor being, the round trip delay that limits

---

[4]Bandwidth specified at $850nm > 500MHz * km$ and at $1300nm > 700MHz * km$
[5]Attachment Unit Interface

each Ethernet segment to approx. 4.5 kilometers.

The wiring HUB is the central point for passive network components, i.e. the *backbone optical fibres* that link all the HUBs together in a circular structure around the storage ring tunnel. It also acts as the termination point for all star wired fibres that are attached to the LOBES.

When installing the circular *backbone*, provision for 10 independent rings has been made, out of which four will be used initially. One ring functions as the main *control segment*, to which all upper layer processors are connected. There are three dedicated *process servers* that operate as network gateways to the other three rings. The latter are used as *process segments* to which all middle layer VME systems are connected. Devices in the field are either connected directly to them or accessed through the multidrop highway.

Response times on a heavily loaded Ethernet become quasi stochastic. Having this in mind, the whole system was designed in a way to provide maximum flexibility in distributing sinks and sources of network traffic. This is accomplished by fiber optic patch panels situated by the HUBs. By simply crosspatching between panels, any processor can immediately be connected to any of the independent rings comprising the backbone. Upgrading the backbone to FDDI would be another, although much more elaborate, remedy against network congestion.

4 Computers

4.1 Consoles

The standard console in the control room will be an HP Apollo 9000 Model 720 workstation with local disk to hold the bootable image of the operating system and to provide local swap space. File systems containing control system and physics applications software are remotely mounted[6] from the process servers. Currently we are running 5 consoles in the main control room.

In addition to that, RISC based X Window graphics terminals, like the HP 700/X familiy, that are served by the process servers, will be used as console devices. Their ideal usage will be that of "remote" consoles. They will be outside of the control room but plugging to the backbone's control segment. Their main purpose is to give scientists, working on an experiment, the possibility to control the movements of *their* "insertion device" by themselves.

4.2 Process Servers

As process servers the HP 9000 Series 800 Model 842 and 857 midrange super-mini computers were selected. There are currently three of these machines in the system. System features are:

- high reliability;

---

[6]currently we are using SUN's NFS that we hope to replace by OSF's DCE

- 29 Mips; 6.9 DP[7] Mflops; CMOS RISC technology;
- internal disk storage: <2.68 Gbytes;
- I/O bandwidth 21 Mbytes/sec;
- integrated DAT[8] unit: 1.3 Gbytes capacity;

These machines are configured as network *gateways* between the control segment and one of the corresponding process segments. To accomplish this, each of them is equipped with two high speed LAN adapters.

The process servers and the console workstations run HP-UX; an AT&T System V Rel 3.0, and BSD 4.3 compliant implementation of the UNIX operating system. All machines run MIT's X Window System, which allows applications to run as X-*clients* on the process servers and to perform interactive I/O through X-*servers* running on the consoles.

4.3 VME Systems

All VME systems have an identical base configuration. This comprises a CPU with Motorola 68030 @ 20 MHz, 4 Mbyte RAM, on board Ethernet adapter, and additional 512Kbyte battery backed up RAM on a separate board.

On these systems we run Microware's OS9 realtime kernel/operating system. In addition the systems are equipped with a TCP/IP Internet Support Package providing Berkely sockets, and SUN's Network File System.

All systems are running in a *diskless* configuration for ease of maintenance and reliability. The battery backed 512Kbyte RAM holds the OS9 real time kernel and a minimal TCP/IP configuration. When the system boots, it loads additional OS9 modules, applications, and the NFS modules from a process server. Using NFS, it then copies configuration files from the process server into its RAM disc and initialises the applications.[9] Cold start-up still has to be done manually by toggling a switch on the board's front panel, but we are working on a remote facility.

5 Interfacing

Many of the VME systems drive fast multidrop highways. This FBUS is not a general purpose network but implements a low cost remote input/ouput facility. It relies on a master-slave relationship, where a controller (VME based module) drives a large number[10] of slave nodes. The nodes comply with the G64 standard, so that full advantage can be taken of existing interface boards from industry. The FBUS multidrop highway is based on a synchronous serial protocol. Physical implementation uses an extremly noise resistant Manchester encoding with transformer isolation, i.e. each node being galvanically isolated from the

highway. The data rate is up to 2 Mbits per second on a 200m-300m long highway. FBUS can still be safely operated at a speed of 1 Mbits per second on distances of up to 1 km and 30 nodes without repeater. The transmission medium is a flexible shielded twisted-pair cable with a characteristic impedance of 78 Ohms.

G64 and FBUS interfacing has been used for control of main magnet power supplies, beam position monitors, magnet interlocks, corrector magnet power supplies, and injection/extraction elements. Other significant subsystems that include G64 crates are the system to distribute the slow timing pulses, the video cross point switch, and the video multiplexors for fluorescent screen monitors. The rest of devices is directly interfaced to the VME systems; either by asynchronous serial lines or digital I/Os.

As a result of an early taken policy, to stick to industry standards, *only a few* boards had to be designed by the ESRF Digital Electronics team:

- video multiplexor[11] (VME)
- delay unit[12] (VME)
- ADC with on-board memory (G64)
- FBUS master (VME and PC/ATbus)
- FBUS slave (G64)
- clock divider[13]

Unavoidably some other dedicated electronics had to be designed to adapt some exotic devices to the standards chosen.

Table 1 gives an overview of interfacing.

6 Software

6.1 Equipment Access

The low level system software for the distributed control system is based on a "Client/Server" architecture. The Client/Server technology is a simple mechanism to distribute software tasks across any number of processors. This approach is *open and object oriented*, can be implemented on existing systems (eg. OS9 and UNIX), and will be discussed in detail by a contribution to this conference from A.Götz.

Objects are sets of *hidden data* on which well *defined operations* may be performed by authorized users. Associated which each object is a "Server" process that manages the object and exports its functionality as a service. This server-based model is implemented using "Remote Procedure Calls"[14][5, 6]. When a user process wants to perform an operation on an object, it sends a request message to the Server in charge of it. The message contains access keys, a specification of the operation to be performed, and any parameters the operation requires. The user process,

---

[7]With floating-point processor, Double Precision

[8]Digital Audio Tape

[9]Microware has recently started to ship the BOOTP Port Packs for OS9. BOOTP is used to make a boot PROM with the possibility to boot OS9 directly over Ethernet, thus avoiding the battery backed up RAM

[10]up to 64 on one highway

[11]15:1

[12]6 channels, 32MHz resolution, 0-524 ms range

[13]for RF-synchronous triggers, 352MHz:32MHz

[14]RPC; in our implementation we use SUN's RPC and XDR

VME crate	VME dig I/O	VME serial I/O	VME special I/O	G64 crate	G64 analog in	G64 analog out	Where
1	20	8	6	15	84	84	Transfer Line 1
1	48		6	4	3	3	Transfer Line 2
1		6	2	3			Injection/Extraction
1			16	5	75		Synchr. Diagnostic
2		3	15	9			Synchr. Magnet Power Supplies
3		56					Synchr. Vacuum
1		34	36	8			SY/SR slow timing
1			37	33	224		Storage Ring Diagnostics
1		21	1	36	36	36	Storage Ring Magnet Power Supplies
16		448	34	32	2050		Storage Ring Vacuum
1		1	5	51	576		Geodesy & Mag. Interlocks
14		48	14	9		9	Insertion Devices
			68	17	816		Front Ends
		41	14		82		X-ray BPMs
2	34	53					Radiation & Safety
2				2			misc. Monitoring
4							RF & LINAC (subcontr.)
51	102	719	254	224	3946	132	TOTAL

Table 1: Overview of Hardware Interfacing for the ESRF accelerator plant

known as the "Client", then blocks. After the Server has performed the operation, it sends back a reply message that unblocks the Client. The combination of sending a request message, blocking, and accepting a reply message forms a Remote Procedure Call, which can be encapsulated to make the entire remote operation look like a local procedure call.

The lowest level of objects in the accelerator plant are the actors and sensors (or physical devices). These objects are "terminator objects"; they are easy to identify, and their behaviour can be modelled and documented.

"Device Servers" are terminator objects that operate on physical devices. The more general term "Server" or "Virtual Device Server" covers objects on a higher level of abstraction. Objects on a higher level of abstraction can use terminator objects to offer a given service[15].

The Device Server is an intermediary between application programs and the physical resources of the accelerator system. It contains all device-specific code, and insulates applications from differences between hardware. It performs the following tasks:

- Allows access to the device by multiple clients. To implement security, the server, depending on its state, may deny access from certain clients.

- Interprets network messages from clients and acts on them. Messages are generated by clients through RPCs.

- Maintains complex data structures, including device state information. Server maintained information reduce the amount of data that has to be maintained by each client and the amount of data that has to be passed over the network.

At ESRF the Device Servers follow the OOP[16] paradigm and have to be implemented in a certain, fixed style. The OOP paradigm is based on the "widget" model from the X11 Intrinsics Toolkit[7] of MIT and is implemented in ANSI C.

The control system designers have decided to implement *all* important functions necessary to run the distributed system in Servers. This includes the processes to boot and manage the system, to access the database, to handle graphics objects, as well as Device Servers to access equipments.

According to our current state of knowledge about 53 Device Servers have to be written for the complete system. About 50% of them are currently released. The average size of a Device Server ranges typically between 2000–2500 lines of C code.

---

[15]The encapsulation of lower level services into higher level services can continue until a very high abstraction like physical machine parameters, i.e. energy, chromaticity, tune, emittance,... , is achieved.

[16]Object oriented programming

3rd Int. Conf. Accel. Large Exp. Phys. Control Syst.   ICALEPCS1991, Tsukuba, Japan (partial, pages 194 of 611)   JACoW Publishing

ISBN: 978-3-95450-254-7   ISSN: 2226-0358   doi:10.18429/JACoW-ICALEPCS1991-S03SRD05

6.2   Graphics User Interface

This field of technology is in a state of tremendous innovations. Fortunately some standards exist now: **X11**, and **Motif**. The X11-window system, or X11, is a network-transparent window system. With X11, multiple applications can run simultaneously in windows, generating text and graphics. Network transparency means that application programs that are running on other machines scattered throughout the network, can be used as if they were running on a local machine.

The core components of the OSF/Motif technology include an extensible user interface[17], an applications programming interface[18], a user interface metalanguage[19], and a window manager. Motif is based on the X Intrinsics, a toolkit framework provided with X11. The Intrinsics use an object oriented model to create a class hierarchy of graphical objects known as *"widgets"*.

Both X11 and Motif, are extremely helpful but their libraries are complex to learn and to use for programming. Coding of applications started initially with those libraries, and demanded substantial efforts in becoming familiar with this new technology.

User Interface Management Systems[20] (sometimes called interface builders) are the tools which help the application programmer to design the user interface part of the application interactively. A UIMS is generally composed of a graphic oriented editor and a code generator, and sometimes other complementary tools. There are several Motif compliant UIMS available. A few of them have been tested at ESRF. At present one of them has been selected for use[8].

This UIMS drastically eases now the design of Motif-based user interfaces. It generates stand-alone C code and/or a combination of Motif-compliant C and UIL code.

Synoptic drawings with selectable objects are scarcely supported by the above mentioned tools. We therefore work on an implementation of a Motif compliant widget that uses vectorial drawings generated by PHIGS[21]. Synoptics will be generated by CAE systems like AUTOCAD or EUCLID.

6.3   Database

The control system data are stored in relational databases which manage two logical parts:

- Resource data;
- Runtime data.

The resource database keeps permanent data. Examples are: start-up resources, calibrations, equipment definitions, installation- & maintenance data, etc...

---

[17] UI
[18] API
[19] UIL
[20] UIMS
[21] stands for Programmer's Hierarchical Graphics System and is an ISO standard

The implementation of the resource database uses ORACLE and its powerful set of development tools. Modifications on the resource database cause automatic update of runtime data sets, that are redundant copies of the *parent data set* in the resource database.

The runtime database is a central warehouse for all sorts of temporary or transient data. It is not a medium for permanent storage. It is simply a front for permanent storage. Only *memory resident* database systems can meet the demands for sufficiently short transaction times. A prototype of the runtime database is operational and uses a *Real-Time Database Base Management System*[22] available on HPs.

The runtime database can be used to alleviate congestion problems. Multiple processes can update data asynchronously in the database. Other processes can retrieve this information aynchronously without blocking the process doing the updating. Since the memory resident database profits from a much higher than normal I/O throughput, this mechanism is used to resolve information traffic jams that may occur.

The runtime database's prime source is a so-called *Update Daemon* that updates the current machine status if a particular client requests this. The database contains ring buffered tables to store brief histories of the results of requests issued to a device. Although physically the runtime database is distributed over all process servers, access to it is transparent to database clients. On-line data can be archived continously. Only a time window of some five minutes is kept in memory by RTDB, the rest of the data is dumped into the disk-based ORACLE database. An index to these data is constructed to allow *queries* in accelerator physics terms on archived data. Data can be stamped with time or accelerator status information.

The same runtime database can also be used by applications as a mean for interprocess communication. Applications dynamically allocate "tables" of formatted data, that can then be piped or multiplexed to other applications.

The volume of data that will be managed by the resource database is estimated to be some 10Mbytes. The whole control system comprises more than 3000 devices and more than 50000 static resources. The throughput of on-line data at its worst is estimated to be some 40kbytes/sec. If all data coming from the machine is stored at an interval of a second, it would mean 12Mbytes every 5 minutes.

6.4   Applications

Application program development at ESRF has been taken care of at an early stage of the project. Usually this class of software tends to be too little and too late. To give accelerator physicists the possibility to develop their applications in parallel with the control system software, an applications programmer interface[23] has been defined very early and kept stable until then.

---

[22] called RTDB
[23] API

	NFS/RPC		API
	UDP	TCP	UDP
open connection	20-25ms	35-45ms	55-65ms
close connection	0.1-0.2ms	0.3-0.5ms	10-20ms
RPC with 100bytes	10-15ms	15-20ms	15-20ms
RPC with 8kbytes	25-35ms	55-65ms	30-40ms
RPC with 40kbytes		220-250ms	

Table 2: Performance figures for RPC and API

Access to the Device Servers is provided by a small set of C calls. These calls allow the users to develop their applications in peace without being affected by what goes on in the Device Server software. Initially very simple Device Servers have been written, that ran on the local host, and that only *simulated* devices. These API-calls hide the complexity of Device Servers and their implementation from users by offering them a set of high level commands as access method. *How* and *where* the Device Server executes the high level command is hidden. In the distributed environment this workload is spread over a number of machines.

The following functions form the basis of the Device Server API:

- `int dev_import(name, access, ds_ptr, error)`
  ```
  char *name;          /* Device Server Name */
  long int access;     /* Access Type */
  devserver *ds_ptr;   /* DevServer Handle */
  long *error;         /* Error Code */
  ```
  Is called by the application to establish a connection to a Device of the specified name.

- `int dev_putget(ds, cmd, argin_ptr, typein argout_ptr, typeout, error)`
  ```
  devserver ds;          /* DevServer Handle */
  DevCmd cmd;            /* Device Command */
  DevArgPtr argin_ptr;  /* Call Parameter */
  DevType typein;        /* Parameter Type */
  DevArgPtr argout_ptr; /* Return Parameter */
  DevType typeout;       /* Parameter Type */
  long *error;
  ```
  Is called by the application to execute a command on the device. This is a "blocking" call which doesn't return until the command requested has been executed.

- `int dev_put(ds, cmd, argin_ptr, typein, error)`
  Is called by the application to execute a command on the device. This is an "asynchronous" call which will return as soon as the command has been delivered to the server or an error occurred. This call can only be used to start a command, no knowledge is returned about its execution and/or success. It is up to the application to interrogate the Device Server to determine its status.

- `int dev_free(ds, error)`
  ```
  DevServer ds;   /* DevServer Handle */
  long *error;
  ```
  Is called by an application to release a device properly.

Measurements of RPC and API performance that we achieve between a process server and a VME node are given in table 2.

Using strictly this device access interface and the X11 and Motif standards for interactive graphical I/O, an impressive number of physics applications have been developed in parallel with the basic control system software, and have considerably helped to commission the booster in due time. An enumeration of applications presently available follows:

Transfer line 1 & 2: These programs execute specific procedures for step by step alignment, emittance measurement, and modelling of beam envelopes.

Closed orbit: The program performs basic control of steerers and bumps, beam position readout, orbit plots, fourier analysis of orbit and steerers, and automatic orbit correction.

Booster vacuum: This program controls the vacuum system. It allows individual device control, display of periodically updated status, and display of pressure profile.

Booster injection/extraction: This program allows control of current- and timing settings of pulsed injection/extraction elements.

Booster optics: Different options in this application allow tune measurement/setting, chromaticity measurement/setting, measurement of β-functions at quadrupole locations, and measurement of dispersion (η-function).

Storage ring injection: Used for tuning of the injection kicker/septa to maintain an injection bump and control injected beam position and angle.

7 Conclusion

The ESRF control system is operational since August 1991. It played an important role during commissioning of the booster synchrotron. The system has been designed from bottom up, using object oriented programming techniques, and is based on proven industry standards. Its design has been guided by a clear preference for mature commercial systems over custom- or home-made ones, without formally excluding the latter.

The system is not finished yet, but it is easily extendable and adaptable to future needs. It is through the standards that have been selected for the control system, that ESRF will be able to migrate together with industry to new technologies while preserving considerable investments in hard- and software. The choices of UNIX, X11/Motif, VME/OS9, Ethernet, and TCP/IP have been fundamental in this sense.

8 Aknowledgements

9 References

References

[1] J.L. Laclare, "Overview of the European Synchrotron Light Source" in *IEEE Particle Accelerator Conference*, Washington, D.C., March 1987, pp. 417-421.

[2] W.D. Klotz and C.Hervé, "The Conceptual design of the ESRF Control System" in *European Particle Accelerator Conference*, Rome, June 1988, pp. 1196-1198.

[3] *Proceedings of Europhysics Conference on Control Systems for Experimental Physics*, Villars-sur-Ollon, Switzerland, Sept.28 - Oct.2, 1987.

[4] *Proceedings of the International Conference on Accelerator and Large Physics Control Systems*, Vancouver, BC, October 30 - November 3, 1989.

[5] Pal S. Andersen, V. Frammery, and R. Wilcke, "Tools for Remote Computing in Accelerator control" in *Proceedings of the International Conference on Accelerator and Large Physics Control Systems*, Vancouver, BC, October 30 - November 3, 1989, pp. 225-230.

[6] *Network Programming Guide*, SUN microsystems, Part Number: 800-3850-10, Revision A of 27 March, 1990.

[7] A. Nye and T. O'Reilly, *X Toolkit Intrinsics Programming Manual*, O'Reilly & Associates, Inc., 1990.

[8] *The Builder Xcessory User Guide*, Cambridge, MA: Integrated Computer Solutions, Inc. 1990

Trademarks

Centralized Multiprocessor Control System for the Frascati Storage Rings DAΦNE

Giampiero Di Pirro, Catia Milardi, Mario Serio, Alessandro Stecchi, Luciano Trasatti.
INFN, Laboratori Nazionali di Frascati
P. O. Box 13, 00044 Frascati (Rm) ITALY, e.m. trasatti@irmlnf
and
Barbara Caccia, Vittorio Dante, Raffaellino Lomoro, Eleuterio Spiriti, Stefano Valentini.
Istituto Superiore di Sanita' and INFN, Sezione Sanita',
Via Regina Elena, 299, 00161 Roma, ITALY, e.m. caccia@sanita.infn.it

Abstract

We describe the status of the DANTE (DAΦne New Tools Environment) control system for the new DAΦNE Φ-factory under construction at the Frascati National Laboratories. The system is based on a centralized communication architecture for simplicity and reliability. A central processor unit coordinates all communications between the consoles and the lower level distributed processing power, and continuously updates a central memory that contains the whole machine status. We have developed a system of VME Fiber Optic interfaces allowing very fast point to point communication between distant processors. Macintosh II personal computers are used as consoles. The lower levels are all built using the VME standard.

I. DAΦNE

DAΦNE [1] is a two ring colliding beam Φ-Factory under construction at the Frascati National Laboratories (See Fig. 1).

Construction and commissioning is scheduled for the end of 1995.

The luminosity target is $\sim 10^{33}$ cm^{-2} sec^{-1}.
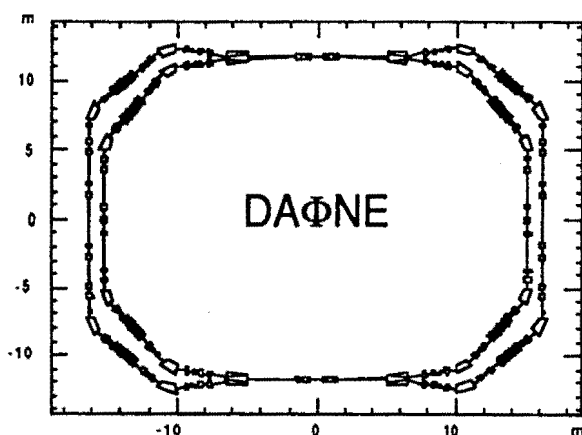
Fig. 1 : The DAΦNE Φ-Factory layout

II. SYSTEM STRUCTURE

Fig. 2 shows the general architecture of the control system. Three levels are defined:

PARADISE (PARAllel DISplay Environment) is the top level, implementing the human interface. Several consoles, built on Macintosh personal computers, communicate with the rest of the system through high speed DMA busses and fiber optic links.

PURGATORY (Primary Unit for Readout and GATing Of Real time Yonder) is the second and central level of the system. It essentially contains only a CPU and a Memory in a VME crate. The CPU acts as a general concentrator and coordinator of messages throughout the system. The central Memory is continuously updated and represents the prototype of the machine database.

HELL (Hardware Environment at the Low Level) is the third level of the system and is constituted by many (about 60) VME crates distributed around the machines.
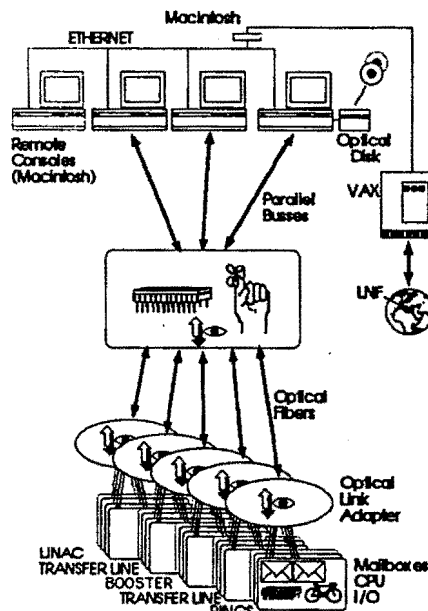
Fig. 2: Control System Schematic Diagram

A CPU in every crate performs control and information hiding from the upper levels.

VME is used throughout Purgatory and Hell

A first estimate of the system gives about 7000 channels to be controlled.

Centralized Communication Control

We have chosen an architecture based on a single central

controller of communications instead of the usual network for reliability and performance.

A system with a central CPU controlling all the others through high speed point to point links and a polling mechanism is much easier to implement and to maintain than a network.

Data integrity is easily achieved through a write-readback mechanism and the failure of a peripheral unit can be diagnosed and isolated very efficiently.

Performances are much better than usual networks, due to link speed and protocol simplicity: in a previous paper[3] we measured:

1700 messages/s from Paradise to Hell

10 μs polling time for Purgatory to check out a Hell CPU

450 KByte/s data transfer from Hell to Paradise.

In this architecture the link between the consoles and the central coordinator (Purgatory) is easily implemented through high speed DMA busses. The same is not true for the links between Purgatory and Hell, since the peripheral CPUs are spread over a wide geographic area, well above the standard 70m allowed by DMA busses. The best solution for these links is to use fiber optic connections, with their high bandwidth and noise immunity.

III. OPLA' (OPTICAL LINK ADAPTER)

The OPLA' (OPtical Link Adapter) project for an interface between a standard third level VME crate, and an optical fiber has been developed. This project aims at realizing a multipurpose system for fast data transfer over long distance.

Use of the AMD Taxi chips allows data rates of up to 160 Mbit/s, which is more than a standard CPU can transmit on a VME bus.

A simple architecture will be implemented: a 16 bit word presented to the transmitter will be stored on the other side of the link in a 2048 word FIFO. FIFO overflow will be automatically prevented by back transmission of a FIFO-full status message.

A first prototype board has been developed and tested. The board can be divided into two sections:

i) Tx/Rx toward the optical fiber;

ii) VME interface.

Software for controlling and testing the board has been developed on a Macintosh IIfx using LabVIEW®[6]. LabVIEW® allows to create a front panel that specifies inputs and outputs providing the user interface for interactive operations. Behind the front panel there is a block diagram, which is the executable program.

A panel for the preliminary tests on the OPLA' prototype boards has been built. This panel allows access to two VME boards connected through optical fibers.

To access to VMEbus a MICRON (Mac Vee Interface Card Resident On Nubus)[2], developed at CERN and a MacVee (Microcomputer Applied to the Control of VME Electronic Equipment) are used.

The next step of the project will be to implement four complete Tx/Rx sections on a single board. In fig.3 the schematic design of the board is reported. Gate arrays will be used to reduce component count and therefore design time and

number of required boards. For each section a Xilinx programmable gate array will implement the control of data transmission and the receive logic. A single gate array will implements the interface toward VME.

The AMD Taxi chips are used to implement the interface toward the optical fiber, due to their ease of connection and high integration.
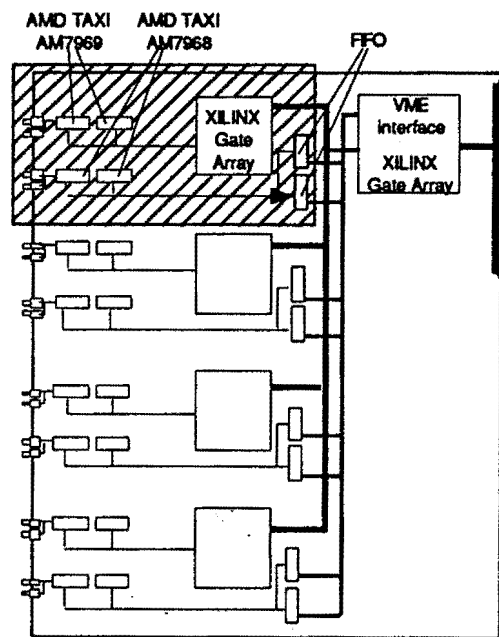
Fig. 3 : OPLA' schematic diagram

IV. CONSOLES

Macintosh personal computers have been chosen for the system consoles. In the last few years we have seen an impressive effort by personal computer firms and third parties to supply large quantities of very high quality software at very low prices. The situation is now, as far as software is concerned, definitely in favour of the use of large diffusion machines as opposed to high cost, "high" power, low diffusion workstations. Hardware prices keep getting lower, while the cost of software development has reached about 80% of the total cost of an installation, with all the reliability risks of in-house software development. The Macintosh family of computers is at the moment the best candidate for a human interface development, since the effort expanded on software development on this machine has been the most striking on the market.

Previous experience with Hypercard [4,5], on the other hand, has shown that high level software packages can decrease software development times by strong factors. Faster and more powerful human interface packages are coming out every day.

We already mentioned LabVIEW®: it is the first large diffusion software package specifically designed for data

acquisition and controls. Its main features are:

- Very easy creation of "virtual instruments", i.e. human interface panels containing controls and displays, acting on the appropriate hardware interface;

- Graphic development of programs through an "icon" language.

- Large scientific library containing frequently used tools (histograms, fourier transforms, etc.).

We are convinced that the use of this kind of large diffusion packages will allow us to save a very large amount of time and effort, not only in program development, but mainly in debugging and maintenance.

Remote Consoles

A good example of the above is the problem of remote consoles. Using an Ethernet or LocalTalk link and a commercial program, Timbuktu® [7] it is possible to gain complete control of a remote Macintosh, under the protection of a system of passwords. This was a specific request for the DAΦNE control system. How long would it have taken to build and debug such a facility?

V. VME OPERATING SYSTEM

In our previous experience with a similarly structured control system we used no operating system for the lower level CPUs. Simple FORTRAN or C programs took care of the relatively easy tasks of a small and dedicated CPU that only has to perform a few simple tasks. The general idea is still:"A CPU for each task". While this is a rather extreme statement, we think that the software environment for the lower level CPUs must be kept as simple as possible, at the expense of increasing their number. On the other hand, the advantages of using a standard environment are obvious as far as bookkeeping and standardization are concerned. At the moment we are evaluating several Real Time Kernels and we plan to reach a decision by the middle of next year.
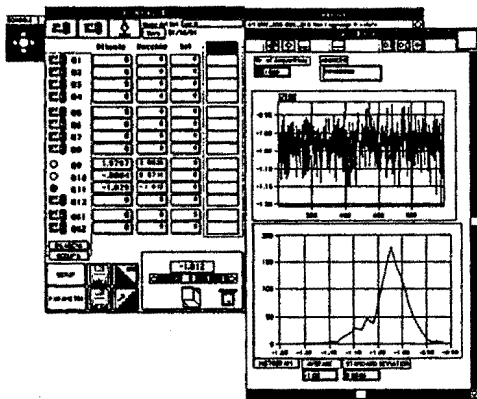

Fig. 4 : Field tests showing progressive migration from Hypercard to LabVIEW

VI. FIELD TESTS

We are testing out these ideas on small accelerators in Frascati. A first implementation on a set of steering coils at ADONE showed the feasibility of Hypercard as a human interface tool, at least for small systems.

Later, on a small machine, LISA, we have tried out the three level system and we have started migrating the human interface, originally written in Hypercard, to LabVIEW (see fig. 4). We have shown that a progressive migration is feasible, and these tests will allow us to measure the real performance of these software packages in the field. At the moment we believe that LabVIEW will prove adequate even for the large DAΦNE control system.

VII. SUMMARY

The control system we are building is based on highly distributed hardware and software capabilities, with a strong accent on openness to other environments. We believe that the human interface will be the most arduous problem to solve, and that the use of high diffusion software packages can be a big help in that direction.

A high speed fiber optic link adapted to the accelerator control environment is being developed.

VIII. ACKNOWLEDGEMENTS

We would like to thank the Accelerator Group of the LNF for continuing discussions and encouragement. The work with the LISA group helped us develop a set of techniques that will be very useful in the future.

IX. REFERENCES

[1]   G. Vignola et al., talk presented at the San Francisco Particle Accelerator Conference, May 1991.

[2]   B.G. Taylor, "The MacVEE Hardware User Manual"; Rev.4.1, CERN EP Division, 1987.

[3]   L. Catani, G. Di Pirro, C. Milardi, L. Trasatti, F. Giacco, F. Serlini "The High Speed Bus System for LISA" Presented at the International Conference on "Accelerator and Large Experimental Physics Control System" Vancouver, Canada, Nov 3, (1989) *In Press to Nuclear Instruments and Methods*.

[4]   L. Catani, G. Di Pirro, C. Milardi, A. Stecchi, L. Trasatti.:MARCO - Models of Accelerators and Rings to Commission and Operate, presented at the San Francisco Particle Accelerator Conference, May 1991.

[5]   HyperCard v2.1, Apple Computer Inc, © 1987-90.

[6]   LabVIEW, National Instruments Corp., © 1990.

[7]   Timbuktu, Farallon Computing Inc, © 1987-91.

THE OPERATOR VIEW OF THE SUPERCONDUCTING CYCLOTRON AT LNS CATANIA

D.GIOVE, INFN MILAN
LASA - V. F.LLI CERVI 201
SEGRATE- ITALY
G. CUTTONE AND A. ROVELLI, INFN LNS
V.LE A.DORIA ANG. V. S. SOFIA
CATANIA-ITALY

Abstract

The upper level of a distributed control system designed for the Superconducting Cyclotron (SC), will be discussed. In particular, we will present a detailed description of the operator view of this accelerator along with the tools for I/O points management, data rappresentations, data archiving and retrieval. A dedicated program, developed by us, working under X-Window will be described as a starting point for a new man-machine interface approach in small laboratories opposed to the first industrial available packages.

I. INTRODUCTION

The SC developed at the Milan University, where was performed the first test on the magnetic field, has been moved to the final destination, at LNS in Catania. The work on the accelerator will start at the begin of the next year with the magnet excitation and the installation of the RF cavities. The extraction of the beam, injected in the booster by a 15 MV Tandem, is foreseen before the spring of 1993. The main features of this heavy ions accelerating facility are reported elsewhere [1] [2].

According to the experience gained in Milan on the control system during the first magnetic measurements, we planned improvements mainly on the upper level devoted to the man-machine interaction. The first console designed and realized in 1985-1987 [3], followed an old philosophy. Altough the main hardware and software choices had been proved satisfactory, it gave us a flexibility not so good as we expected.

Nowadays the availability of standard graphic software and the capability to create networks are the two features which make the workstation a pratical cost effective way to provide an universal environment for the development of the operator interface. It is possible to use powerful hardware and software standards which make straightforward the setting up of a network where hardware and software resources can be easily shared in a really efficient environment.

In the follow we will discuss the hardware and software architecture of the Superconducting Cyclotron operator console together with its performance measured during the test.

II. THE OPERATOR CONSOLE

In 1989, during the shut-down of the SC in Milan, we decided to review the structure of the console. Some general rules were fixed for the project.

- The architecure must be independent of the number of worksites in use: the insertion or the removal of a worksite must be invisible to the whole system, realizing in this way a real "easy expandible system".

- The architecture would allow to have the same graphical workstation conncted both in the main control room and in a remote place closed to the accelerator.

- The architecture of the console must be fully independent of the lower levels so that the choices made in process and plant levels don't influence the supervisor structure.

- The presentation level of the software must not require any practice in computer science and must be picture driven. The operator must be able to define its own working environment with few choices and the access to every information that he wants to deal with has to be guaranteed.

- The operations on an accelerator subsytem must be possible by each workstation but not at the same time. The display of all machine parameter must be possible on different workstations at the same time.

- The on line software configuration must be guaranteed by means dedicated programs taking advantage of a database.

- The allarms and malfunctioning logging task must be managed by a dedicated unit able to provide particular tools to help the operator in his trouble shooting job.

- The maintenance of the whole structure must be easy and centralized as much as possible on a single machine.

It was decided that the development of most of software would take not more than 3 man-years of work. The choice of the final solution was not easy and a lot of different considerations, like our experience with graphical workstations and their operating system, the estimated technical support avilable from vendors in our country, were taken into account. At last, we decided to implement our hardware architecture on a Local Area VaxCluster (LAVC) of 3100 Vaxstations with a μVax 3100 as boot member. A gateway was provided towards the lower levels. A μVax II was dedicated to this task along with the storage of the memory map of all sensors and actuators. Two 80386 PCs were dedicated to allarm logging and to manage the data necessary for the application tasks.
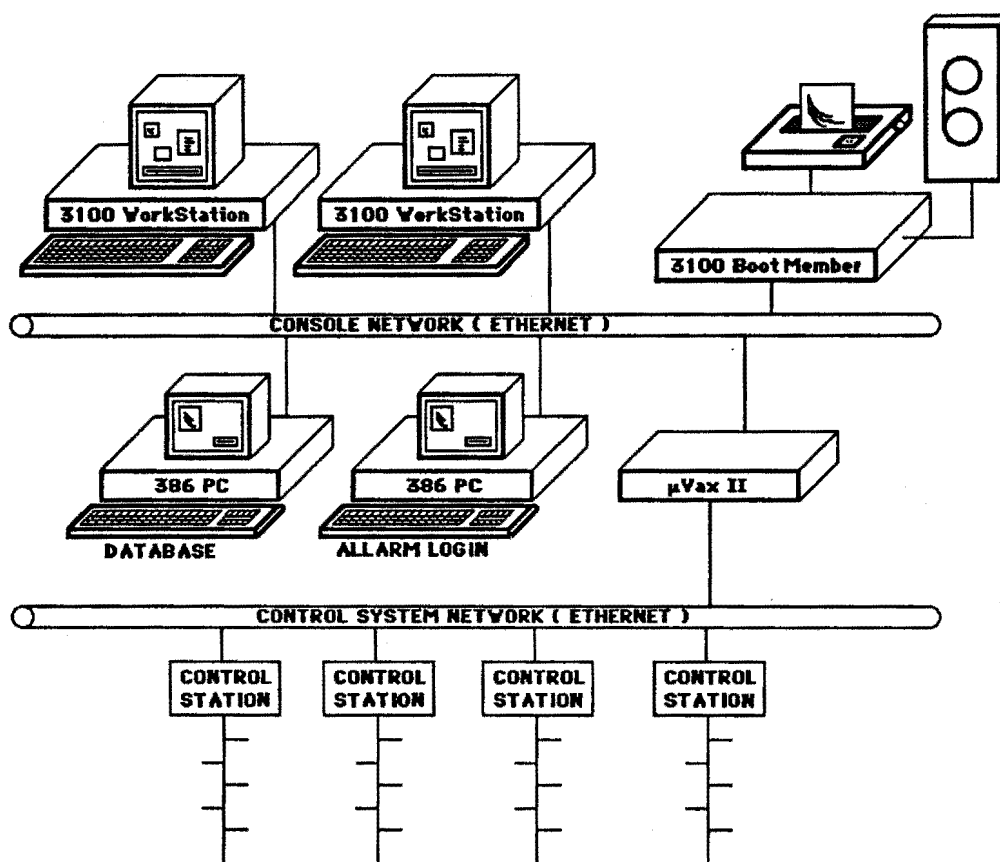
Fig. 1 The hardware architecture of the control system.

III. HARDWARE ARCHITECTURE

The main console has been designed as an Ethernet segment with a distributed software running on the resources which are connected to it. The Ethernet segment is physically the same which interconnects the process stations but we have superimposed on it the LAVC. This software is quite reliable and is well tested in a lot of installations all around the world. The structure resulting is easy to mantain as it recalls centralized architectures but it is really flexible and able to suit growing requirements. The intrinsic problem to have a central machine which will paralize the whole system in case of failure has been considered but our experience gained in similar architecture used for computer rooms reports a very low failure rate.

The boot member is dedicated to the management of Cluster operations and of print and tape resources. Historical logging of all the accelerator parameters acquired or calculated by the control system is provided by the μVax II, which is a satellite in the LAVC. Powerfull 3100 VaxStations with 16" Trinitron colour monitor have been chosen as the hardware platform for the operator interaction tools. All the workstations, whose number is completely unrelated to the architecture of the console, share the same programs and are able to work on every acces point to the coaxial Ethernet cable. This is the true sense of statement that: "the console is a network". A workstation will be dedicated to the beam dynamic simulation either in accelerator or in the beam transport lines. This machine, working on a complete accelerator or beam lines setting, will calculate the necessary corrections for the beam optimization.

Two particular nodes on the control room Ethernet are two 80386 based PCs, chosen because of their high performance, low cost, easy programming and full integration in the Digital network architecture (DECNET). The first is dedicated to run a database application (INFORMIX) used for the storage of all relevant parameters of each sensor and actuator of the cyclotron, like its name and software position in the control system, the different alarm thresholds and the range of possible setting value. Data are organized according to different query schemes to provide an interactive tool for everyone needs a particular information. A particular use of this application is the console itself. Infact, the data section of the applicative programs refers to this database and each variation is immediately available to the operator workstations taking advantage by the possibility in a DECNET environment to share disk between DOS and VMS

machines. The second PC provides a lot of tools to help the operator during alarm handling and troubleshooting operations. This computer was conceived like an hypermedia machine, with enhanced graphic capabilities for the display of pictures or drawings related to particular event and with the possibility to play back defined speeches as an auxiliary tools. Fig. 1 showes the hardware architecture of the control room Ethernet so far described.

IV. SOFTWARE ARCHITECTURE

The most relevant challenge in the development of a new, flexible and portable console is in the software design. The whole hardware architecture described in the previous section would be meaningless if it would not be provided an adequate software support.

The main choice in the software architecture has been to use X-Window as basic development environment. Taking as a reference the software model proposed by X-Window, a modular object oriented code has been developed: GIULIA (Graphical Interactive Unit Level for Improved Automation). It is based on DEC-Windows, an extension of X-Window developed by DEC, which permits to use a powerfull toolkit and a User Interface Language (UIL) able to define the structure of the graphical interface. The Object description is stored in a separate file where the methods to which the objects will respond are outlined. The necessary code for the implementation of every method described in the UIL file is contained in a second file. Such a structure enhances the seperation between form and content of an application. The possibility to use low level Xlib functions together with DEC calls in the same code is guaranteed. We chosed to work under VMS and to use the "C" language to develop our code.

GIULIA is composed by two main parts running on at least two different computers: the first provides man-machine interaction tools while the second is dedicated to the management of the accelerator data received from the control stations and guarantees the operations on an accelerator subsytem by only a workstation at a time. GIULIA creates a remote task on the μVAX II devoted to the data exchange with the control system, in order to realize a trasparent task to task communication scheme. Data exchange has been optimized to reduce the traffic on the network and to semplify the handling of the shared memory. For each control station devoted to a particular accelerator subsystem according to a functional scheme of intelligence distribution, we have on the μVAX II two different VMS global sections where are written respectively the accelerator data received and the command to be sent. The access to these sections is controlled by means of flags. Tasks running on remote workstations look and actuate on the control hardware in an indirect way by means of this sections, making the software really device independent. Full operations on each accelerator subsystem are possible by each workstation asking to the μVAX II the allocation of this resource. At the end the operator must deallocate the resource to permit the operation by an other machine. The bandwidth measured on the network channel is of nearly 1.1 Mbit/sec for 2.5 Kbytes packets quite in agreement with the performance of other control networks. In this way we can send by a workstation up to 35 commands for second, receiving any time by the control station all the data acquired at that moment.

The interaction tools in GIULIA are based on the defintion of three different classes: rappresentation, interaction and BPM. Main attributes of these classes are "true" resizing and iconic interface. Rappresentation has three subclasses (table, bar chart and graphic) that identify the different ways to display every parameter of the cyclotron. Interaction has three subclasses (button, slider and knob) that identify the different ways to operate on each parameter of the cyclotron. A particular class is BPM which is realized to display the reconstructed beam shape acquired by the beam diagnostic devices, with a low frequency. The environment provided to the operator for its job is like that one of a writing desk: foils, particular rappresentations, are distributed on the screen and the parameters to be displayed or actuated have been chosen during a short navigation inside the program by the operator itself according to its preference. Foils may be opened, iconified, deleted and reconfigured. Graphic attributes of the subclasses may be easily changed by interactive setup utilities or modifing the UIL description file.

A process in GIULIA checks all variables for allarms or safety limits, and when a threshold is exceeded all the related informations are transferred to the second PC for further dedicated analysis. GIULIA provides an extensive on-line context sensitive help explaining how it works and all the details concerning the objects classes which it implements. Some choices, which may be dangerous as the modification of an allarm limit by the operator, are always recorded in the display of the workstation and are effective only for that related display.

V. CONCLUSION

The goal to develop the software so far discussed has been reached. Preliminary test has been carried out in order to verify the overall architecture and its performance that seems to be quite in agreement with the accelerator requirements.

In order to implement a real time picture of the beam a dedicated system is under development based on a CCD camera and a VME frame grabber board. A dedicated monitor will display the images so acquired at a high rate (about 30 Hz) with the possibilty to show up to 4 different images at the same time.

VI. REFERENCES

[1] E.Acerbi et al. "Progress Report on the Heavy Ion Facility at LNS" in EPAC Proceedings, Nice, France, June 1990, pp. 425-427.

[2] L.Calabretta et al. " The Tandem Injector for the Superconducting Cyclotron at LNS" in EPAC Proceedings, Nice, France, June 1990, pp. 497-499

[3] F. Aghion et al. "The Operator Interface in the Control System for the Milan Superconducting Cyclotron" in 11th Int. Conf. on Cylotrons and their Appl. Proceedings, Tokyo, Japan, 1987, pp. 432-435

The UNK Control System

V.N.Alferov, V.L.Brook, A.F.Dunaitsev, S.G.Goloborodko, S.V.Izgarshev, V.V.Komarov, A.F.Lukyantsev,

M.S.Mikheev, N.N.Trofimov, V.P.Sakharov, E.D.Scherbakov, A.I.Vaguine, V.P.Voevodin, V.D.Yourpalov,

S.A.Zelepoukin, IHEP, Protvino, USSR, B.Kuiper, CERN, Geneva

Abstract

The IHEP proton Accelerating and Storage Complex (UNK) includes in its first stage a 400 GeV conventional and a 3000 GeV superconducting ring placed in the same underground tunnel of 20.7 km circumference. The beam will be injected into UNK from the existing 70 GeV accelerator U-70. The experimental programme which is planned to start in 1995, will include 3000 GeV fixed target and 400*3000 GeV colliding beams physics. The size and complexity of the UNK dictate a distributed multiprocesssor architecture of the control system. About 4000 of 8/16 bit controllers, directly attached to the UNK equipment will perform low level control and data acquisition tasks. The equipment controllers will be connected via the MIL-1553 field bus to VME based 32-bit front end computers. The TCP/IP network will interconnect front end computers in the UNK equipment buildings with UNIX workstations and servers in the Main Control Room. The report presents the general architecture and current status of the UNK control.

1.Introduction

The UNK complex will combine - in one tunnel of 20.7 km circumference - a 400 GeV conventional magnet synchrotron (UNK-I) and a 3000 GeV superconducting synchrotron/storage ring (UNK-II). At a later stage a second superconducting ring (UNK-III) may be added with the aim of doing proton-proton collider physics at 6 TeV (Figure 1).

The UNK-I is injected at 70 GeV from the existing proton synchrotron U-70. For one filling up to 12 pulses from U-70 may be stacked, accelerated to 400 GeV and transferred to the UNK-II which in turn accelerates them up to 3 TeV.

Three main modes of operation are presently foreseen.

1. Fixed Target at 3 TeV: fast or slow extraction will send the 3 TeV beam to the fixed target experimental area. During the acceleration in the superconducting ring, U-70 may produce beams for its own 70 GeV experimental area.

2. Colliding Beams at 3 + 0.4 TeV: the beams from the UNK-II and UNK-I are made to collide. For this the UNK-I is operated first as booster and, after field reversal, as a storage ring run at 400 GeV.

3. Colliding Beams at 3 + 3 TeV: the UNK-I will first inject into one superconducting ring and, after field reversal, into the second one.
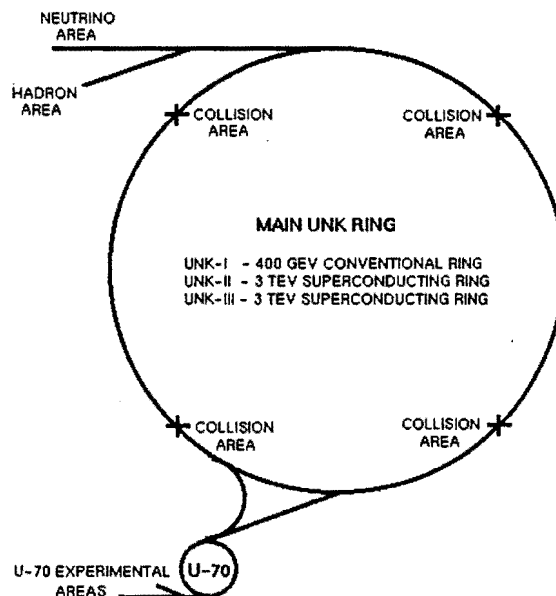
Figure 1 Layout of the UNK Complex

The accelerator controls equipment will be distributed over 24 on-surface buildings situated mainly along the accelerator ring. In one of the these is the Main Control Room (MCR), the other ones house the remote nodes of the control system. The latter are totally controlled from the MCR and are in general not manned. The typical distance between any two adjacent buildings is about 1.8 km and the maximum is about 3.5 km.

The more than 3500 superconducting magnets require a cryogenic plant and elaborate distribution, recovery and safety installations and their concomitant controls in the surface buildings around the ring tunnel.

The UNK operation is supported by general electricity and water distribution networks, tunnel ventilation, radiation protection, fire safety and other utilities which require highly reliable controls with 24 h/day 365 d/year availability.

The secondary beamlines and external experimental areas cover an area of roughly 12 km length. Controls for their

equipment may follow closely the principles of the accelerator controls.

The upgrade of U-70, for meeting UNK injector specifications, requires intensive machine studies which in turn make a controls upgrade mandatory. Since this must precede UNK, the principles and equipment may differ somewhat from UNK controls proper.

2. HARDWARE ARCHITECTURE

The size and complexity of UNK, together with real time and other requirements, dictate a multilevel multiprocessor controls architecture (see figure 2).

Various components of the accelerator equipment are driven by more than 4000 equipment controllers (EC) which perform low level control and data acquisition tasks in hard real time and provide a uniform equipment representation for the upper levels of the control system.

A typical EC is Euromechanics crate with Multibus I compatible backplane bus, contains a single board microcomputer, a fieldbus interface and a number of equipment specific I/O cards. The standard EC microcomputer is based on microprocessors similar to the Intel 8086/8087. In the UNK-I control system about 800 of such ECs will be used for beam instrumentation, power suplies, RF and vacuum system controls.

Similar technology will be used in the UNK-II and cryogenics complex equipment interface. The main difference is in the EC microcomputer type which in this case is based on the LSI-11 compatible microprocessors (see chapter 4). There will be about 1300 of such ECs in the UNK-II and cryogenics complex controls.

About 1500 of the UNK-I correction magnet power supplies will be controlled by embedded ECs based on the Intel 8051 microcontrollers. The EC is implemented on two standard Eurocards and performs all the power supply control functions, including timing and function generation. It has direct interfaces to the MIL-1553 fieldbus, timing and fast alarm systems.

A general timing system distributes reference events and clock trains to all ECs. A separate alarm and interlock network collects signals from all ECs monitoring vital accelerator subsystems. These signals may be used to trigger the beam abort system and inhibit beam injection.

The next higher level of control is represented by the front end computers (FEC) spread around the main UNK ring and beam transfer lines and interconnected with the upper level computers by the UNK controls network. The FECs will drive EC clusters through the 1 Mbit/s MIL-STD-1553 serial multidrop bus, which provides a cheap solution on a standard chip, noise immunity and galvanic insulation. Physically, the FEC is a modular board assembly in a standard VMEbus crate. The basic FEC configuration will consist of a 32-bit processor board, a network interface and a number of MIL-1553 bus controllers. It may also include I/O modules for direct interfacing to equipment needing full network functionality or/and bandwidth to handle high data rates (e.g. for sophisticated beam diagnostic devices).
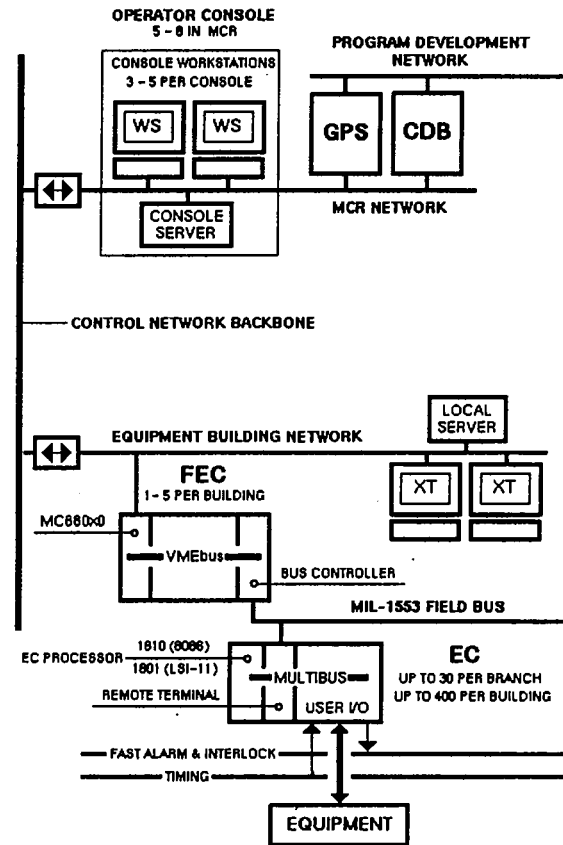
Figure 2 General Hardware Architecture

The FEC's main purpose is providing access for the upper level control software via network to the ECs. It may thus be considered as a gateway linking the MIL-1553 field bus to the UNK controls network and as a specific kind of network server, providing a set of equipment access services for application tasks running in the networked computers.

In addition to this general task, a number of FECs may be dedicated to certain functions through their own set of I/O modules. For example, one presently considers dedicating certain FECs to a task of UNK-II superconducting magnet main power supplies control and quench protection. Finally, a FEC may run some application tasks, in particular for local closed loop control and local equipment access, test and diagnostics. It will also perform ECs downloading and surveillance via the MIL-1553 field bus.

The network will be layered: a so-called backbone will interconnect the buildings and a number of LANs will interconnect equipment at the MCR and inside other buildings. The development infrastructure forms a sub-network which will be attached to the backbone. LANs will mostly be Ethernet. The backbone will be fiber optics FDDI or 16Mb fiber optics Token Ring. The LANs and backbone will be connected by bridges/routers. Most networking hardware will be standard commercial products.

The UNK operator's consoles, 5-8 in total, will each be equipped with 3-5 graphics workstations and one server. The latter will provide common services (file, print, plot, etc.) and can also be used to execute certain run-time application software.

There will be a dedicated UNK control system's data base management server CDB and a general purpose server GPS for number crunching, modelling. The latter also caters for general user program development, thus supporting numerous workstations and terminals spread around the buildings.

3. SOFTWARE ARCHITECTURE

Experience in development of accelerator control system software packages world wide shows that certain functionalities are existing in one form or another in the majority of them. The general trend, which we try to follow, is therefore to extract these common parts and provide them as standard facilities which may be used by each *specific application* task. This would allow to eliminate multiple code and improve system reliability and maintainability. The software implementation of those common functionalities may be called the *application environment* (Figure 3).
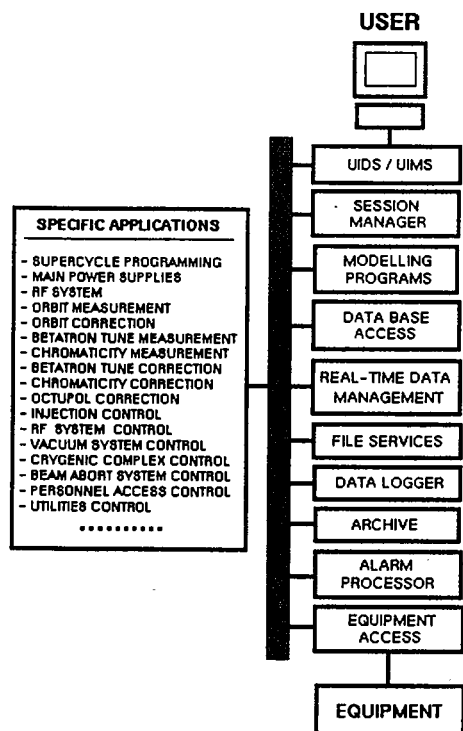

Figure 3 Specific Applications and Application Environment

In contrast with a general purpose application independent system software the application environment is problem (controls) oriented. It is supposed to be relatively stable and only evolve slowly during a control system's life cycle.

Application Software

A first analysis of the UNK control system functionality has been made using the SASD methodology. Thus the main procedures, data flows and data stores were identified. The application software functionality can be represented by the following broad groups of tasks.

1. Process modeling and preparation of data sets which determine the operation of accelerator devices. Advanced modeling programs can simulate a particle behavior for the different modes of operation of the accelerator and in such a way test a validity of the data set prepared.

2. Data down-loading, acquisition and trimming.

3. Routine surveillance of the technological accelerator subsystems (like electricity, vacuum, etc.) Alarm signal generation and processing, fault recovery.

4. Data logging and archiving services which allow to keep working track and history of the accelerator operation.

This analysis, design and data structure development will be pursued using a modern integrated CASE tool package. The latter should also provide project management and documentation support, which are important when numerous programmers of different level must cooperate.

System Software

The application environment is built on the basis of elementary functions provided by the system software. The general UNIX operating environment has been chosen for the UNK control system.

The UNIX (Ultrix) will be used in the operator workstations and servers while a Unix-like real-time sytem will be chosen for FECs. For this part the LynxOS from the Lynx Real-Time Systems Inc. is in the process of evaluation.

The TCP/IP, RPC, NFS packages, now a standard features of most Unix and real-time systems, will be used in the general network.

For the 8086 based ECs, two operating systems are considered now: the MTOS-UX from Industrial Programming Inc. and Intel RMX compatible DOS-86 which is available on the USSR software market. A final choice will be made by the end of this year. The LSI-11 based ECs will use an operating system similar to the DEC RSX-11.

User Interface

The user interface is based on graphical workstations and windowing techniques. Commercially available UIDS/UIMS (User Interface Development/ Management Systems) built on the basis of the X-Windows and OSF/Motif standards are considered now for the user interface creation and management (XFaceMaker, TAE+, Teleuse, VUIT). Such systems allow easy interface design and quick prototyping.

Much recent work in this field is concerned with specific extension of commercial products to a standard set of control tools and screen layouts which would allow to unify

human interaction procedures for a wide range of applications. Early specification, prototyping and demonstartion of such a tools would allow to start application software development and meet user's requirements.

DBMS and Real-Time Data Management

Off-line data preparation, including descriptions of machine and control system objects and relations, will be done using the Oracle DBMS and related tools.

One presently considers organizing all data inside of the control system in a specialized home-made real time DBMS. It should contain both static read-only data, derived from Oracle and dynamic data of the current machine state, some number of pre-defined UNK states for pulse-to-pulse modulation (PPM) and accelerator development, etc. This real time DBMS will support access to read/write data for fixed and off-line prepared data structures. It will serve a number of distributed data bases. Each DB is a standard file, containing data in the form of three-dimensional tables, some of which may be duplicated in the memory of specified computers..

Equipment Access

The diversity and multiplicity of process devices, requires some uniformisation, i.e. hiding the device specifics from the operational applications. For doing so, the device specifics shall be encapsulated in software envelopes having a standardized access protocol.

This concept leads naturally to the object-oriented approach to logical equipment representation. An equipment object can model a real, physical unit of the equipment or abstract entity, like a feedback loop or a data buffer.

The applications' vision of equipment is strictly limited to a relatively small number of logical device classes. Each logical device corresponds to a physical component or group of components of the accelerator equipment able to perform a complete task and considered as a single entity in context of the accelerator operation. A device object class can represent, for example, an ion pump with all its attribures (e.g., status, voltage, pressure, etc.) and services, which it provides for an application task (switch on, set voltage, read pressure, etc.). Each particular ion pump in the system will be an instance of the ion pump class.

A logical device is a complex object that encapsulates a collection of cooperating component objects. Each component object class represents a stable partial functionality inside of a logical device (input/output, local data buffering and processing, surveillance, timing, etc.). The set of the component object classes forms a toolkit for logical device construction. The set of component objects constituting a device, with their relationships and dependencies, formally represent logical device model.

A logical device undergoes the following main phases during its "life cycle" in the control system.

1. Device class creation: positioning of the class in device class hierarchy, definition of new logical device attrib-

utes and services using the component objects class library.

2. Implementation: integration of standard software modules, corresponding to the library classes, and, possibly, writing of a code reflecting a new device specific features. It should be noted that the same logical device can be implemented in several versions on different hardware platforms.

3. Instantiation: creation of a particular device object which is an instance of the device class. The object identifier, implementation version and equipment network address are specified at this phase.

4. Initialization: downloading of the device software into the specified platform, initial test and device setting in a predifined initial state.

Commercial Control Software Packages

Standardization trends result in the appearance of "generic" integrated commercial controls software packages which can be configured to realize a wide range of functionalities. Examples are V-System from VISTA, G2 from Gensym Corp., GENESIS from Iconics, etc.

For the purpose of thorough evaluation the V-System software will be used as a kernel of a relatively small prototype control system . The system will support commissioning of the U-70 to UNK-I beam transfer line, planned for the middle of 1992. The V-System will run in the VAX/VMS workstation connected via Ethernet to front-end computers (MS-DOS IBM/PC). Each of these front-ends will control a certain number of technological subsystems (including all types of power supplies, beam instrumentation, vacuum and utilities subsystems like electricity, ventilation etc.) by using equipment controllers connected via RS-232 interface. These PCs will be used as local consoles for equipment tests at the begining and thereafter they will work as a "data pumps" to supply data for the V-System data base.

4. CRYOGENICS ASPECTS

The cryogenics and related equipment is a substantial part of the UNK project and falls into three broad groupings.

1. The cryogenics complex that provides cooling of the superconducting magnets placed in the UNK ring tunnel.

2. The quench protection system.

3. The superconducting magnet main power supplies with their ramping and dc programs.

By their nature these systems have a close internal binding and require only a weak coupling with the main UNK control system. A fair degree of autonomy and stand alone capability is therefore foreseen, which is helpful in commissioning and later servicing.

According to the functionality and geographical distribution of the equipment, the cryogenics complex control is subdivided among four relatively independent subsystems.

1. The gaseous helium storage, compression and purification control. The corresponding equipment is located in three compressor station buildings placed around the UNK ring.

2. The helium liquefiers control. There are 6 helium liquefiers located in the central helium liquefier building.

3. The satellite refrigerators and magnet cooling control. There are 24 refrigerators located in 12 buildings regulary distributed around the UNK ring. Each refrigerator supports operation of one superconducting magnet string.

4. The Nitrogen Storage and distribution control.

Local access to the cryogenics equipment will be provided via local consoles in each cryogenics complex equipment building. The local consoles will be used in the equipment commissioning, autonomous tests and troubleshooting. The normal operation will be performed from the Cryogenics complex Control Room (CCR) located in the central helium liquefier building. The CCR will be equiped with the same sort of tools as the UNK MCR and provide 5-7 operator's workplaces.

Cryogenics complex controls will, like the main UNK control system but with some special flavours, use the Multibus-I based ECs with 16 bit processors connected by MIL-STD-1553 to the standard UNK FECs.

A specialized Multibus-I programmable controller module (SPC) has been developed and will be used at the lowest control level in the cryogenic system. The module consists of the general purpose 16-bit Multibus-I processor card and a functional card which is connected to the processor via a private bus. There are few types of the functional cards developed for interfacing to various kinds of transducers and actuators used in the cryogenics equipment. The SPC will autonomously realize closed loop control and relay logic algorithms according to the program stored in its PROM. A Multibus-I crate may house a number of SPCs working in parallel under supervision of the main EC processor. The total number of SPCs in the cryogenics complex control will be more then 2000.

In collaboration with CEA, Saclay, a first version of a FNAL-like quench protection system will be tested on an experimental sector of 8 protection units, each consisting of 12 dipoles and 2 quadrupoles. The front-end electronics and emergency heating power supplies are located in shielded cavities in the ring tunnel.

The main power supply controls will form a closed subsystem, loosely coupled to the main controls and interlocked with the quench protection system.

5. UTILITIES AND SERVICE NETWORK

The UNK complex includes a number of various utilities providing, as a whole, the safe and reliable environment for the main accelerator subsystems and personnel working in the accelerator area. These utilities are the general electricity and water distribution networks, tunnel ventilation and gas analysis system, radiation protection, fire safety and personnel access control to restricted access areas.

The utilities controls shall be supported by relatively slow, but very reliable and uninterruptable control system . A special highly reliable service network will interconnect a number of regional control nodes with central utilities control room. The service network will be linked to the main control network and access to the utilities can be provided also from the MCR consoles.

The market study, evaluation of commercially available components and prototyping have been started. A part of this work is design and implementation of utilities control system for the beam transfer line from U-70 to UNK-I (BTL). The system is based on the standard industrial programmable logic controllers (PLC). The PLCs perform low level control and monitoring functions and are linked via RS-232 lines to the IBM PC compatible console computer in the temporary beam treansfer line control room. The IBM PC collects data from the PLCs, performs data analysis and generates alarm messages and interlock signals in case of faults. It makes also periodic data logging and keeps a log of the system operation.

6. EXTERNAL BEAM LINES

The external beam lines will comprise a 12 km long neutrino channel, leading up to the neutrino experimental area, and three 6 km long hadron beamlines leading to their own experimental area each. The operational patterns of the beamline zones are, by their nature, different from the accelerators. The essential aspect is the more frequent and rapid changes, following the experiment's requirements.

The technology of the equipment used in these beam lines and experimental areas is similar to the one of the accelerators proper, including the use of superconductivity hence cryogenics. A strongly different aspect, however, form the target areas and splitting stations with their radiation problems and remote handling requirements. Some advanced devices such as polarimeters and crystal bending and focusing may be used.

One presently considers a controls architecture which is very close to the one described for the main UNK ring. There will be workstations with the modern windows and graphics oriented software, these will be interconnected with a TCP/IP based network, which in turn connects to the accelerator network, and at the frontend to VME based 32 bit microprocessors driving ECs over the MIL-STD-1553 fieldbus. In contrast with the accelerator system, the ECs of the external beam zones controls may still be CAMAC based.

Powerful number-crunching multiprocessor assemblies will be used for digital signal and image processing in the polarization hodoscopes, electron-gammas tagging system and TV- cameras based beam instrumentation.

Systems software will essentially be the same as for the accelerators. Applications will be strongly data driven and

model oriented and an expert system is being contemplated for operator support.

Although routine operation may be done from a central MCR, a strong local access and stand alone component remains essential.

7. U-70 CONTROLS UPGRADE

Contrary to the situation for the accelerator and beam zones controls, for which options were open, the U-70 injector group has a strong historical bias since both booster and U-70 proper are largely computer controlled. Moreover, the upgrade must be done virtually without interruption of the U-70 experimental programme. Finally, the urgency of this upgrade practically dictates using products now readily available in the USSR and using a stepwise implementation, converting small slices during the short planned shutdowns.

The FECs for the U-70 upgrade will be the Multibus I based SM1810.30 computer with an Intel-like 16 bit 8086/8087 processor board, using an RMX compatible real time kernel. They will have appropriate RAM capacities, a LAN interface and a parallel branch highway CAMAC driver. Each of the about 12 FECs, spread over 4 buildings, maximum 4 km apart, drives up to 3 CAMAC crates catering for I/O. Servers for files and data base, 4 to 6 in total, will be enhanced configurations of the FECs, featuring larger memory, hard disk and a more complete generation of the operating system. Interaction will be using PC-AT computers under DOS. A commercial LAN product is still being sought in the USSR.

8. PRESENT STATUS

A conceptual design study of the upper part of the control system has been made and is accepted. Prototype partial integrations of the main control system and front end assemblies, as well as a quench protection test facility, are being prepared and should be available early 1992. An applications development environment with servers and workstations is being prepared. The external beam zones controls are in the conceptual design phase. The U-70 controls upgrade has been largely defined and frozen and implementation has started.

The Multibus-I based ECs have been largely defined, prototypes of most modules exist, industrial contracts are being negotiated. All ECs for the BTL are assembled and tested and installation in the equipment buildings is in progress. The V-System applications for the temporary BTL controls are largely prepared and corresponding communications and PC front-end software is under development.

9. REFERENCES

1. M.Mikheev, N.Trofimov, E.Scherbakov, S.Zelepukin, B.Kuiper, "The UNK Control System Conceptual Design Study", IHEP and CERN internal report, June 1990.

2. V.Alferov, S.Goloborodko, S.Logatchev, A.Skugarevsky, "The Control System for the UNK Cryogenic Complex", IHEP internal report, November 1990.

3. V.Sakharov et al., "Preliminary Considerations on the Top Level Part of the Control System for UNK External Beam Channels", IHEP internal report, November 1990.

4. S.Balakin, V.Brook, V.Voevodin, V.Tishin, "The Control System for the Booster Synchrotron of IHEP", 19th School fpr Control Systems in Science Research, Novosibirsk, 1985, p.11.

5. V.Voevodin, "Preliminary Considerations of the Real-Time Data Management for the UNK Control System", UNK Control Group internal report, June 1991.

6. M.Mikheev, "Proposal for Use of the V-System Software in the UNK Beam Transfer Line Control System", UNK Control Group internal report, April 1991.

7. N.Trofimov, "On Object-Oriented Approach to the Equipment Access", UNK Control Group internal report, September 1991.

MOSCOW UNIVERSITY RACE-TRACK MICROTRON CONTROL SYSTEM: IDEAS AND DEVELOPMENT.

Chepurnov A.S., Gribov I.V., Morozov S.Yu., Shumakov A.V.,Zinoviev S.V.
Institute of Nuclear Physics, Moscow State University
119899, Moscow ,USSR.

Abstract.

Moscow University race-track microtron (RTM) control system is a star-shape network of LSI-11 compatible microcomputers. Each of them is connected with RTM systems via CAMAC; optical fiber coupling is also used. Control system software is designed on Pascal-1, supplemented with real time modules and Macro. A unified real time technique and reenterable data acquisition drivers allow to simplify development of control drivers and algorithms. Among the latter three main types are used: DDC methods, those, based on optimization technique and algorithms, applying models of microtron's systems. Man-machine interface is based on concept of the "world of accelerator". It supports means to design, within hardware possibilities, various computer images of the RTM.

INTRODUCTION.

Moscow University race-track microtron - when it's construction will be finished - is to produce 175 MeV 100% duty factor electron beam with low transverse emittance (0.05 mm*mrad) and up to 0.01% energy monochromaticity [1,2]. To support means for easy programming of microtron's behavior, when being adjusted, and to meet requirements of experimental work, computer-based control system is to be developed. It's configuration is shown on fig. 1

HARDWARE.

Each micro computer of the control system is a 1-PCB LSI-11 compatible machine (EIS,FIS CPU; 1 mips; 56 Kb RAM). In control station it's connected with CAMAC via JCC-11 compatible crate-controller. Among CAMAC modules the following types are used: output and input registers (standard and specialized), FET multiplexers, 13,14,16 bit ADCs, step motor drivers. To prevent inadmissible interference, control system is isolated electrically from accelerator's equipment. For this purpose optically coupled measurement devices are used. Their terminal modules can be of three types: 19 bit TTL transmitter or receiver, 16 multiplexed a dozen bit ADCs or eight 12-bit DACs. Control stations (three of them in operation now) are connected via RS-232C interface in a star-shape network, formed by concentrator station. The latter is also tied with host-machine, used for software development and system loading. Man-machine interface and data-bases station is linked up with network like a control station. Concentrator machine and control stations have no any extra memory storage except CPU RAM. Man-machine and data bases station includes two microcomputers. One of them supports graphics, another handles data bases and communication protocols. This station, as well as host machine, is supplied with disc memory - including electronic one. Alphanumeric displays

(VT-100) and some other peripherals are also attached. Among them - RS-232C interface, allowing to link up control system with external computer or network. Man-machine interface will also be supplied with four infinite-turning knobs to facilitate manipulation with one, two or three dimensional objects (control parameters value, terminal cursors etc).
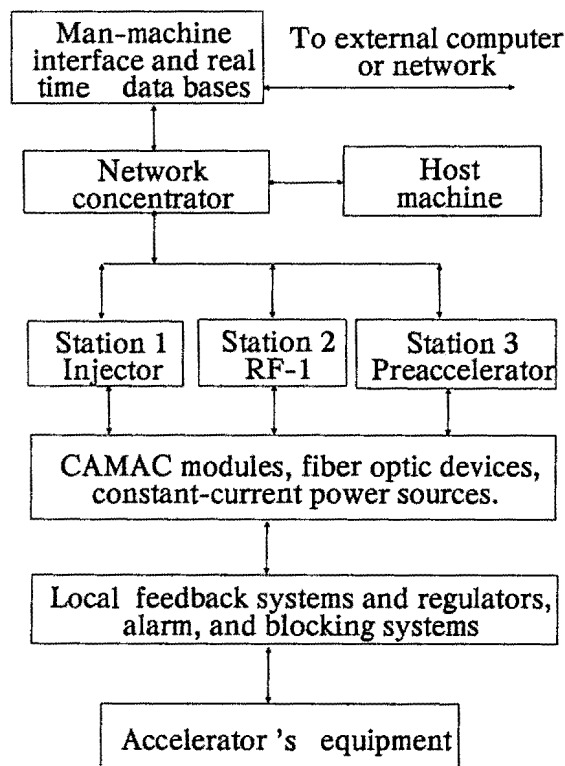

Fig. 1. Block diagram of Moscow University RTM control system.

SOFTWARE.

Compilation tools.

Basic compilation tools are shown on fig. 2. Source Pascal code with Macro insertions is being compiled with Pascal-1. Then a specially designed improver heightens an effectiveness of resulting Macro code. Afterwards, on Macro stage, it's being combined with CAMAC support

modules. The resulting object code is being linked with Pascal library and the one, containing control system support modules.

Control station software structure.

The structure is depicted on fig. 3. Feedback control loops include control drivers, algorithms and reenterable data acquisition drivers. Reenterability allows to obtain measurement data by any program module, initiated with interrupt, while the same device is used by another program unit. Control drivers and algorithms are supervised with monitor. Real time processes deal with interrupts, initiated from different sources (network, CAMAC, timer). Network support system handles net protocol; access to data transfer buffer is also reenterable. Low level real time techniques include P-V operations to protect critical resources, repetition of critical section with non-savable resources and counting flags to prevent CPU overload.
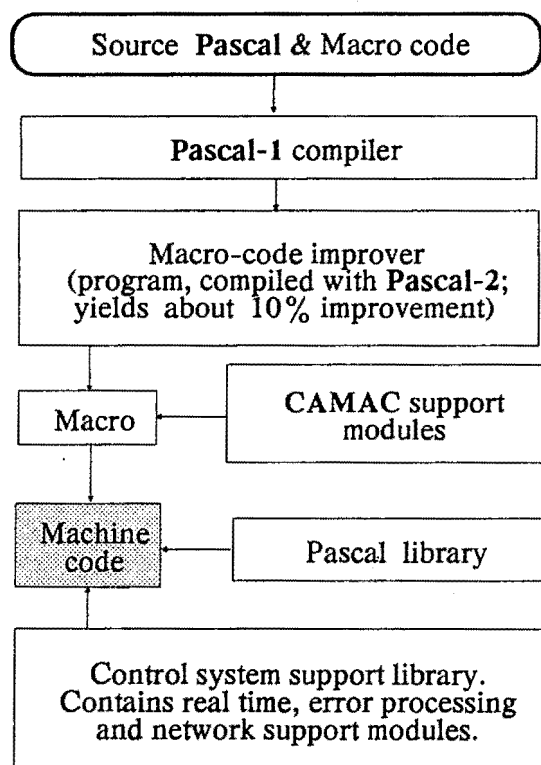

Fig. 2. Basic compilation tools.

Network data communication.

Data transparent network protocol is supported by exchange of byte-serial frames. They include command code, destination and source codes, data counter, unit of data and checksum. Command code indicates function to
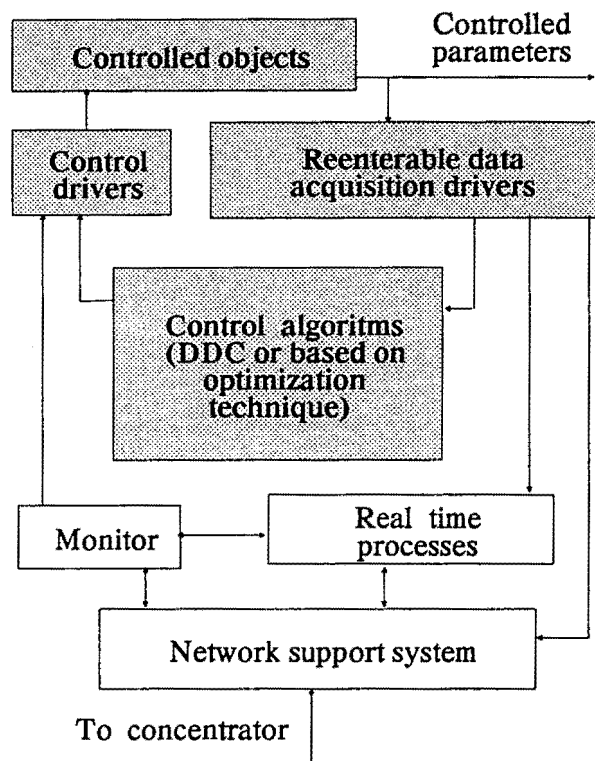
Fig. 3. Control station software structure.

be accomplished with received parameters. These functions include: status of parameter inquiry, control, parameter value acquisition, free coefficients setting (calibration) and implementation of predefined operations. Special frames, consisted of only command code, are used to support network protocol and characterize general subsystem status.

Algorithms.

Algorithms, realized in RTM control system, can be divided into three groups: dynamic methods, functioning under strict time conditions, and two groups of quasistatic algorithms, where timing is not essential. The latter are based either on optimization technique or apply physical models of microtron's systems. Control system itself executes only dynamic and optimization methods. Dynamic ones include DDC methods both for logical control (switches) and analog algorithm simulation. The latter one (PI method) is used for temperature stabilization of accelerating sections [3]. Optimization algorithms are realized in one and two dimensional modes. One dimentional algorithm is based on "regula falsi" method and is used for fine tuning of various control parameters. in particular, to stabilize reference frequency generator. Two dimensional algorithm is used to steer electron beam via collimators by minimizing it's leakage current. The method deals with approximately symmetric goal function with flat bottom and uses non-derivative direct searching algorithm. At first stage steering algorithm scans with beam, using spiral trajectory, to find out goal function

141

position. Model algorithms are not immediately supported by control system. They can be carried out with the help of any external computer, interfaced with control system. For this purpose man-machine interface computer also handles remote terminal protocol, allowing to inquire and set new values of any controlled parameter or to activate predefined functions.

Software development technique.

A considerable number of controlled parameters (about 400 now) requires definite technique to develop software. The development cycle is shown on fig. 4. It allows to avoid discrepancies between expression of requirements and actual control station program. During the cycle parameters data base is being corrected and supplemented. This base is used later to develop man-machine and "external world" program interfaces.
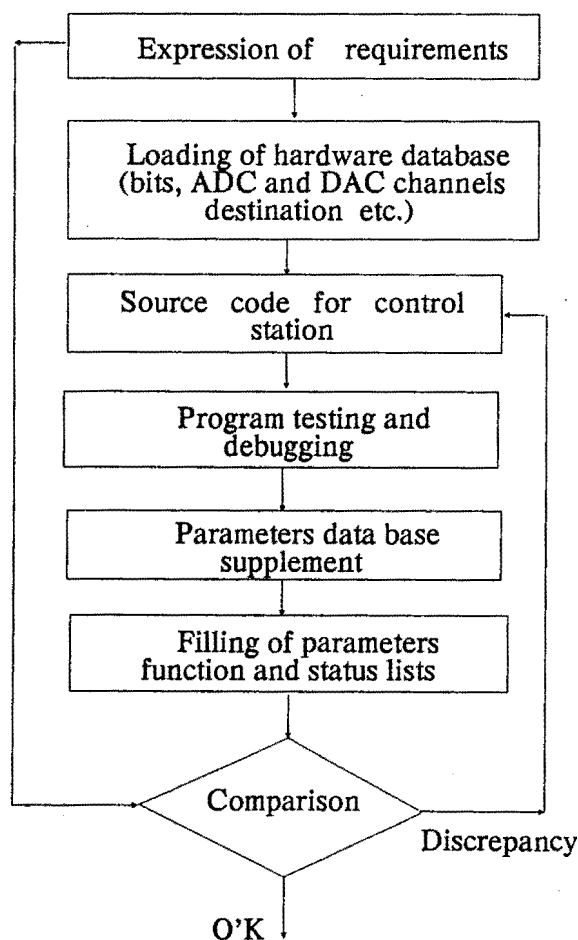

Fig. 4. Software of control station development cycle.

Man-machine interface and external world communication.

Man-machine interface software is based on approach, which gives means for operator to design - within hardware possibilities - his own computer images of the accelerator. It is supported by several types of windows and a list of their names (menu), allowing to activate any window. Special program supports creation of menu and windows. Graphical window can be chosen among several predefined types, which include those, representing several two-dimensional curves and the windows, which support an amplitude analyzers mode. Operator is not able to change geometrical shape of a window, but have a possibility to set colors, inscriptions and some other attributes. Any graphical window can then be loaded in a graphics support computer. Alphanumeric windows contain a set of parameters, extracted from parameters data base. Their values are represented on a terminal in any chosen position. Operator can previously fill the window with an arbitrary text. One type of alphanumeric windows is used to set constant parameters of accelerator (calibration coefficients etc). Their values are defined during a window's editing and transferred to subsystems when it's activated. Another type of windows - a working ones - support actual interaction with control system. They inquire and represent current values of parameters on a display. Operator is able to set a new value of any window parameter or initiate predefined control procedures. To adjust microtron's systems, it's possible to scan with any parameter (time among them), while others are represented graphically or (and) listed in a data-storage file. Working windows automatically support local data base files. Parameter's values are stored there before exit and extracted from the file when window is activated. There are also several windows, which are not programmable and are used to support system functions (time setting, password etc). Statuses of parameters, obtained from control stations, are stored in a special data base and can be displayed by operator's request. As mentioned above, man-machine computer also supports "external world" interface with a required data communication protocol.

References.

[1]. A.S. Alimov, I.G. Artiukh, V.I. Shvedunov et al. Preprint NIIYAF MGU, 88-012/33, Moscow, 1988. (in Russian).

[2]. A.S. Alimov, V.G. Gevorkyan, I.V. Gribov et al. "Beam emittance forming line of the CW Race-Track microtron of the institute of Nuclear Physics of Moscow State University (INP MSU)". Nuclear Instruments and Methods in physics research A278 (1989)p. 379-388.

[3]. A.S. Chepurnov, I.V. Gribov, S.Yu. Morozov, A.V. Shumakov and S.V. Zinoviev "Feedback Systems for Local Control of Race-Track Microtron RF Accelerating Sections". These proceedings.

Present Status of Control System at the SRRC

G. J. Jan*, J. Chen, C. J. Chen, C. S. Wang
Synchrotron Radiation Research Center, Hsinchu 30077, Taiwan
* also Department of Electrical Engineering, National Taiwan University, Taipei 10764, Taiwan

ABSTRACT

The modern control technique was used to design and set up a control system for the synchrotron radiation facilities at the synchrotron radiation research center (SRRC). This control system will be finally to operate the dedicated machine to provide the 1.3 GeV synchrotron radiation light. The control system will control and monitor the components of storage ring, beam transport and injector system. The concept of the philosophy is to design a unique, simple structure and object-oriented graphic display control system. The SRRC control system has the major features such as two level architecture, high speed local area network with high level protocol, high speed microprocessor based VME crate, object-oriented high performance control console and graphic display. The computer hardware system was set up and tested. The software in top level computers which include database server, network server, upload program, data access program, alarm checking and display, as well as graphics user interface (GUI) program were developed and tested. The operational system and device driver on the field level controller were implemented. The overall performance of the SRRC control system were tested and evaluation. The preliminary results showed that SRRC control system is simple, flexible, expandable and upgradable open system to control and monitor devices on the small scale synchrotron radiation facility.

I. INTRODUCTION

The 1.3 GeV synchrotron radiation facility is going to construct at SRRC to provide a low emittance and high brilliance light source. The dedicated synchrotron light source will include three subsystems that are a turn key 1.3 GeV electron full energy injector, beam transport line and storage ring. The 1.3 GeV full energy electron injector was constructed to Scanditronix AB at Sweden and installed at SRRC site. The injector is going to commission within 2 or 3 months. The beam transport line has been designed and its major component was also fabricated partially. The storage ring with triple bend achromat lattice [1] has been designed and the most components were constructed and passed its qualification of the specification.

The control system at SRRC provides a unique control and monitoring three subsystems which include the injector, the beam transport line and the storage ring facilities. The 1.3 GeV energy electron injector is composed of a 50 MeV linear accelerator and 1.3 Gev booster synchrotron accelerator. The control system of booster synchrotron can be run in a turn-key system and/or to be integrated into SRRC control desk to form an unique control system. The design concept is to standardize the same computer architecture, digital communication network with some protocols and some

database structure as well as some console computer. The control system of the electron injector can play as a stand-alone subsystem to test and commissioning machine as well as machine study for booster synchrotron.

The control system of SRRC is cost-effectively designed by using recent developed computer technology [2,3] and modern control technique [4-9]. Two level hierarchical computers, process and console computer as a top level and multiple intelligent local controllers (ILC) as a low level, was configured. One process computer and several console computers at top level provide the database management and maintenance, devices control and monitoring, data logging and archiving, machine modeling, object-oriented graphical display. The lower level computers are multiple VME crate based system which handle device related data acquisition and control as well as local interlocking functions. It simplifies the architecture of control system at SRRC. The upper and lower level computers are connected by ethernet using high level protocol.

The process computer will handle the static and dynamic data base. It also offers to carry out the calculation of the electron orbit and simulation of the machine physics parameter. It is a high speed computing, multi-task and muti-user virtual memory system (VMS), and high through input/output (I/O) capability. The console computers will play a similar job as a process computer except maintenance of the database. The console computer plays an important role to operate and monitor the synchrotron radiation facility using man-machine interface that is developed based on the concept of the object-oriented graphic display. This is the most recent development on the third generation of the synchrotron radiation facility [7-9].

The ILC is a field level (or called device level) controller which performs the local data collection, local interlocking and closed loop control for the components and/or the equipment of the various system. It is also very important for the real time feedback control system.

II. COMPUTER HARDWARE SYSTEM

Two level hierarchical computer configuration has been designed and installed partially at this moment. The hardware configuration of the control system at SRRC is shown in figure 1. The top level computers are composed by one process computer and several console computers. The VAX 6000 model 610 supermini computer is chosen as a process computer. The VAXstation 3100 model 76 is selected as a console computer. The big semiconductor and mass storage capacity as well as high speed I/O peripheral devices are also considered and configured.
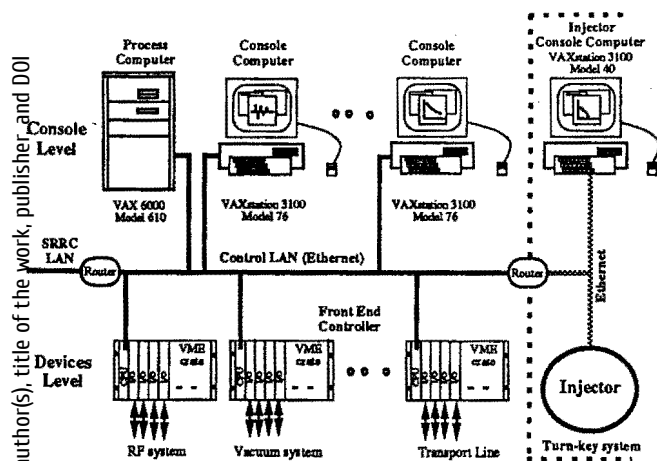
S03SRD10

143

Figure 1. Hardware configuration of the control system at SRRC

The process computer creates and maintains the database and manage them at current development. The central database has been used at SRRC control system due to the on-time schedule to commission the machine. It will be modified as a distributed data base in near future. The update data rate from ILCs is set 10 Hz due to broadcasting mode or under request mode. These data can be received by any top level computers. The top computers can also set the parameters of the device and read the signal back and display it graphically. The process computer is running in VMS operating system but the console computer can run in VMS or Ultrix operating system. The ethernet using TCP/IP protocol is used to link all top computers and multi-ILCs.

The console computers are composed of VAXstation 3100 model 76, model 40 and DECstation 5000 model 200 that is run Ultrix operating system. The workstation will provide the console control and monitor the machine parameter and/or device signal through user graphic interface software. It provides the real time data trend or graphic display friendly.

The ILC is an VME crate system which includes Motorola MVME-147 central processing unit (CPU) board and variety of I/O cards. The CPU board consists of 68030 microprocessor , 68882 floating point processor, 4 Mbyte on-board memory and ethernet interface. The field devices are connected into ILCs through parallel analog or digital input/output or IEEE-488 bus standard or serial communication interface. The major tasks of the ILCs will handle the data setting, data acquisition interlocking, and close-loop control and monitoring function of the equipments or devices. Twelve set ILCs will handle the magnet power supply, RF system, vacuum system, beam diagnostic, and general purpose measurement instruments, ... etc. The user from the workstation can set and read back the parameters of the device easily.

III. COMPUTER SOFTWARE SYSTEM

Based on the VMS operational system and utility package, the software of the control system at SRRC has been designed and basic program of this software was coded and tested. The block diagram of software structure is
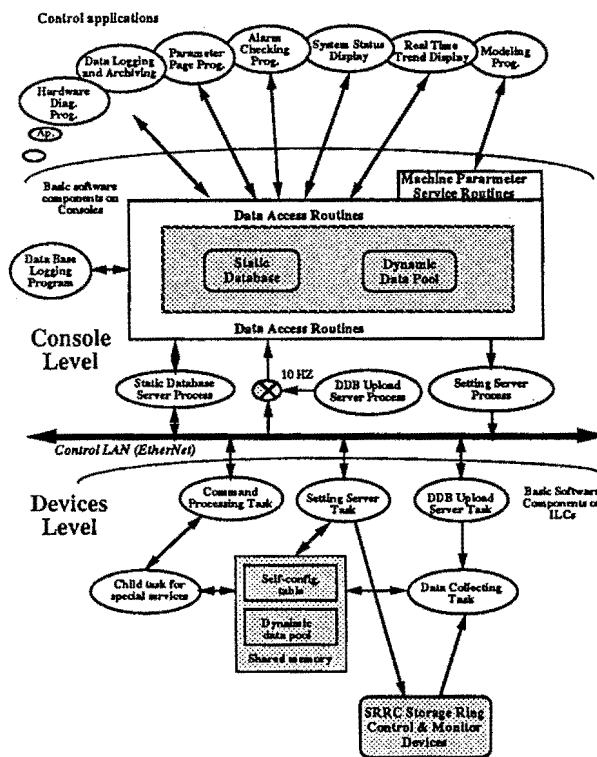
Figure 2. Block diagram of software system

shown in figure 2. The software structure is divided into several logical layers that is device access, network access, database management, graphics user interface and applications. The goal to modularize the software into layers is to reduce the developing time.

The device access process are run on the ILCs. The pSOS+ real time kernel provides the ILC with support for task scheduling, memory allocation, even handling and message queuing. The pNA+ network support package provides socket network interface. The control tasks and various I/O tasks are developed and tested on the ILC. The device dependent software drivers are implemented partially. The magnet current regulator, RF low-level electronic controller and vacuum gauge controller were implemented and tested successfully. The speed of the dynamic data uploading to the top level computers is about 10 Hz. Downloading the database from the process computer into the ILCs to form a local distributed is initialized and underway.

The network access software is in charge of the information exchange between the top and the low-level computers. The protocol of the IEEE-802.3 is used to communicate with the turn-key injector system. The high level protocol TCP/IP is using at this moment. The reason to use IEEE-802.3 and TCP/IP protocols due to the beginning of the vendor contract to made an agreement for IEEE-802.3. That time the TCP/IP is not popular enough for VAX computer and 68000 series computer vendors. However, it would be changed into TCP/IP after the commission of the electron injector to form a unique local area network.

3rd Int. Conf. Accel. Large Exp. Phys. Control Syst.   ICALEPCS1991, Tsukuba, Japan (partial, pages 194 of 611)   JACoW Publishing

ISBN: 978-3-95450-254-7   ISSN: 2226-0358   doi:10.18429/JACoW-ICALEPCS1991-S03SRD10

The database access is developed on the console level computers. The console level computers are VAX/VMS system (or can be an Ultrix system for the console computer). The software is coded in C language. The function of the process computer and operational console computer (or called workstation) is slightly different. The process computer keeps the system-wide static database and maintains it. The upload sequence is also requested by the process computer. At system start-up, each workstation requests and receives a copy of the static database from process computer. Each console computer then has all the database information necessary to process dynamic database frames received from the ILCs.

The workstation are mainly to provide user to operate the machine. The upload sequence is requested by the process computer, the ILCs multi-cast the dynamic data sequentially. All of the console level computer received dynamic data and updated into database at the same time.

The central database on the console level computer is used as buffer between the low level tasks at the ILCs and the console level applications. The application programs of the user can access the equipment parameters directly from their own database rather than from the ILCs. The application programs are device transparent. The development programs on the top level computers can be run concurrently.

The basic control and monitoring program has been developed at current stage. These programs are urgently need for machine commissioning. Those programs include the data logging, archiving, alarm checking and display routine etc., were coded and tested successfully. The machine model calculation can be run on process computer and/or individual workstation. The graphics user interface software was developed based on X Windows and Motif in VAXstation 3100 model 76. The block diagram of the relationship to develop the graphics user interface is shown in figure 3. The graphic edit program is to edit the display pattern of the machine components and build up the linkage relationship between the component and the static database. Those pattern file is stored with ASCII format in the hard disk. The control program is to read these ASCII file and make a connection between the component pattern and dynamic database. This program is also to execute the task of the data reading, setting and display from the DDB server. All subprogram is written in modularize software package. This human interface software will provide the operation of the synchrotron radiation facility more friendly.

IV. SUMMARY

The control system for the dedicated synchrotron radiation light at SRRC has been designed and the computer hardware and software was implemented partially since last year. Two level hierarchical computer control system has been configurated and the high speed local area network that use ethernet and high level protocol such as TCP/IP have been implemented and linked. The data upload rate is to maintain about 10 Hz without increase of traffic load at network. The over-all performance of the process computer and multiple ILCs as well as digital communication network has been
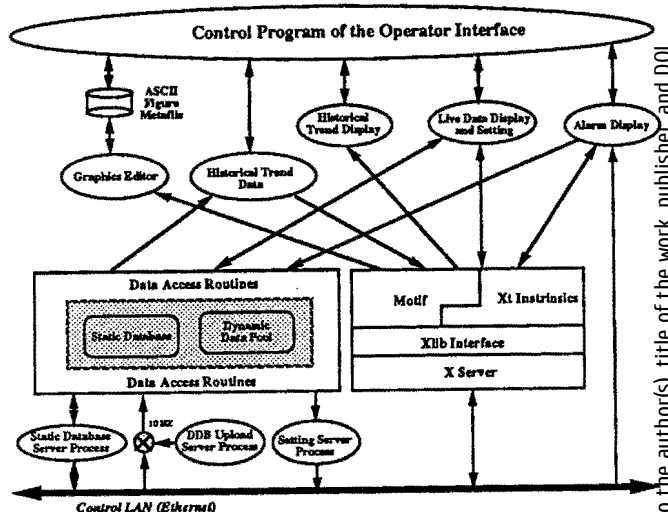
Figure 3. Block diagram of the operator interface

tested and evaluated. The results show the 10 Hz update from the ILCs is not big issue to control and monitor the devices for synchrotron radiation facility at SRRC.

The object-oriented development tool of the graphic display and data trend display were developed and tested. The user can set and read back the signal from the field level devices within 10 Hz. The flexible change of the time interval for the data display at specified signal is allowable to change from 1 sec up to 60 sec. The GUI software can be developed from the developed basic edit program to code and modify it. This software will provide the user to operate the machine were easily and friendly. Finally, the two level computer architecture, multiple ILCs that are linked by ethernet local area network using TCP/IP, object-oriented graphic display software will be a nice control system for synchrotron radiation facility.

V. ACKNOWLEDGEMENTS

VI. REFERENCES

[1] "SRRC design Handbook", Synchrotron Radiation Research Center, April 1991.
[2] M. Castellano, S. De Simone, G. Di Pirro, S. Guiducci, P. Patteri, M. Serio L. Trasatti, "Diagnostic and control system for the LISA project", INFN, Laboratori Nazionali Di Frascati, European Particle Accelerator Conference, Rome, 1196 (1988).

145

[3] A. J. Kozubal, D. M. Kerstiens, J. O. Hill and L. R. Dalesio, "Run-time environment and application tools for the ground test accelerator control system", Nucl. Instrum. and Meth. A293, 288 (1990).

[4] S. Magyary, M. Chin, C. Cork, M. Fahmie, H. Lancaster, P. Molinari, A. Ritchie, A. Robb, C. Timossi and J. Young, "The advanced light source control system", Nucl. Instrum. and Meth. A293, 36 (1990).

[5] B. Robinson, and P. Clout, "Modularization and standardization of accelerator control systems", Nucl. Instrum. and Meth. A293, 316 (1990).

[6] J. P. Stott and D. E. Eisert, "The new ALADDIN control system", Nucl. Instrum. and Meth. A293, 107 (1990).

[7] W. D. Klotz, C. Hervé, "The conceptual design of the ESRF control system", European Particle Accelerator Conference, Rome, 1196 (1988).

[8] Marco P. Mignacco, "The ELETTRA control system", Nucl. Instrum. and Meth. A293, 44 (1990).

[9] F. Potepan, "The ELETTRA man machine interface", Internal Report ST/M-91/2, Sincrotrone Trieste, Italy, March 1991.

3rd Int. Conf. Accel. Large Exp. Phys. Control Syst.   ICALEPCS1991, Tsukuba, Japan (partial, pages 194 of 611)   JACoW Publishing

ISBN: 978-3-95450-254-7   ISSN: 2226-0358   doi:10.18429/JACoW-ICALEPCS1991-S03SRD11

Status Report on Control System Development for PLS*

S.C. Won, S.S. Chang, J. Huang, J. W. Lee, J. Lee, J. H. Kim

Pohang Accelerator Laboratory,
POSTECH, P.O.Box 125,
Pohang 790-600, Korea

Abstract

Emphasizing reliability and flexibility, hierarchical architecture with distributed computers have been designed into the Pohang Light Source (PLS) computer control system. The PLS control system has four layers of computer systems connected via multiple data communication networks. This paper presents an overview of the PLS control system.

Introduction

The accelerator control system provides means for accessing all machine components so that the whole system could be monitored and controlled remotely. These tasks include setting magnet currents, collecting status data from the vacuum subsystem, taking orbit data with beam position monitors, feedback control of electron beam orbit, regulating the safety interlock monitors, and so forth. To design a control system which can perform these functions satisfactorily, certain basic design requirements must be fulfilled. Among these are reliability, capability, expansibility, cost control, and ease of operation.

Considering above requirements, the PLS accelerator topology, available resources, special accelerator hardware requirements and personal preference, we propose a hierarchical system architecture. To implementation of the control system, highly commercial approach should be made because of the tight construction schedule. Using well proven technology will promote reliability, and reduce the development effort. A development environment is set up to develop the prototype of beam close orbit correction system which is to damp up to 15Hz movements. The Beam close orbit correction system will use DSP(Digital Signal Processor) board for the fast computation. All BPM electronic modules and corrector power supply interface I/O modules for one acromat will be put into one VXI crate.

Hardware Hierarchy

To monitor and control the thousands of signals for PLS, it is desirable and cost-effective to establish a distributed control system based upon microprocessors. The PLS control system has a hierarchical structure as shown in Fig.1. The hierarchy consists of four layers, each of which has a different role. The four layers
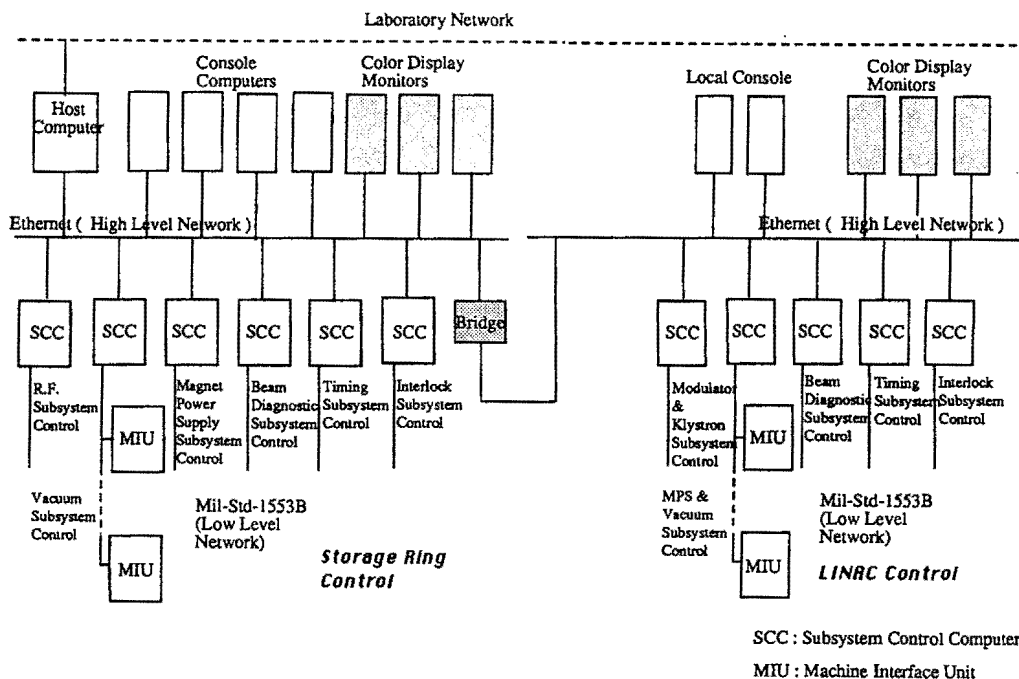

Figure 1. PLS Control System Architecture

are the computing service layer, the human interface layer, the subsystem control layer, and the machine interface layer. The computing service layer is a time-shared host computer in the computer room, and it is linked to various operator consoles which form the human interface layer. These workstations are located in the main control room. The consoles are connected to the master crates which form the subsystem control layer via an Ethernet. These master crates are located in local control stations distributed throughout the machine. The master crates are connected to slave crates, which form the machine interface layer, via a MIL-STD-1553B network. Each slave crate can use a variety of protocols to interface with machine components. Among these are IEEE-488, RS-232, RS-422. New protocols can be added with plug-in modules.

Computing service layer - Host Computer

A host computer with a high computational speed, large memory, and multiple job capability is needed for code and data management, mathematical modeling and simulation, off-line analysis, and for general purpose computing. For a low emittance light source, there are many critical parameters and it is more important to provide good computer modeling or simulation of the beam optics. For this purpose, the computation speed should be fairly high, and the machine should have at least a 32 bit processor with substantial physical and virtual memory space.

The DEC/VAX SYSTEM 6000^{TM} may be a good choice for this as the VMS^{TM} operating system provides an excellent software foundation upon which to run software donated by cooperative accelerator laboratories. However, if we can port the required software to the UNIX^{TM} environment with little effort, a high performance RISC computer might be an alternative.

The host computer will be installed at the end of 1993 after the storage ring building is completed.

Human interface layer - Console Computers

Mid-range engineering workstations will be used as operator consoles to put in the operational parameters and show the operating status. Input parameters will go through some arithmetical processes and be converted into control commands for each affected piece of equipment.

Operating status should be shown on color display monitors in the form of simulation diagrams, various kinds of charts, and other graphical expressions. For this role, an engineering workstation is the best fit due to its high computing power and relatively low cost, high resolution color graphics capability, and high performance multiple window display system.

Currently, several SPARCstations^{TM} of Sun Micro Computer Corp. are being used for console software development since they are well equipped with competitive capability, open system policy, and software availability.

Even though workstations provide excellent graphics capability, typical implementations of easy-to-use man-machine-interfaces have required sophisticated programming effort. However, we expect the use of a GUIDE (Graphical User Interface Development Environment) will be a great help in saving development effort.

DataView^{TM} of V.I. Corp. is a potential solution. DataView^{TM} consists of two components: DV-draw^{TM}, an interactive drawing editor for creating sophisticated screens without programming, and DV-tools^{TM}, a library of subroutines that connects graphics created with DV-draw^{TM} to an application program. Thus it saves development effort, allows rapid prototyping, and future

software interoperability derived from its support for industry-standard platforms. In addition, we will be able to easily integrate with other software environments such as relational databases and expert systems, and effectively develop integrated software by separating the GUI from application code.

A total of six workstations will be used as operator consoles for the storage ring and the linac, and a few additional color display monitors are planned for continuous display of information such as beam current, vacuum status, and magnet power supply current.

Subsystem control layer - Subsystem Control Computers

The Subsystem Control Computers(SCC) are microprocessor assemblies based on VMEbus and Motorola's 680x0 microprocessor. The reason for adopting VMEbus is its reliability and popularity. Each SCC consists of one Single Board Computer(SBC), an Ethernet interface module, and a MIL-STD-1553B network interface module, all of which are put together in a VMEbus crate. Motorola's MVME147^{TM} with 32-bit 68030 CPU and 4 Mbyte memory is used for the SBC of the SCC.

The SCC does many control and monitor functions, such as reading and setting parameter values of machine components, feedback-control, alarm handling, and raw data processing for each subsystem, i.e., vacuum, magnet power supply, R.F., beam diagnostics, timing, and interlock. Each SCC is interconnected to the multiple Machine Interface Units(MIU) through MIL-STD-1553B network. The SCCs are also linked to the high level computers through Ethernet.

Machine interface layer - Machine Interface Units

MIUs are also microprocessor assemblies based on the VMEbus and Motorola 680x0 microprocessor family. Each MIU has one SBC, one MIL- STD-1553B network interface module, and a number of analog and digital input/output modules, all of which are put together in one or more VMEbus crates. An SBC equipped with Motorola's 68000 16-bit microprocessor and 1 Mbyte memory is used for the MIU. In order to handle various types of analog and digital input/output signals, a wide range of standard analog and digital input/output modules as well as home-made or non-standard interface modules are used. MIUs are distributed in local field stations, which are located around the machine, to reduce the length of the signal cables between the MIUs and the machine components and also to reduce electromagnetic noise problems with the cables. There are 12 local field stations around Storage Ring, 3 around Linac and 2 around BTL. Multiple VMEbus crates in the MIU are interconnected with VMEbus-to-VMEbus repeater module.

A development environment is being set up for the development of SCC and MIU. This environment is composed of development hosts and target VME assemblies, which are connected through Ethernet(TCP/IP). Motorola's SYS1147^{TM} VME station, which uses MVME147^{TM} as its single board computer, is used for the resident development host, and Sun Micro Computer Corp.'s SPARCstation/IPC^{TM} is used for the cross development host. Microware's real-time operating system, professional OS-9^{TM}, is ported to the resident host and industrial OS-9^{TM} is ported to every target VME system. Initially, this development environment will be used mainly for the development of the beam close orbit correction system and the modulator/klystron control system.

Data Communications Network

The components of the PLS control system are linked via two levels of data communication networks; a low level and a high level network. The low level network is used for data acquisition and forwarding of control commands. The high level network delivers operational setpoint values to the lower level computers and information acquired by the lower level computers to the console computers.

The low level network must be tolerant of electro-magnetic noise because MIUs should be installed close to various noise-generating equipment and cables. Due to this requirement, MIL-STD-1553B specification is used. MIL-STD-1553B is a multi-drop network specification which is operated in a master/slave mode on which one Bus Controller(BC) may communicate with up to thirty Remote Terminals(RT). The SCC acts as the BC and the MIUs as RTs.

For the high level network, Ethernet is chosen because of its popularity which allows cheap and easy implementation for both hardware and software. TCP/IP will be adopted as the higher layer protocol for the same reason. Ethernet is a CSMA/CD network with a maximum transfer rate of 10 Mbps. However, the length of transmitted packet and transmission speed must be carefully selected to guarantee the appropriate transfer delay time and throughput.

Software

System software

A real-time operating system should be used for the SCCs and MIUs because some machine control jobs such as closed orbit correction requires real-time performance. OS-9^{TM} from Microware Systems is selected because of its advanced kernel functions and variety of software development tools. OS-9^{TM} not only provides a real-time kernel and its associated system modules, but all the file managers and device drivers necessary to support integrated i/o processing. The user interface includes an easy-to-use UNIX-like shell, hierarchical directory/file structure and over 70 utility programs to allow simple user access to the operating system's management functions.

The database on the SCC contains all subsystem parameters. It also has a component table which acts as a name sever providing the translation between the different symbolic names allowable at the high level and signal names in the MIUs, as well as linking these names with addresses of the appropriate MIU. The database on the MIU contains component data such as hardware addresses, set values, limits, conversion and calibration factors, and so forth.

Machine control software

Operating status of each subsystem such as vacuum, R.F., and magnet power supply subsystem should be displayed or archived to files, which might be later processed to analyze the operating history.

Desired setpoint values such as magnet power supply current and cavity gap voltage should be set. Setpoint values and other operational parameters might be put into the control system manually by the operator or automatically by the control software.

Intolerable differences between setpoint values and corresponding readback values should be continuously monitored and re-ported to the operator. These values are archived in files at the same time. The differences might be compensated by a slow feedback control program as long as there is no serious failure in correction. Continuous failure in correction would result in an alarm situation, which would be reported to the operator immediately. Any fault in apparatus should be reported to the operator, who can cope with it appropriately.

A total of 110 man-month is estimated to be necessary to develop the machine control software including the man-machine-interface, for both storage ring and linac .

Database

A comprehensive database defines all machine parameters and device signals. The database is generated in one of the console computers. The generated database consists of two parts: the static and the dynamic part.

The static database includes static machine parameters and device information such as names, locations, and various coefficients which might be used to convert scientific units into actual signal values, and vice versa.

Generated static database should be shared among the console computers by transferring the static part and maintaining consistency between the original and copies. Appropriate portions of the static database should be downloaded to the SCCs to be used to control each subsystem properly. The SCC might download parts of its static database to MIUs under its control. MIUs use this static database to convert scientific setpoint values into actual control signal values or actual monitoring signal values into scientific readback values.

The dynamic database consists of setpoint values such as magnet power supply currents and cavity gap voltages, and readback values such as ion gauge currents, magnet power supply currents and cavity gap voltages.

Dynamic database on a console computer has just a structure at the very beginning. It might be filled with valid data, when a control process which might need appropriate data were spawned. The valid data should be transferred from the appropriate SCC to the requesting console computer on "Supply-On-Demand" basis. Setpointing might also cause updating part of dynamic database with setpoint values.

For the PLS database structure, we took the idea of SPEAR's database system, since it is fairly portable and helped us to save development effort[4]. However, the update policy of dynamic data is quite different between the two database systems. PLS has distributed databases on the SCCs, the data of which can be supplied to upper layer computers on "Supply-On-Demand" basis. On the contrary, SPEAR has centralized database which is continuously updated by the hardware. We expect our scheme to maximize the network throughput by keeping unnecessary data from being transferred via network. Resulting gain in network capacity could be used for faster data acquisition of any particular signal group.

Beam diagnostic software

Knowledge of accurate machine parameters is very important for the efficient operation and study of the machine. We intend to automate all the beam diagnostic processes with various beam diagnostic programs such as beam orbit measurement, real time orbit correction, tune measurement, beam lifetime, beam emittance, and lattice function measurement, etc. Some diagnostic software will be run at the low level control computers for the

real time beam diagnostics and feedback.

Modeling and simulation software

Traditionally commissioning of particle beamlines is a very time-consuming and laborious task. Even in day-to-day operation after start-up, various types of machine and beam errors have to be corrected. To reduce the time and effort for these tasks, fast and easy-to-use computer programs are needed.

Conclusion

An important requirement in designing PLS control system is flexibility. Natural growth of the system will require expansion of the control system or more complex control of the accelerator, and it is often necessary to implement new technology on the existing control system. We have designed a control system which makes substantial use of industry standard components, thus maintaining a very good level of flexibility and expansibility. we expect (and hope!) that industrial development of these standard component will procede faster than the ever increasing requirements on the PLS control system.

References

[1] S.C.Won, Jae W. Lee and Jinwon Lee, "Computer control system", PLS Conceptual Design Report, pp. 2-101, POSTECH, Korea, Jan. 1990

[2] S.C.Won, Jae W. Lee and Juno Kim, "Computer control system for PLS", Proc. of the 4th Asia Pacific Physics conference, Seoul, Korea, Aug. 1990, pp. 1114-1117

[3] S.C.Won, Jinwon Lee and Ji Hwa Kim, "DACS(Data Acquisition and Control System) for PLS", Proc. of the 4th Asia Pacific Physics conference, Seoul, Korea, Aug. 1990, pp. 1139-1142

[4] S. Howry, T. Gromme, A. King, M. Sullenberger, "A portable database driven control system for SPEAR", SLAC-PUB-3618, SLAC, Apr. 1985

Design of SPring-8 Control System

T. Wada, T. Kumahara, H. Yonehara, H. Yoshikawa, T. Masuda, and Wang Zhen
JAERI-RIKEN Spring-8 Project Team
Honkomagome 2-28-8, Bunkyo-ku, Tokyo, 113 Japan

Abstract

The control system of SPring-8 facility is designed. A distributed computer system is adopted with a three-hierarchy levels. All the computers are linked by computer networks. The network of upper level is a high-speed multi-media LAN such as FDDI which links sub-system control computers, and middle are Ethernet or MAP networks which link front end processors (FEP) such as VME system. The lowest is a field level bus which links VME and controlled devices. Workstations (WS) or X-terminals are useful for man-machine interfaces. For operating system (OS), UNIX is useful for upper level computers, and real-time OS's for FEP's. We will select hardwares and OS of which specifications are close to international standards. Since recently the cost of software has become higher than that of hardware, we introduce computer aided tools as many as possible for program developments.

I. INTRODUCTION

The SPring-8 facility consists of an 8 GeV storage ring of 1436 m circumference with a natural emittance of 7.0 nmrad, an injector linac of 1 GeV and an 8 GeV synchrotron. Figure 1 shows the layout of the facility. Construction starts in 1990, and the first stored beam is foreseen in 1998 [1]. The construction of control system will start in 1994.

From the designer's viewpoints, the control system consists of the following parts:
1) Computer System;
 1-1) Host and front end computers;
 1-2) Network system;
 1-3) Software;

2) Interlock System;
3) Analog Signal Observing System;
4) Timing System;
5) Television Network System;
6) Links with other System:

II. DESIGN CONCEPTS

Followings are the design concepts of SPring-8 control system:

1) Distributed processors system which are linked by high-speed networks.
2) Sub-system control computers are loosely coupled, due to different accelerator construction schedules and for the convenience of independent operation at maintenance time.
3) In normal operation, all the accelerators are operated at one control room by small number of operators.
4) VME and MAP for front end processors and networks, respectively.
5) PLC (Programmable Logic Controller) for fixed control sequence.
6) Real-time operating system for VME.
7) For high productivity of application program;
 7-1) Object-oriented programming;
 7-2) Computer aided program developing tools;
 7-3) Computer aided operation tools:
8) Standard hardware and software.

(1) 1 GeV Electron / positron linac Build.
(2) 8 GeV Synchrotron Tunnel
(3) Storage Building
(4) Exprimental Hall
(5) Experimental Hall using Radioactive Materials
(6) Laboratories for users
(7) Central Building
(8) Long Beam Lines (1000 m)
(9) Long Beam Lines (300 m)
(10) Control Room
(11) Cafeteria
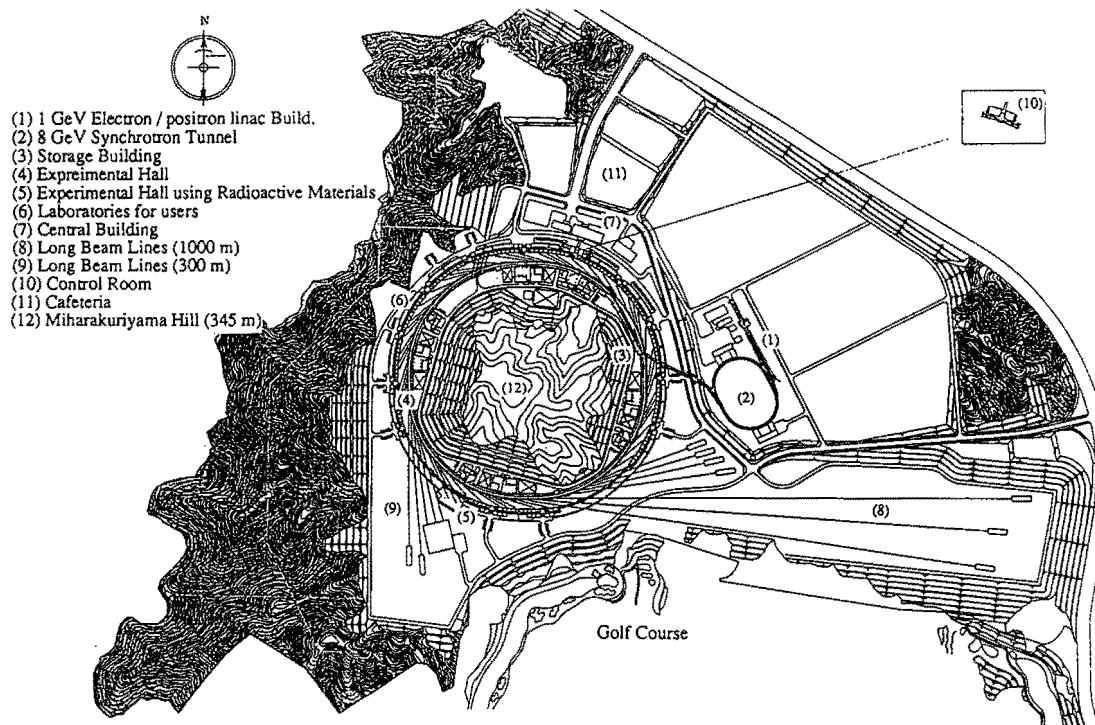(12) Miharakuriyama Hill (345 m)

Golf Course
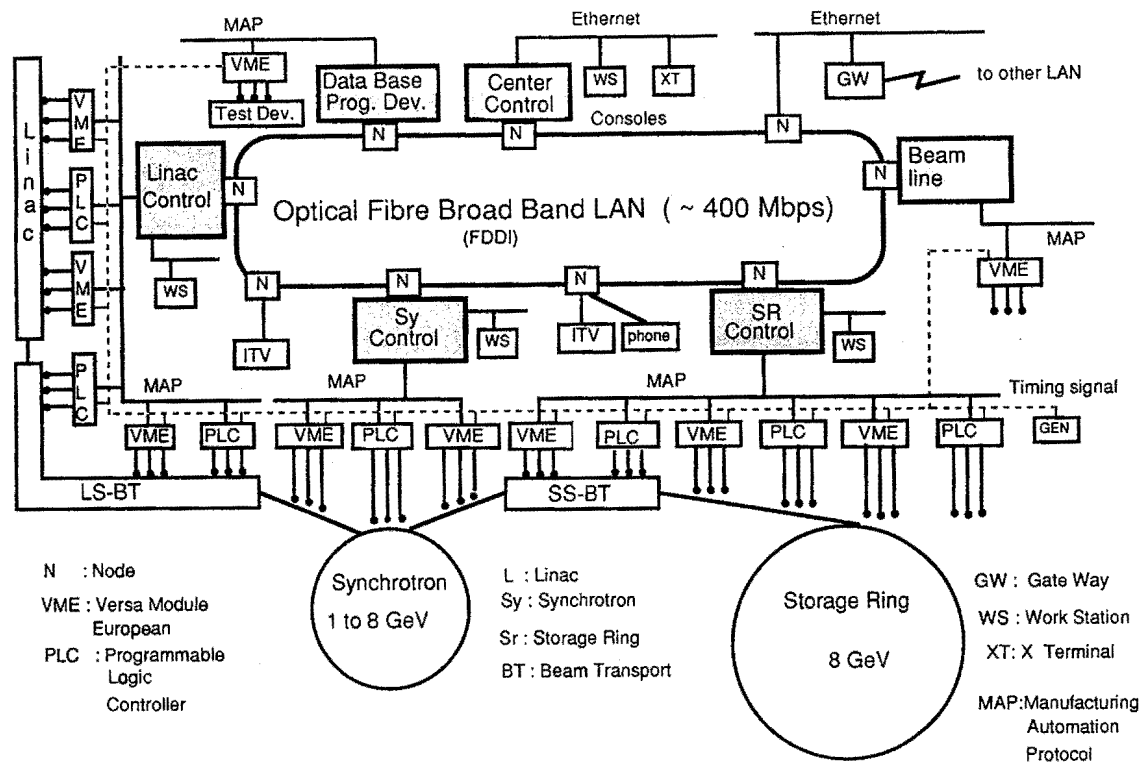
Fig. 1 Layout of the SPring-8.

151

Fig. 2 The computers and their interconnecting networks.

III. COMPUTER SYSTEM

The control system consists of a central control system and several sub-system for each accelerator. These sub-system and central control system are linked through computer networks. Each sub-system consists of a host computer, several front end processors (FEP) and an operator's console. Figure 2 shows schematically the computers and their interconnecting networks. The function of each system are:

A. Central Control System

1) Selection of operating mode and scheduling;
2) Monitoring, logging, and display;
3) Linking with other computer system such as that of the computer center, of experimental system, and of a radiation safety control system.

B. Program development and Database

Central management of application programs and database:

C. Accelerator Control System

The control of injector linac, the synchrotron, and the storage ring. These sub-system controller are loosely coupled, due to the different accelerator construction schedules and the convenience of independent operation and maintenance.

At the lower hierarchy level, some FEP's are linked. These FEP's are microprocessors such as VME and programmable logic controllers (PLC). When high-speed data processing is required, such as by a fast digital feedback system, a VXI system will be used, since the data transmission rate of the VXI bus is much higher than that of VME system.

Workstation (WS) or X terminals are used as operator's consoles, which perform X-window servers.

The FEP's are distributed throughout the power supply rooms and on the circumference of the storage ring building, and the maximum length of the networks will be about 1 km. Table 1 shows the estimated number of input/output points, and the comparison with KAON facility [2]. Where LS-BT means beam transport line between linac and synchrotron, and SS-BT between synchrotron and storage ring. More than 200 VME system are to be used in whole control system.

Table 1 Estimated numbers of input/output points.

S P r i n g – 8

Part	Signals			
	DO	DI	AO	AI
Linac	1848	1128	672	96
LS-BT	62	92	0	37
Synchrotron	588	1057	86	203
SS-BT	725	397	63	142
Storage Ring	9113	15723	1382	3253
Total	12336	18397	2203	3731

T R I U M F

	DO	DI	AO	AI
K A O N	7973	16310	2554	10214

IV. NETWORK SYSTEM

As is shown in Fig. 1, a so called multi-media LAN will be adopted for backbone control, with full- and mini-MAP LAN for real-time control. Voice signals and TV signals are also transmitted on FDDI (Fibre Distributed Data Interface) cables. MAP has two advantages over Ethernet (TCP/IP). One is that messages are exchanged by token-passing method. Hence, even under heavy traffic conditions, real-time response can be maintained. The other is that mini-MAP bypasses the middle 4 layers (3~6) of the 7 OSI layers. Thereby quick response is achieved.

V. SOFTWARE

A. Operating System

An UNIX is used for upper level computers, and real-time OS for FEP's. Table 2 shows examples of commercially available real-time OS[3]. Some of them conform to IEEE POSIX (portable operating system interface for computer environments) standard. An actual performance measurements have been reported on four kernels, all running on the same hardware platform [4].

B. Language

The candidates are Pascal, C++, and Ada which are appropriate for object-oriented programming. C and C++, however, are the main candidates because many program developing tools prepare C library functions as application programming interface. FORTRAN is useful for large numerical calculation programs, such as COD corrections.

For the effective development of application programs, it is convenient to use computer aided software development tools, such as CASE tool and GUI developer.

C. Artificial Intelligence

We intend to use expert systems mainly for alarm message handling system, and partly automatic operations. Introducing an AI tool, "NEXPERT OBJECT", a feasibility study of an expert system for the operation of a test stand of a klystron is started.

VI. TIMING SYSTEM

The common clock (508.58 MHz) is distributed with temperature compensated optical fibers to the linac, the synchrotron, and the storage ring. Trigger request pulses are to sent from the storage ring to the synchrotron and from synchrotron to the linac. Received devices generate beams after accurate delay time. We are developing an accurate delay pulse generator using GaAs preset counter circuit.[5].

VII. R&D

1). Development of accurate timing system for a 508 MHz clock.
   GaAs preset counter,
   E/O and O/E for optical fiber cable with small temperature coefficient.
2). Development of a pattern generator for the synchrotron operation.
3). Development of a field level bus.
4). Control of a klystron test stand.
   HW: EWS, X-terminal, VME, Ethernet,
   SW: UNIX, LynxOS, C, TCP/IP (socket),
   GUI: Xt, ("SL-GMS"),
   AI: "NEXPERT OBJECT"

VIII. REFERENCES

[1] H. Kamitsubo, "8 GeV synchrotron radiation facility project in JAERI-RIKEN SPring-8 Project", Nucl. Instr. and Meth. vol. A303, pp. 421-434, 1991.

[2] W, K. Dawson, R. W. Dobinson, D. P. Gurd and Ch. Serre, "A Conceptual Design for the TRIUMF KAON Factory Control System", TRI-87-1, 1987.

[3] Partly from T. Imai, Nikkei Electronics, no. 517, pp.149-153, 1991 (in Japanese).

[4] K. Low, S. Acharya, M. Allen, E. Faught, D. Haenni and C. Kalbfleisch, "Overview of Real-Time Kernels at the Superconducting Super Collider Laboratory", in 1991 IEEE Particle Accelerator Conference, San Francisco, CA, May 6-9, 1991.

[5] T. Kumahara, T. Wada, H. Yonehara, and H. Yoshikawa, "Design Concepts of The SPring-8 Control System", in 7th Conf. on Real Time Computer Applications in Nuclear, Particle and Plasma Physics, Julich, June 25-28, 1991.

Table 2 Examples of real-time operating system.

OS Name	VRTX32	pSOS+	VxWorks	VMEexec	C EXECUTIV	OS-9000	LynxOS	PDOS
Company	Ready Systems	Software Components	Wind River	Motorola	JMI Softwa Consultant	Microware Systems	Lynx Real-Time S	Eyring Research I
Microprocessor	680X0,80X86 SPARC	680X0 80386	680X0	680X0 88000	680X0 80286	80386,1486	80386,680X0 1860,88000	680X0
Bus	not fixed	not fixed	VMEbus	VMEbus	not fixed	not fixed	not fixed	
Network	Ethernet	Ethernet	Ethernet	Ethernet	Ethernet	Ethernet	Ethernet	Ethernet
Protocol	TCP/IP	TCP/IP	TCP/IP	TCP/IP UDP/IP	TCP/IP	TCP/IP	TCP/IP	TCP/IP
Kernel Size(KB)	8	11.5~ 13	60~ 465	8.5 min.	5~ 10	53	190	25
Task Max. no.	256	65535	65535	No Limit	32767	No Limit	No Limit	
Priority	256	255	255	256	32767	65536	32	225
Multi-Processor Bus	○	○	○	○	×	×	○	○
Network	○	○	○	○	×	×	○	
File System	MS-DOS	Dedicated, MS-DOS	RT-11,UNIX, MS-DOS	Fast File System	MS-DOS	Dedicated, MS-DOS	i-node	
Exec. Time Switching	15μ s	19μ s	17μ s	19μ s	17μ s	220μ s	13μ s	26.2μ s
Interrupt Wait Time	10μ s	7μ s	8μ s	6μ s	NA	40μ s	30μ s	
Measured Condition	68020,25MHz 1 Wait	68020,25MHz no Wait	68020,25MHz no Wait	68020,25MHz no Wait	68020,25MH	80386,20MHz	80386,33MHz	68020,20MHz
AP Language	C,ASM,Ada	C,ASM,Ada	C,ASM,Ada FORTRAN	C,ASM FORTRAN	C,ASM	C,ASM	FORTRAN,C ASM,Ada	FORTRAN,C ASM,PASCAL
Environment for Devlopment	UNIX,VMS	UNIX	UNIX	UNIX	ANY	OS-9,UNIX OS-9000	Self UNIX	Self UNIX,PC

S03SRD12

Design of a Control System of the Linac for SPring-8

H.Yoshikawa, Y.Itoh and T.Kumahara.
Accelerator Systems Development Group.
Office of Synchrotron Radiation Facility Project.
Japan Atomic Energy Research Institute.
Tokai-mura, Naka-gun, Ibaraki-ken, JAPAN.

Abstract

The design of a control system of the linac which is a large scale system including many unstable components like klystrons and modulators. The linac for SPring-8 requires to be operated automatically for injection to the synchrotron. Under these conditions, we chose a distributed control system architecture of a single layer net-work to simplify the protocol of the net-work between the linac, the booster synchrotron and the storage ring. A VME computer of 68030 is put in every modulator of the linac, and all control signals are gathered to the nearest VME computer. OS-9 and OS-9000 are on trial for investigation of the performances. TCP/IP is tentatively chosen as a protocol of the net-work, but we expect that MAP/MMS makes a high performance, and we are preparing a test of it.

INTRODUCTION

We decided that all hardware should be selected from ready-made machines for security of reliability, and our needs is satisfied at low cost without any customizing modules. Because this linac will be used on commercial base, reliability is the most importance for this control system. Moreover, easiness to use is necessary as a user oriented system. This linac consists of 26 pairs of a accelerator section and a klystron, so control signals mainly exist around modulators. Every signal is connected to the nearest VME computer in the modulator. 26 VME computers are connected to the flat network of one layer. Each VME computer works for a modulator, magnet power sources, vacuum pumps, RF phase control and monitors. Software of 26 VME computers are almost same, and it is easy to check whole performance of this system by a test bench of one set.

CONFIGURATION OF THE SYSTEM

A.Hardware

MVME-147s(Motorola) is selected as a CPU board, and it is set in a cage of 20 plots. Digital input/output boards are photo-isolated type, and analog input/output boards are two type of 12bit and 16bit. All signals directly come to the computer through a interface circuit of no CPU, but signals of monitors must have interface devices to establish satisfactory performances.

B.Software

The operating system is OS-9, and the language is C, and partly assembler is used. To select OS, VxWorks, VRTX, LynxOS, OS-9000 and others are check up. OS-9 is selected at the points of reliability, ability of stand alone work without development systems and suitability for bottom up build of a system. Applications and libraries will be made from practical use of object oriented programming.

STRUCTURE OF PROCESS

Every task consists of a control process, file-managers and device drivers. To get higher flexibility, all parameters are described in several parameter-files, and processes have no inner parameters.

A.Communication process

The data format of communication between processes is the same as the format of data through the net-work. If one process send a event (or a signal) to the out of one's cage through a network, the event is received by a communication process. The communication process searches the network address of the cage to which the target process belongs, and it send the event signal as a datagram to the cage. The address of machines are not fixed. The machine address resolution procedure(MAR procedure) defined machine addresses. When a searched process name belongs to a machine, the MAR process of the machine answers its address by broadcast.

B.Logging process

Most of the data sets are logged by a double buffered method. A double buffered method is combination of a ring buffer and a event buffer. The ring buffer stores continuous data of short interval, and if any troubles appear, shifting of the ring buffer is stopped to trace of the origin of the troubles. The event buffer stores log of status for long term.

C.Interlock process

Inter lock signals must be taken by hard wires without computers. But it is able to reduce the wires by a inter lock bus system. In this linac, 26 modulators are almost same, and inter lock signals are classified into different emergency

3rd Int. Conf. Accel. Large Exp. Phys. Control Syst.   ICALEPCS1991, Tsukuba, Japan (partial, pages 194 of 611)   JACoW Publishing

ISBN: 978-3-95450-254-7   ISSN: 2226-0358   doi:10.18429/JACoW-ICALEPCS1991-S03SRD13

levels. The number of hard wires is the number of the emergency levels, and each hard wire is connected in series with same emergency level signals. Inter lock process on the computer sends the detail of what kind of inter lock works to the host computer.

CONCLUSION

Conceptual design of the linac control system is almost finished, and development of the system for a test stand will be started from next June. Coding of software is started already, and now modification of network protocol is going on.

Linac Control System Diagram
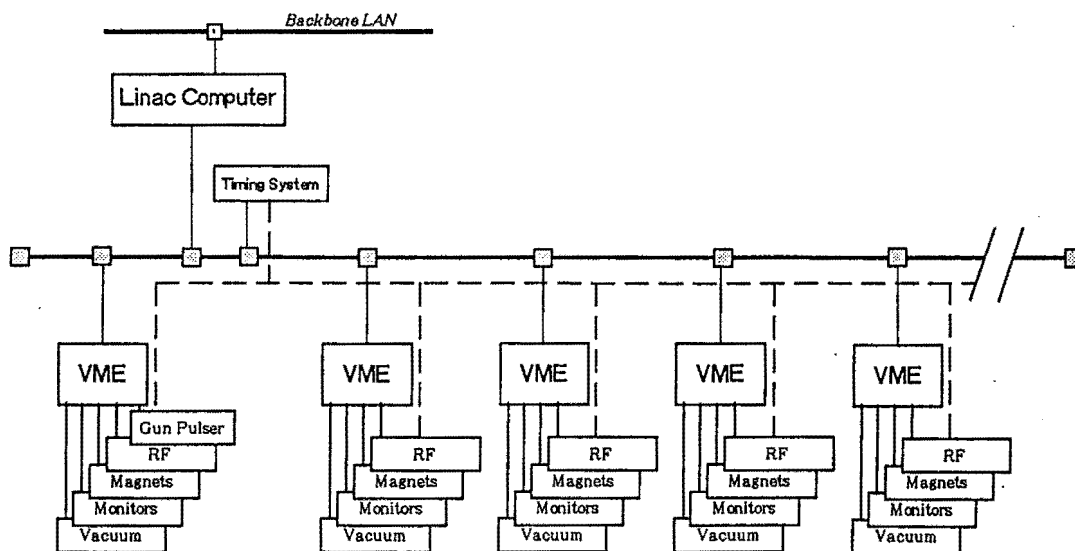
CONTROL SYSTEM FOR HIMAC SYNCHROTRON

T. Kohno, K. Sato, E. Takada, K. Noda, A. Itano,
M. Kanazawa, M. Sudou, K. Asami,* R. Azumaishi,* Y. Morii,†
N. Tsuzuki,‡ H. Narusaka,§ and Y. Hirao

National Institute of Radiological Sciences
4-9-1 Anagawa, Chiba-shi, Chiba 260, Japan

Abstract

A control system for HIMAC synchrotron has been designed. The system consists of a main computer, console workstations, a few small computers and VME-computers connected via Ethernet. The small computers are dedicated to the control of an injection line, an extraction line and an RF system. Power supplies in main rings are controlled by the VME-computers through FDI/FDO, DI/DO modules. This paper describes an overview of the synchrotron control system.

INTRODUCTION

HIMAC is a heavy-ion accelerator complex for the clinical treatment of tumors and now under construction at National Institute of Radiological Sciences. It consists of an injector, a synchrotron, a high-energy beam transport and an irradiation sub-systems. Heavy-ions with a charge-to-mass ratio as small as 1/7 are accelerated up to 6 MeV/u through an RFQ and Alvarez linacs and injected to the synchrotron sub-system.

The synchrotron sub-system has an injection line, an extraction line and a pair of separated function type synchrotron rings with almost the same structure. These rings operate independently at different energies and same ion-species except that power supplies of two rings are 180° out of phase each other. The output energy of each ring is designed to be variable in a range of 100 - 800 MeV/u for ions with $q/A = 1/2$. The general description about the HIMAC synchrotron was given in the previous article[1].

An overall control system for HIMAC consists of a supervisor computer and four sub-system control computers connected via Ethernet as shown in Fig.1. The supervisor computer is used for the global control of the whole system of HIMAC. It is also linked by hardware and/or software to the other equipments in this facility such as a water-cooling system, an air-conditioning system and a radiation safety system. The sub-system computers control individual devices and carry out programmed sequences for device groups. The control system for the injector was already reported[2]. The control system for the synchrotron is also

*Omika Works, Hitachi, Ltd., Hitachi-shi, ibaraki 319-12, Japan
†Toshiba Corp., Chiyoda-ku, Tokyo 100, Japan
‡Fuchu Works, Toshiba Corp., Fuchu-shi, Tokyo 183, Japan
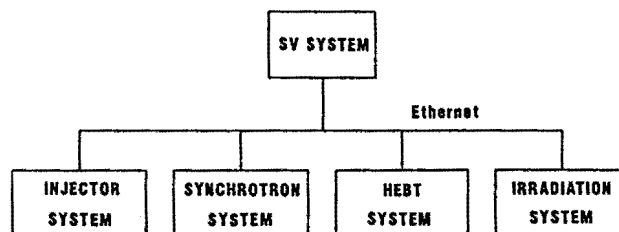§Digital Equipment Corporation Japan, Toshima-ku, Tokyo 170, Japan

Fig. 1: A total control system for HIMAC consists of a supervisor computer and four sub-system control computers.

designed in the same manner, that is, 1) each system is operated rather independently by reducing the data to be transferred each other, and 2) hardware and software concerned with a man-machine interface must be standardized among all sub-systems.

SYSTEM CONFIGURATION

The synchrotron sub-system has many devices of different operational characteristics, for example, dc power supplies in the injection and the extraction lines, high-voltage power supplies in the RF system and the power supplies in the main rings operated with patterns or pulses. In order to handle these devices of different types efficiently and to reduce the load of the main computer, we adopted a distributed and hierarchical structure. A schematic view of the synchrotron control system is shown in Fig.2. Components and their functions are as follows.

A main computer and console computers

A main computer serves mainly as a man-machine interface and a file server. DEC VAX4000/300 is proposed for this purpose under VMS with 64 MB memory, 3 GB disk and communication interfaces for RS-232C, GP-IB and Ethernet. In the HIMAC system, parameters such as current values, current patterns, timing relations etc. are saved as a parameter file in the main computer and referred as the reference data in the next operation of the identical condition. The main computer has to manage this database and carry out programmed start-up and shutdown sequences using these files.

For a man-machine interface two operator consoles are available corresponding to two rings. Two VAX Station
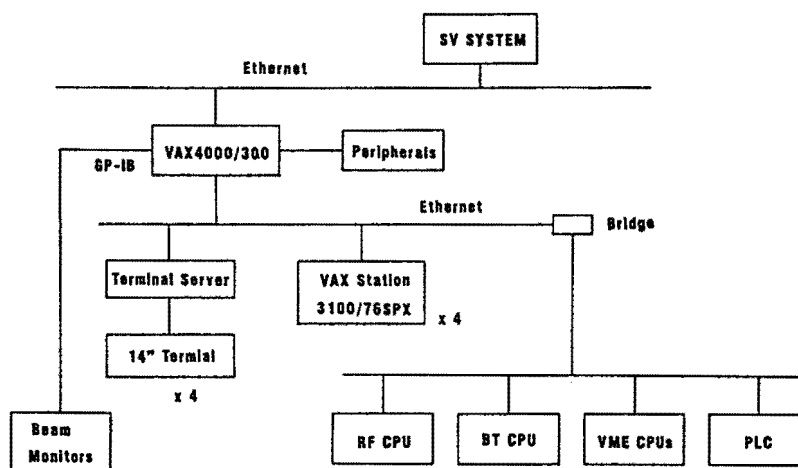
Fig. 2: A control system for HIMAC synchrotron.

3100/76SPXs with 16 MB memory and 104 MB disk and two 14-inch terminals are installed at each console for operation and display. Touch panels and rotary encoders are also equipped at the consoles and almost operations are performed by them with common procedures to the other sub-systems.

RF computer

A dedicated small computer connected via Ethernet controls the RF system including position monitors. Details of the RF control system are described elsewhere[3].

BT computer

The injection line from the injector sub-system and the extraction line to the high-energy beam transport sub-system are also controlled by small computers. Devices in both lines are mainly dc magnet power supplies, beam monitors and vacuum pumps except for a pulse magnet power supply to switch the injection beam to the upper and the lower rings. Device status, parameters and measured data are transferred through Ethernet.

PLC

A Programmable Logic Controller(PLC) is used as a interface between VAX4000/300 and the power supplies in the main rings to communicate ON/OFF commands, status signals and warning signals, although device parameters and measured data are transferred through DI/DO, FDI/FDO modules and VME-computers.

Power supply controller

All power supplies in the main rings are controlled by power supply controllers which consist of micro-computers and Digital I/O, Fast Digital I/O modules based on the VME standard. We plan to use 68000 family CPUs and a real-time, multi-tasking operating system, PDOS.
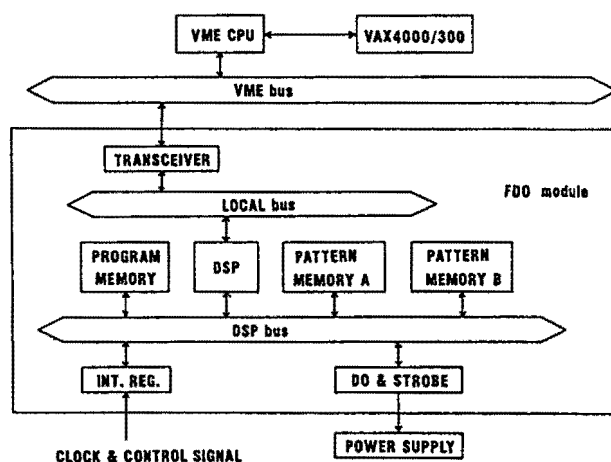

Fig. 3: A blockdiagram of the FDO module. The FDI module has almost the same structure.

The VME-computers communicate with VAX4000/300 via Ethernet and control dc power supplies through the DI and the DO modules.

On the other hand, the FDI and the FDO modules have been developed to control power supplies which should be operated with a pattern, such as a bending, a quadrupole and a bump magnet power supplies. A block-diagram of the FDO module is shown in Fig.3. The FDO module has a Digital Signal Processor(DSP) and two pattern memories(A,B). The DSP sends data from the selected pattern memory to a power supply synchronizing with an external clock of 1200 Hz. While the DSP is reading and sending the data on the selected memory, new data can be written on the other memory, and then the memory is switched quickly from one to the other. The FDI module has almost the same structure and functions. This func-
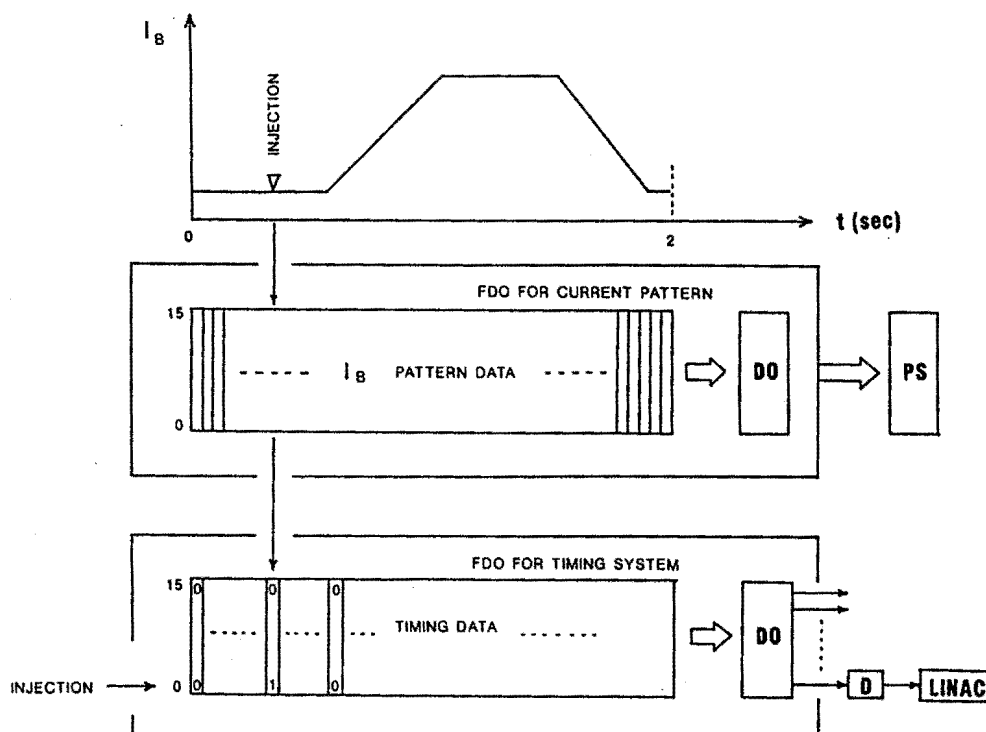
157

Fig. 4: A schematic view of the timing system. The FDO module is also used here.

tion plays an important role for a repetitive feed-forward pattern control of the bending and the quadrupole magnet power supplies in the main rings. In the feed-forward pattern control, the FDO module sends the reference data and the FDI module receives the measured data. Then the VME-computer calculates the correction pattern from the difference between the reference and the measured data using a transfer function of the magnet system and writes the new pattern on the background memory while the foreground memory is facing the power supply.

TIMING SYSTEM

A timing system is very important to synchronize the sub-systems because each sub-system is designed to be operated rather independently except for a software interlock among the sub-systems. Event signals are generated in the synchrotron sub-system and delivered to the own and the other sub-systems by hardware. As the bending magnet power supply consists of 24-pulse thyristor rectifiers and must be operated in synchronization with the ac power line of 50 Hz, the timing system generates clock signals of 1200 Hz (50 Hz × 24 pulses) phase-locked to the ac line voltage. The FDO module is also used for the setting of the event signals. A schematic view of the timing system is shown in Fig.4, where the synchrotron is operated, for example, with the repetition rate of 0.5 Hz. The FDO module for the current pattern data of the bending magnet has 2400 data of 16 bits and they are transferred in synchroniza-

tion with the clock of 1200 Hz. The FDO module for the timing system has the data of the same size. In this case, however, not the 16 bit data but the rows of 0th - 15th bit have meaning because each bit is assigned to a specific event, for example, 'BEAM INJECTION', 'RF ON', 'ACCELERATION START', etc. The event signal corresponding to each bit is generated at the time when the bit is 'ON' or '1' and delivered through a delay controller.

REFERENCES

(1) K. Sato *et al.*, "Heavy-Ion Medical Accelerator in Chiba(HIMAC)", Particle Accelerators, 33, pp.147-152, 1990.

(2) T. Kohno *et al.*, "Control System for HIMAC Injector", Proc. of 7th Symp. on Accel. Sci. & Technol., RCNP, Osaka, Japan, 1989, pp.246-248.

(3) M. Kanazawa *et al.*, "RF Control for HIMAC Synchrotron", in these proceedings.

Digital Control of the Superconducting Cavities
for the LEP Energy Upgrade

G. Cavallari and E. Ciapala
CERN, 1211 Geneva 23, Switzerland

Abstract

The superconducting (SC) cavities for the LEP200 energy upgrade will be installed in units of 16 as for the present copper cavity system. Similar equipment will be used for RF power generation and distribution, for the low-level RF system and for digital control. The SC cavities and their associated equipment however require different interface hardware and new control software. To simplify routine operation control of the SC cavity units is made to resemble as closely as possible that of the existing units. Specific controls for the SC cavities at the equipment level, the facilities available and the integration of the SC cavity units into the LEP RF control system are described.

I. INTRODUCTION

The RF system for LEP phase 1 consists of a total of 128 RF accelerating/storage cavity assemblies operating at 352 MHz, providing a total circumferal voltage of 400 MV. The cavities have been installed around interaction points 2 and 6 in the form of eight individual RF units. Each consists of 16 cavities, two 1 MW klystrons, DC high voltage power converter, low-level electronics and controls. For the LEP200 energy upgrade to 90 GeV, requiring 2000 MV, a further 192 superconducting (SC) cavities will be installed in 12 new RF units, two at each of points 2 and 6 and four at each of points 4 and 8. The RF frequency for the SC units is the same as for the copper units and as far as possible they will use identical equipment i.e. klystrons, power converter, low-level equipment and controls.

The SC cavities are housed in groups of four in a common cryostat to make up an "RF module". Initially only one klystron will be installed per unit but there is provision to add a second for higher beam intensities.

Control of the SC cavities and associated equipment is based on the same principles as for the copper cavities with the same type of interface equipment and software. To render overall operation as straightforward as possible differences between the two types of RF unit are taken into account by local software. Unlike the copper cavity units the SC units can be run in different configurations e.g. four, eight, 12 or 16 cavities and with either one or two klystrons per unit. The configuration must be taken into account by the local software. With two klystrons SC cavity units could be operated more effectively as two "sub-units" of eight cavities and one klystron sharing the common HV power supply. This option is allowed for in the hardware layout and software.

At present one unit with 12 SC cavities is operational in LEP and further units will be installed gradually up to the completion of the project, planned for the beginning of 1994.

II. CONTROLS AND INSTRUMENTATION FOR THE LEP SC CAVITIES

Within the RF unit the various pieces of equipment associated with each major element of the unit are grouped together and controlled by a G64 bus standard based 'Equipment Controller' (EC). As for the copper cavity units there is one EC for each SC cavity but for each RF module there is an additional EC for cryogenics data. The EC is of modular construction and maximum use is made of a small range of interface cards. Standard modules are used to interface the following equipment :

Cavity tuning systems,
RF power measurement,
RF window and helium gas return heating,
HOM coupler fundamental mode power measurement,
Helium gas pressure measurement,
Helium gas valve control,
Cryostat insulating vacuum,
Cavity vacuum.

Interlock protection systems exist to switch off cavity tuning, RF or the HV power converter in the event of a fault or unsafe condition. These systems are largely contained within the ECs in the form of standard interlock modules and G64 readout interfaces which provide fault status and record trip sequences. For the SC cavities a "beam dump" interlock has been added. In the event of very high helium gas pressure or low liquid helium level, RF is switched off in all RF units.

For temperature measurements in the cavity a dedicated module inside the cavity EC measures up to 32 temperatures from signals from Pt100 sensors at various critical points. Conversion of voltage levels to temperature values in degrees Kelvin is done by the EC. Independent hardware logic inside the module triggers RF or tuner interlocks in the event of a change outside preset levels stored in EPROM. Fault information is stored and can be read via the the G64 bus.

A dedicated module in the cryostat EC provides a readout of liquid helium level in the cryostat. This, together with the gas pressure readings and RF levels, is made available via hard wired links to the regulation system of the cryogenics plant.

The design of the above equipment and interface hardware has been finalised and the material for the 180 cavities yet to be installed is being manufactured in outside industry.

S04SRS01

III. LOCAL CONTROL

The control layout for the equipment of the RF unit, situated in racks underground in the klystron gallery beside the LEP machine tunnel, is shown in Figure 1.
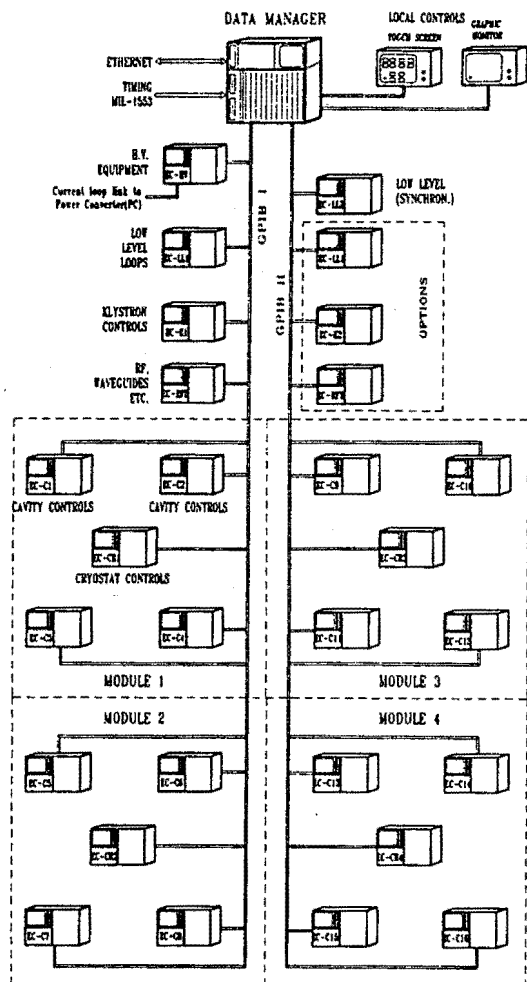

Figure 1 Control Layout within an SC Cavity RF Unit

An EC is dedicated to each major element of the unit i.e. HV equipment, klystron(s), low-level equipment, waveguide system, cavities and cryostats. For the SC units one or two klystrons can be installed, depending on available beam intensity. Second low-level and waveguide system ECs are allowed for to give independent RF operation of two ʳ ᵗht-cavity sub-units if two klystrons are used. Overall co*r* each RF unit is provided by a VME based "Data Mʹ (DM) running a multitasking operating systʳ   ʼ. It communicates with the ECs via two local IEEE   ⌐ (GPIB) buses.

The DM provides remote and local access to all equipment parameters via the ECs either singly or in groups. The DM equipment access functions have been modified for the case of the SC unit in order to cope with the various configuration possibilites. The low-level bus access routines make use of a preset array stored in battery-backed-up memory which indicates active ECs. The DM also executes control procedures involving access to various pieces of equipment, for example HV switch-on, RF switch-on, RF ramping and local surveillance.

For example the RF switch-on procedure for the SC cavity units involves the presetting of the various control loops, setting of initial klystron input drive level and cathode currents, clearing of any RF interlocks, setting the tuning to the correct mode and setting nominal tuning system reference values. After RF switch on klystron drive level and current are slowly increased to a given value of total forward power, sufficient to allow the cavity tuning systems to operate in the presence of LEP beam. As the cavities approach resonance they are slightly detuned by introducing a tuning reference offset to avoid producing full RF voltage and resulting large variation in synchrotron frequency. Once all cavities are in this state the voltage control loop is switched on, the tuning of all cavities returned to nominal and the starting RF voltage set.

While this and other procedures can be different from those of the copper cavity units they are called up in exactly the same way. All equipment and hardware dissimilarities are handled as far as possible by the DM and the ECs.

IV. OVERALL CONTROL OF THE LEP RF SYSTEM INCLUDING SC CAVITY UNITS

The DMs at each point equipped with RF are interconnected by an Ethernet segment and to the general LEP control system Token Rings by bridges.

The principles of operation for the SC units are exactly the same as for the copper cavity units. Direct remote access must be provided for all equipment functions in view of the large amount of equipment and the long distances from the control centre. At the same time maximum use is made of local intelligence at the levels of the DMs and ECs to reduce the amount of accesses required for standard operational procedures, such as switch on and setting up the RF voltage ramp for acceleration. These procedures are initiated by simple commands, the same for both SC and copper cavity units, and are normally carried out simultaneously in all units.

The Apollo workstations used for LEP operation in the Prevessin Control Room (PCR) communicate with the DMs and ECs over the network and local buses. Network communications are based on standard TCP/IP socket library routines[2]. Simple protocols based on command response using strings of ASCII characters are used throughout. The command formats sent resemble their corresponding equivalent ʹC" language functions with additional information denoting the unit and equipment EC to be accessed. The equipment access functions are implemented as simple macro definitions which invoke command response and appropriate data conversion procedures.

For example in a local DM program the "C" language function "s_valve(C9,3,40.0)" will cause the string "s_valve(3,40.0)" to be sent to the EC of cavity 9 requesting it to set helium gas valve 3 to 40 per cent of its range. The

EC carries out the action and returns a reply in string form e.g. "OK" for successful completion. Similarly the corresponding function at the level of the PCR workstation, "LRFSetValve(LRF_233,C9,3,40.0)", causes the string command "C9_RS,0_0,s_valve(3,40.0)" to arrive at the DM of LEP RF unit 233. The command format is compatible with the LEP standard command format intended for communication over the MIL-1553 bus : C9 is the device to be acted on in member 233 of family LRF and RS denotes that a reply in string form is expected. The reply is always in string form and when a numerical reply is to be returned it is converted at the source, the appropriate function being part of the macro definition.

This method first of all allows simple direct command response for remote access to equipment. This low-level access is essential in initial commissioning and has proved to be of value later on for troubleshooting. The command format reflects the logical path to the equipment and the function itself can be clearly understood and easily remembered by the equipment specialist. The underline character in the example given is included in all commands which change equipment states or values to permit a simple equipment protection system.

Secondly, the method of implementing equipment functions as macros based on the same simple direct command/response communication protocols allows straightforward build-up of more complicated software, both at the levels of the DM and the PCR workstations. The total number of functions defined both for the DM and for the PCR workstation is over 300, of which around 50 are specifically related to SC cavity control and data acquisition. The initiation and monitoring of DM procedures is done in the same way, in this case the equipment accessed is the DM itself.

PCR application programs for overall RF operation are based largely on the use of these DM procedures. They are essentially data driven. Use is made of a table called the RF current data set (CDS) which is set up by the operator to indicate to the program which units are to be acted on. The table is stored in the standard Table File System (TFS) format used for LEP applications software data. The RF CDS also contains data such as HV settings and available RF voltages for the various RF units. This means that one piece of application software can handle both SC and copper cavity RF units without the type of each unit having to be checked.

For the copper cavity units a large amount of software, based on interactive graphics packages, has been provided for the RF equipment specialists to show the detailed state of operation of the various pieces of equipment. The preparation of equivalent software to handle SC cavity and cryostat equipment is in progress.

V. SURVEILLANCE AND DATA LOGGING

In the same way as for the copper cavity RF units a DM background local surveillance program gathers the most important states, settings and readings from the SC cavities and other equipment in the RF unit. The data is stored in an operating system "data module" in the form of a structure. The unit configuration must be taken into account by the surveillance program and it deduces this from the contents of the active EC array. The data structure contains approximately 300 integers, strings and floating point values and is updated every 15 seconds. The information is also stored in a cyclic buffer in the local memory of the DM. At present up to one hour of data can be stored, but this will be increased by adding more memory to the DMs. The structure is transferred to the control room on request using TCP/IP protocols directly over Ethernet and Token Ring. It is used for overall logging of RF system performance and also for a permanently updating PCR display of RF system status.

The DM also carries out logging of parameters of special interest for cryogenics. Helium levels and pressures, RF powers and cavity temperatures are measured every 15 minutes by a background program and stored in ASCII form on local hard disc in TFS format. These files can be read remotely using Telnet or copied using FTP.

VI. OPERATIONAL EXPERIENCE AND CONCLUSIONS

The incorporation of the first SC cavity RF unit and its operation with the rest of the RF system has proved straightforward. The SC cavity unit is now routinely used during the running of LEP and its operation is treated in the same way as the copper units. As in the case of the copper cavity units the response time overall of the control system is adequate and its reliability good.

Whilst the software facilities for PCR operation are as complete as those for the copper cavity units, those for the presentation of collective data and detailed equipment states for the specialist await completion. In addition initial running-in experience has shown a need for more rapid monitoring, for example to help in the diagnosis of certain faults or when *in-situ* conditioning has to be carried out. The possibility of logging helium gas pressure and RF level for example and the detection of transients is envisaged. Special instrumentation, connected via the GPIB buses, to measure such parameters will be installed.

The gradual introduction of more SC cavity units will require practically no modifications to existing operational procedures. The control system hardware is finalised and in production and software facilities will be expanded as LEP200 installation progresses.

VII. REFERENCES

[1] E. Ciapala and P. Collier, "A Dedicated Multi-Process Controller for a LEP RF Unit", in Proceedings of the 1989 Particle Accelerator Conference, Chicago, March, 1989, pp. 1583-1585.

[2] E. Ciapala, P. Collier and P. Lienard, "Use of Ethernet and TCP/IP Socket Library Routines for Data Acquisition and Control in the LEP RF System", Contribution to the the 1991 Particle Accelerator Conference, San Francisco, May, 1991.

A PC BASED CONTROL SYSTEM FOR THE CERN ISOLDE SEPARATORS

R. Billinge, A. Bret, I. Deloose, A. Pace and G. Shering.
CERN, PS Division
CH-1211 Geneva 23, Switzerland

Abstract

The control system of the two isotope separators of CERN, named ISOLDE, is being completely redesigned with the goal of having a flexible, high performance and inexpensive system. A new architecture that makes heavy use of the commercial software and hardware available for the huge Personal Computer (PC) market is being implemented on the 1700 geographically distributed control channels of the separators. 8 MS-DOS™ i386-based PCs with about 80 acquisition / control boards are used to access the equipments while 3 other PCs running Microsoft Windows™ and Microsoft Excel™ are used as consoles, the whole through a Novell™ Local Area Network with a PC Disk Server used as a database. This paper describes the interesting solutions found and discusses the reduced programming workload and costs that are expected to build the system before the start of the separators in March 1992.

I THE ISOLDE PROJECT

The ISOLDE project consists of the move of CERN's Isotope Separators and their experimental area from the recently de-commissioned Synchrocyclotron to a beamline served by the Booster Synchrotron [1] [2]. A new control system was required.

Traditionally, control systems for accelerators have been designed based on specified functionality, and the hardware & software tailored to optimize potential utility. Frequently however, this results in 'home-grown' products which remain incomplete and are overtaken by the rapid advances of the massive industrial base of commercial products.

The ISOLDE Project was taken as an opportunity to explore the extreme opposite approach for the control system. Namely to use 'market-leader' commercial software & hardware available for the huge PC market, with in-house development limited to the necessary software & hardware interconnects. This represents an experiment in providing an inexpensive, user friendly control system, requiring a minimum of manpower both for the implementation & for subsequent maintenance.
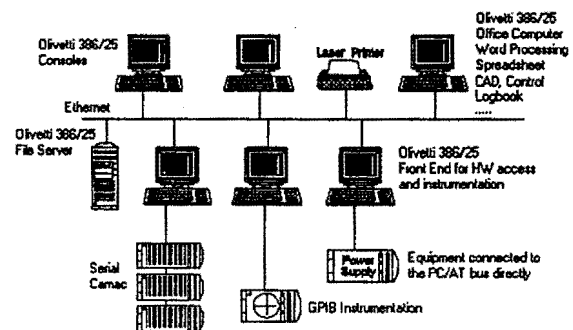
II THE CONTROL SYSTEM ARCHITECTURE

The new architecture [3] [4] [5] being installed reuses the old camac hardware while allowing an evolution of the system towards modern solutions.

Computers, Network and Control boards

The Isolde Control system is simple. It has Olivetti personal computers (PC) at all levels connected using the general purpose Ethernet network available side-wide.

As console in the control room, Olivetti 386/25 are used. i486 based PCs may be introduced next year. The console computers are equipped with 21 inch monitors providing a graphic resolution of 1024 x 768 pixels on 16 colors. Apart from high resolution monitors, the console computers are identical and fully compatible with the several hundred PCs available in the offices as local workstations [6].

The Isolde control system architecture

The personal computers connected to the equipment are called *Front End Computers* (FEC) and are also Olivetti 386/25. The performance of these computers is entirely satisfactory and 80286 PCs have enough CPU power to drive the equipment. In fact for some FEC applications, old IBM AT computers are used, recuperated from the initial Large Electron Positron Collider (LEP) front end controls, now replaced by faster Olivetti PCs. The Front End PCs are identical in configuration to the Office PCs except that they have additional boards for control.

The kind of control/acquisition boards supported in this architecture are:

- CAMAC, to allow the recuperation of all the hardware in the existing control system.
- GPIB to drive sophisticated instrumentation
- Industry Standard Architecture (ISA), alias PC/AT, boards that plugs into the PC directly or in ad hoc extension chassis.

A wide offer of this last type of boards exists and it has, for the moment, being restricted to Analogue to Digital converters (ADC), Digital to Analogue (DAC), Digital Input and Output (DIO), timer interrupts and external interrupt boards and RS232.

Operating Systems and File Servers

The whole system is built on a commercial PC network, Novell Netware™/386. This network provides a shared file system, shared printers, peer to peer and peer to server communications that are used in the control system to share databases and to share the hardware equipment between different consoles. The FECs are in fact just seen from the consoles as *equipment servers* to which client requests can be sent.

These FEC, alias hardware servers, are running MS DOS and the Nodal for DOS program. The Nodal for DOS program is an environment that provides:

- An equipment server to all consoles. The built-in network listener allows the consoles to access the equipment attached to the FEC.
- A local nodal interpreter that allows access to the equipment from the FEC. This is used mainly by the equipment engineers to test modules locally.
- Local alarm treatment

The Netware file system can be mounted also from Macintosh via Appletalk as an Appleshare file server and from Unix workstation via TCP/IP as a NFS server. All the Isolde databases or documentations in Word™ or Excel™ format can then be retrieved.

Numbers, manpower and financial resources
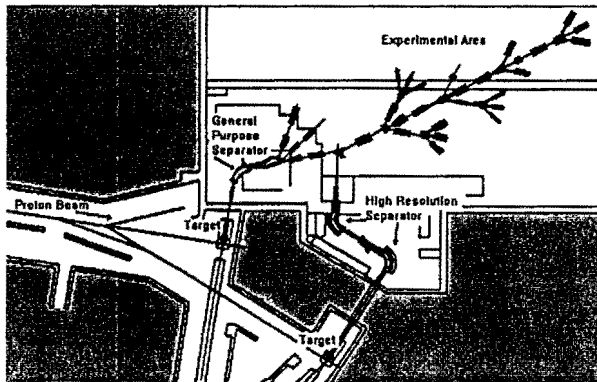
The CERN Isolde Facility

To quantify the dimension of the project, the control system comprises: 2 Faraday cage 60 Kv power supplies, 6 Extraction electrodes, 27 Faraday Cups, 5 Slits, 5 Lens Collimators, 15 Beam Scanners, 5 Wire grids, 6 Thermocouples, 18 Target power supplies in 60 Kv faraday cage, 9 Beam gates, 3 Separator Magnets with gauss meters, 58 Quadrupoles, 25 Steering Quadrupoles, 8 Kickers, 7 Benders, 4 Correction Plates, 3 Multipoles, 5 Deflectors and the Vacuum system. This gives roughly 300 devices (elements) and 1700 analog or digital wires (control channels) coming into the control system on three buses (CAMAC, GPIB, ISA).

More than 80 ISA (PC/AT) boards are being installed in different PCs or PC expansion chassis. 8 FECs are necessary, controlled by 3 consoles in the control room (or any office PC).

The manpower necessary to design and install the whole system is about 3 man*year.

Costs are equally divided between acquisition boards and computers. About 100 KCHF have been invested in PCs and the same amount in ISA boards. Wiring costs, 50 KCHF, are not included.

III BASIC TOOLS

Graphic User Interface (GUI)

The graphic user interface is Microsoft Windows™ and it provides a common access to all applications. The shell to start the different applications is the default *Windows Program Manager.*
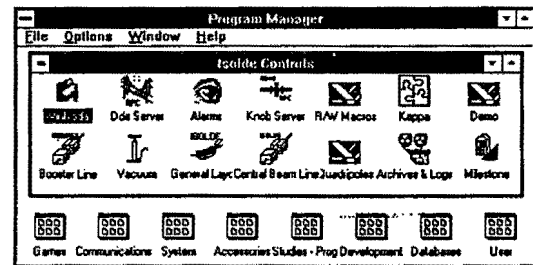
The program manager - Icons and menus oriented shell

Documentation

All documentation is produced using Microsoft Word™ for Windows, and is available on-line on the Isolde Server.

Databases

Databases are built using Microsoft Excel™. Database are stored in the native Excel format (BIFF) or in *Comma Separated Value* (CSV) format when they must be accessible from C or Nodal programs. The system is entirely database driven using Excel.
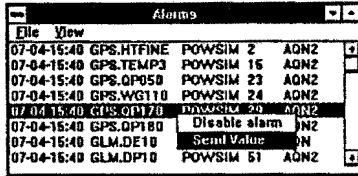
On-line databases

Alarms

An alarm process is running in all Front End Computers and systematically checks for inconsistencies in the element status or in the difference between acquisition and control values.

The alarm process has a list of active alarms that are polled by the alarm program running in the console. The Alarm program in the console is normally minimized in a green icon that becomes red when there are active alarms.

The Alarm eye

Applications

Three environment are available to develop applications. The first is Windows Nodal that provides a Nodal language interpreter with graphics capabilities that emulates the existing Isolde console.

Using Nodal for Windows it is possible to run all old applications developed in the Isolde history. The existing isolde console is made of a keyboard, a track ball, an alphanumeric display, a graphic display, two hardware knobs and a touch panel. Nodal for Windows emulates all these peripherals and permits reuse of the old software (see next section).
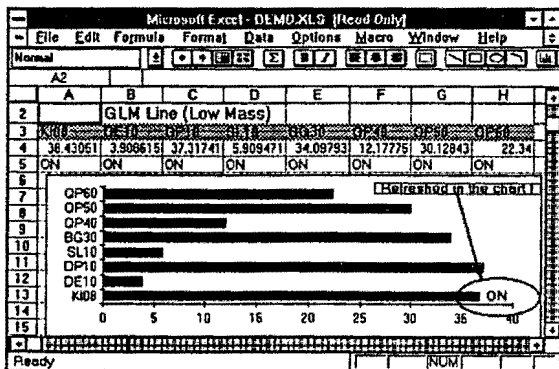
The second environment where control applications can be developed is in any application that supports Windows' dynamic data exchange (DDE). Several high level applications with programming facility could be used, for example Actor™, Toolbook™, Microsoft Word™ and Microsoft Visual Basic™. The recommended tool of this level is Microsoft Excel™. The Spreadsheet facility can be used to correlate physical quantities to control parameters and vice versa.

An Excel based application

The third environment where control applications can be produced is the Microsoft Windows Software Development Kit™ (SDK). Any application can be written in this environment and high performance and acquisition rates can be achieved.
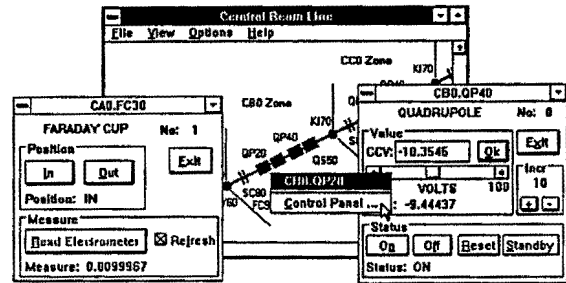
An SDK application

Security

Any application accessing the hardware must be attached to the Isolde Disk server where the client applications are stored. To attach a Netware file server a username and an encrypted password must be provided. In this way the Isolde Control System inherits all the security features that the Netware environment provides that are widely used by banks and other commercial Networks. In particular, it is possible to account the usage, to restrict the hours or days of the week a particular user can attach, to limit concurrent connections, force systematic password changes and several other features.

IV DEVELOPMENT TOOLS

Windows Nodal

The Nodal for Windows interpreter allows to run all existing Isolde software and permits the development of new applications. Interaction with the user is done using the two knobs and the touch panel. Data are presented in the alphanumeric and graphic displays.
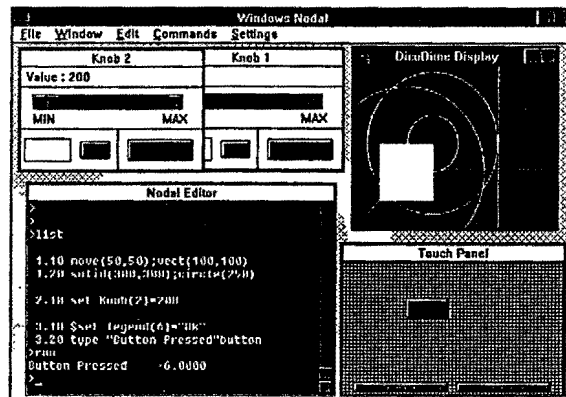
Nodal for Windows

Programming in this environment does not require any particular skill beside having read the Nodal manual.

Microsoft Excel™

Excel in addition to its database facility, provides spreadsheets, charting, database and advanced macro programming facilities.

The spreadsheet facility provides the possibility of having hot-links, i.e. dynamic links between a cell in a spreadsheet and a particular property on a particular equipment, for example the acquisition of the current in a magnet. The spreadsheet cell is updated automatically whenever the physical acquired value changes. Formulae can be used to calculate from control parameters acquired with hot-links any physical quantity related to the parameter. All calculated values are, of course, also automatically updated when any quantity in the control system changes.

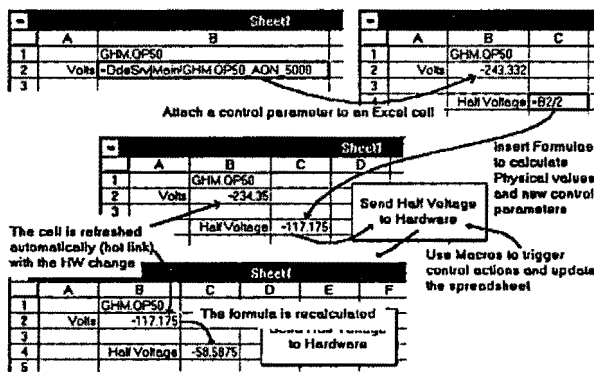

Example of application development using Excel

Basic macros, usually written automatically by the Excel Macro Recorder, allows to execute complicated algorithms. When attached to push buttons on the spreadsheet, macros allow to trigger complicated measurements, start optimization processes or to write calculated values to the hardware.

The Excel charting facility is used to produce graphics for analog presentation. Charts are normally also dynamically linked to the hardware properties and dynamically updated in real time.
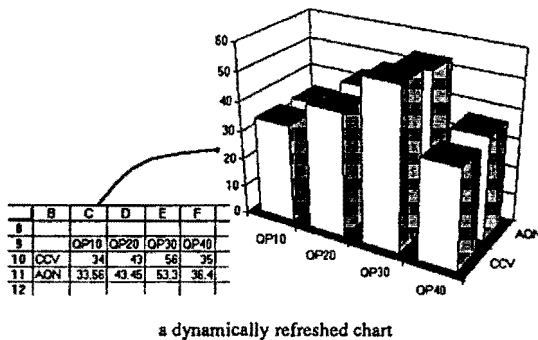
a dynamically refreshed chart

An Excel course for beginners is enough to allow a physicist to start building the applications himself.

Advanced Excel features can be used to produce applications with their own pull down menus and dialog boxes. The Excel Dialog Editor can be used for this purpose.

Windows Software Development Kit™ (SDK)

There are different tools available to design Windows applications, for example, the Dialog, font and Bitmap Editors, or the CodeView Debugger for Windows.
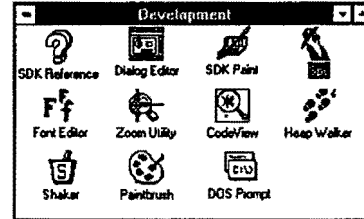
The SDK tools as example.

Writing applications using the SDK gives very high performances but a good knowledge of the C programming language and of the Windows architecture is necessary.

Microsoft Quick C™

The major tool used to develop Equipment modules is Microsoft Quick C. Beside providing an interactive Edit-Compile-Link-Debug environment as if C was interpreted, it has also on-line the complete C language reference. This context sensitive hypertext documentation provides also ready to run examples of all C library functions.

V CONCLUSIONS

Simplicity

Using the same type of computer (ISA PC compatible) at all levels of the control system leads to uniformity and simplicity. This advantage is reinforced by using the same operating system as the office machines.

All the persons involved in the design of this control system master it completely and are able to solve any problem at any level, front end or console. This global view on an uniform system reduces the skill necessary to cope with different parts of the system, because the approach is always the same, data driven from the Excel Database.

Development time

The use of well tested, market leader products (hardware and software) has reduced the development to a strict minimum. The very few necessary development done with the available tools have shown that:
- Development is very quick because of the user friendliness of the tools and because the SDK, the Microsoft C and QuickC compilers offer

- incremental link and compilation options and run time dynamic link.
- Informatic skills are not necessary: Physicists and engineers can write their own programs after very little training.

In particular, the time to develop the application programs, normally the stumbling block in most control system is very significantly reduced: for example, it does not take more than 10 minutes to build a simple application using Excel to display graphically several parameters acquired from the separator.

Integration

The Isolde control system, although completely independent in case of problems, is transparently connected to the office network and all its services are available.

Beside providing an access to the Isolde control system from nearly 1600 points at CERN (the number of PCs installed), it is in the user culture by providing the same user interface to control applications as to any administrative or scientific applications. It is of course possible to Copy/Cut/Paste between any applications, for example, the Excel chart acquiring the status of the Isolde magnets can be pasted into Microsoft Word to produce a written report. Development of control application from any office is possible without the installation of additional hardware.

All A3/A4 laser printers (including the color ones), plotters, scanners, disks, software and data of the office network are available, including the support.

By relying on this infrastructure, the information can be easily distributed cern-wide as Statistics and/or electronic Logbook, as it is done now for the PS accelerator complex. As the file base is shareable between different platforms and operating systems (Macintosh, Windows, Unix), all data in Excel format can be retrieved from any user on the site.

Speed

The benchmarks done with the Olivetti 386/25 computers have shown that these computers are fast enough to control large processes where dozens of parameters from different FECs are involved. The margin can be enlarged in this area by using i486 based 33 MHz machines that are much faster.

The speed of the control system is often limited by the network used. The Novell network based on ethernet give us a performance of more than 100 Kbytes/second transfer rate on the busy CERN ethernet. A typical Remote Procedure Call with a search in the element database to find which FEC and which equipment module to call takes less that 30 milliseconds.

Cost

The market of Personal Computer in the world is roughly of 80 Millions units and with a strong competition between

manufacturers. The PC production is more than 10 Millions/unit per year with performances increasing by a factor of 2 every year.

It' s a world wide standard, with binary compatibility, with major European involvement such as Olivetti, Philips, Bull and Siemens. A chassis, made in Europe, to control 48 power supplies (48 ADCs, 48 DACs, 288 digital I/O) cost about 12 KCHF. The cost per control channel is 9 CHF per digital bit, 55 CHF per ADC channel and 155 CHF per DAC channel.

Limits

There are no limits to the numbers of control channels the system can handle. If more are necessary, just add more FEC and more consoles.

A typical console screen

VI REFERENCES

[1] D.J. Simon, "Transfer of Isolde to the PS Booster", CERN PS/PA-EP Note 90-13

[2] B.W. Allardyce et al, "Isolde at the PS Booster", to be presented at the European Particle Accelerator Conference, Berlin, March 1992

[3] O.C. Jonsson et al, "The Control System of the CERN-Isolde on-line Mass-Separator Facility", EMIS-12, Sendai, Japan, September 1991. To be published in Nucl. Instr. and Meth. B

[4] A. Bret at al, "The Isolde Control System", CERN PS/OP/Note 91-27

[5] A. Bret at al, "The Isolde Control System Reference", CERN PS/OP/Note 91-17

[6] A. Pace, "Office Local Area Network at the PS Division", CERN PS/91-45 (OP)

STATUS OF THE CONTROL AND BEAM DIAGNOSTIC
SYSTEMS OF THE CRYRING PROJECT

J.Starker and M.Engström for the CRYRING project group
Manne Siegbahn Institute of Physics
S-104 05 Stockholm, Sweden

Abstract—CRYRING is a facility for research in atomic, molecular and nuclear physics. It uses a cryogenic electron beam ion source, CRYSIS, together with an RFQ linear accelerator as injector into a synchrotron/storage ring for very highly charged, heavy ions. The first circulating beam was achieved in december 1990. The status of the systems for control and beam diagnostics are described.

INTRODUCTION

The CRYRING project [1] is centered around a synchrotron/storage ring of maximum rigidity 1.44 Tm, corresponding to an energy of 24 MeV per nucleon at a charge-to-mass ratio $q/A = 0.5$. It is mainly intended for highly charged, heavy ions produced by an electron-beam ion source (CRYSIS).

Light atomic or molecular ions can also be injected from a small plasmatron source (MINIS). Ions from the ion sources are accelerated electrostatically to 10 keV per nucleon and transported to a radiofrequency-quadrupole linear accelerator (RFQ) which brings them to 300 keV per nucleon. The ions are inflected electrostatically into the ring where they are accelerated using a driven drift tube. The stored ions will be cooled by an electron cooler. Fig. 1 shows a layout of the CRYRING facility.

The control system is based upon the LEAR (Low Energy Antiproton Ring) control system at CERN [2]. The principles of the system, the main part of the software and some parts of the actual hardware implementation are copied from the LEAR system. A substantial amount of development work has nevertheless been put into the CRYRING control system in order to adapt it to our operational needs which are partly different from the ones at LEAR.
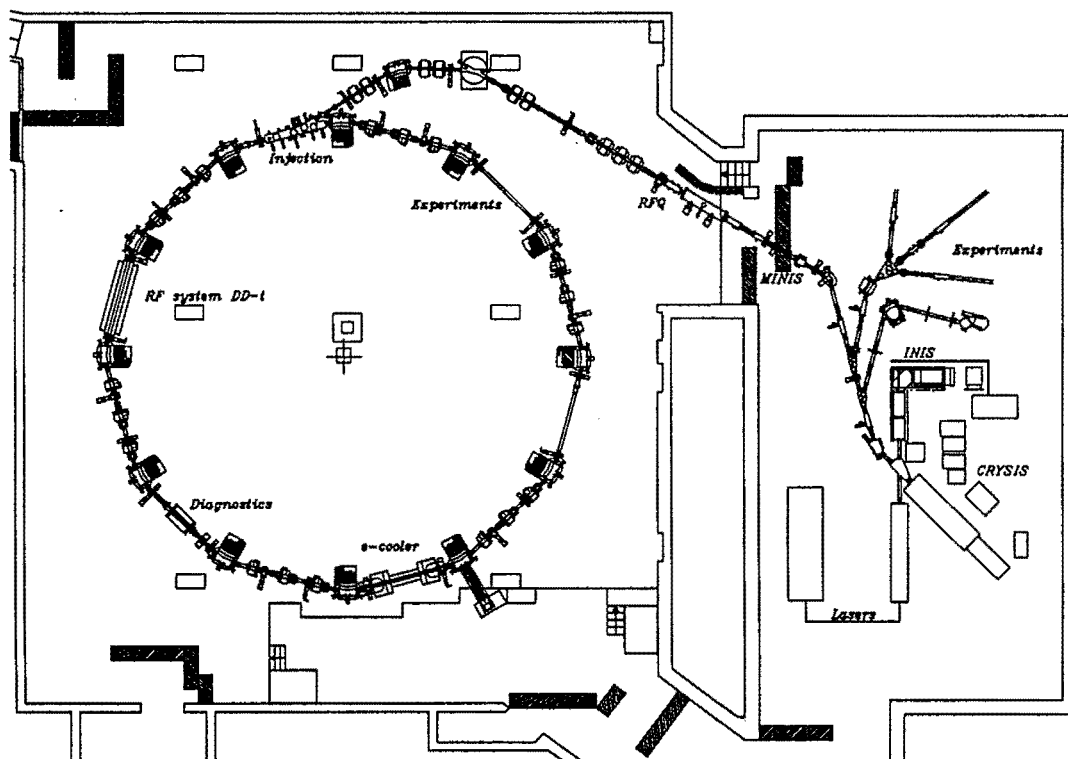

Fig 1 CRYRING layout.

CRYRING is equipped with different diagnostic elements to measure the low energy ion beam profile and current in the injection lines, to monitor the beam properties in the ring and to control the frequency of the accelerator structure keeping the beam centered in the beam tube.

THE CONTROL SYSTEM

The control system is based upon the LEAR control system at CERN. The architecture of the system, as well as the main part of the software and some parts of the actual hardware implementation are copied from the LEAR system. The development work in the CRYRING system has mainly been done in two areas. One is the low-level interfacing to our accelerator equipment using microprocessor systems with software written to allow local control for test purposes and thus make trouble-shooting easier. The second is the adaption of the software in the main computer to the actual hardware and to our operational needs which partly differ from the ones at LEAR. The system controls the whole CRYRING complex, that is the ion sources, the beamlines and the ring. The electron cooler, which will be installed this winter, will also be controlled by the same system. The general structure of the control system is shown in fig. 2.

Structure of the system

The main components of the systems are:

- The main computer, a PDP-11/73, with terminals and two operators consoles.

- A serial CAMAC loop for distribution of data and timing.

- A large number of G-64 microprocessor systems for interfacing to the machine equipment.

A piece of accelerator equipment connected to the system is called a parameter. Analog control signal output to parameters from interfaces in G-64 systems can be of two different types: static, controlled directly by the main computer, or ramped, controlled via programmable function generators in the G-64 systems. These function generators, called GFDs [3], have the shape of the function down-loaded from the main computer via the CAMAC serial loop. They can then be started, stopped, held and released synchronously with each other by pulses from the timing system. The timing system [4] consists of one master and a number of decoder modules, all sitting in CAMAC, interconnected with a timing distribution cable.
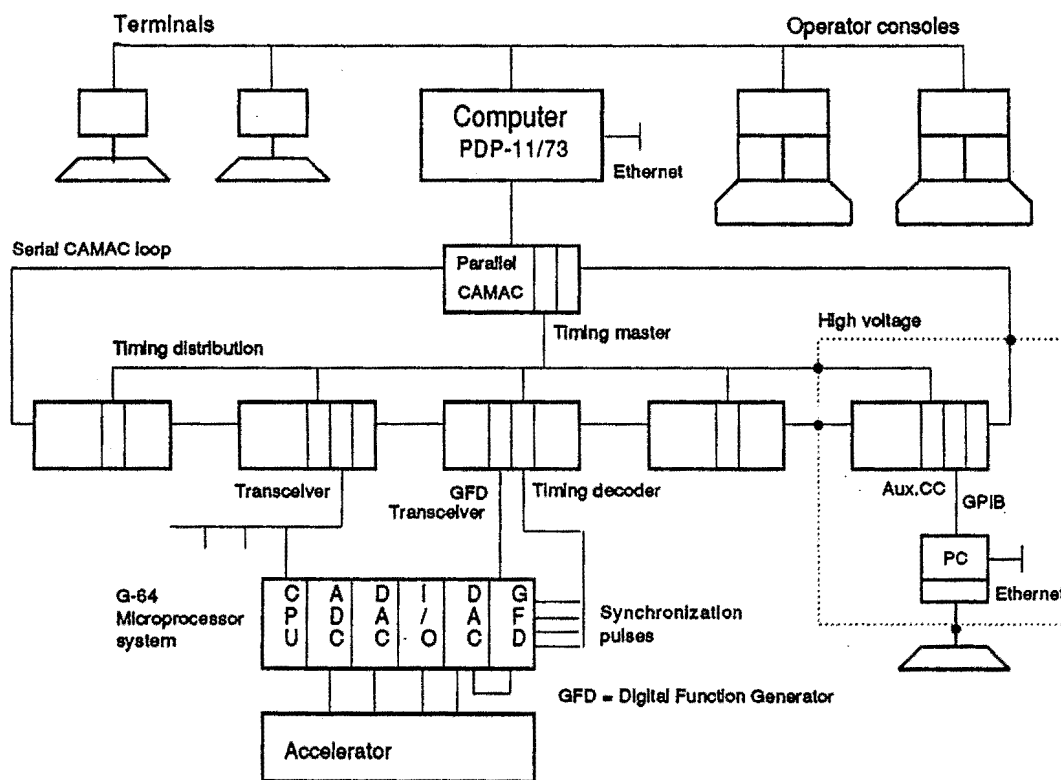

Fig 2 The control system.

The master and decoders are programmed via CAMAC by the computer to execute a predifined set of timing events, a machine cycle. Starting, stopping and repeating the machine cycle is done by external pulse inputs. This means that once the GFDs and the timing system have been loaded the cycling of the ramped parameters is done without need for intervention by the main computer.

The control of the CRYSIS ion source demands special flexibility and quick responses, as well as a more developed local control. These needs have been met by adding a PC that has access to the necessary hardware via an auxiliary crate controller in one of the CAMAC crates. This has requred an optical link between this PC itself and its keyboard, because the PC as well as the mentioned CAMAC crate is put on a 50 kV platform.

Status, November 1991:

The control system

- has been running since the first parts of the accelerator system were taken into operation in 1987.

- presently controls around 160 static and 20 ramped parameters. New parameters are successively added.

- allows control from both control room and equipment areas.

The G-64 systems

- control from 1 to 16 parameters each.

- have hardware and software tailored for their actual applications.

- can be equipped with terminals and run in local mode, thus making it easier to trace faults in accelerator hardware and in main computer software.

Development:

Some upgrades are being considered to improve performance and user interface. These developments are parts of today's control system at LEAR and would be logical updates to the CRYRING system:

- More computing power can be added by network linking to VAX computers at the institute.

- Workstations and PCs are considered as complement to the operators consoles. Operation with graphic presentation, also from equipment rooms and experimental areas, is desired.

DIAGNOSTICS

The beam intercepting devices in the transfer lines are Faraday cups for intensity measurements and strip detectors for beam profile and emittance measurements. Also, chromium doped Al_2O_3 plates viewed through TV-ca/-me/-ras are used here.

The control system moves these devices into/out of the beam, switches TV-cameras to different monitors and the read-out of the beam current to selected instruments.

The signals from strip detectors (each detector consists of 16 horizontal and 16 vertical strips) are amplified, multiplexed and sent to ADC's in a VME computer system placed in the control room, fig. 3.
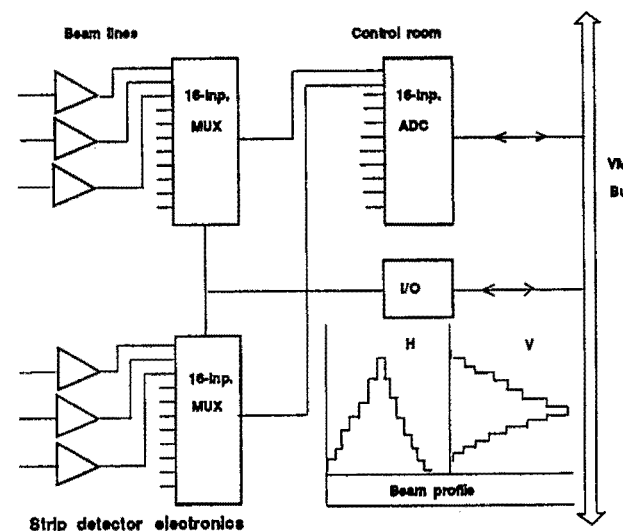
Fig 3 Strip detectors.

In the ring non-destructive measurements are performed using 9 horizontal and 9 vertical electrostatic pick-ups, a beam current transformer and a Schottky noise detector.

The signals from the pick-ups, fig. 4, can be processed by a fast peak-detection system [5] or by using synchronous rectifiers for low-bandwidth measurements. An even faster peak-detection system is now being installed on one pick-up, to allow for measurement over 128 consecutive turns, to study transient behavior of the beam [6].

A high resolution (300 nA) current transformer from Ber/-goz, for measuring the absolute value of the current of the unbunched beam, was installed. So far there has been problems with the signal-to-noise ratio where the noise has been observed to come mainly from the ring magnets . Work is going on to solve this problem. The electronics of the Schottky detector is shown in fig. 5.

Fig 4 Pick-up detection system.

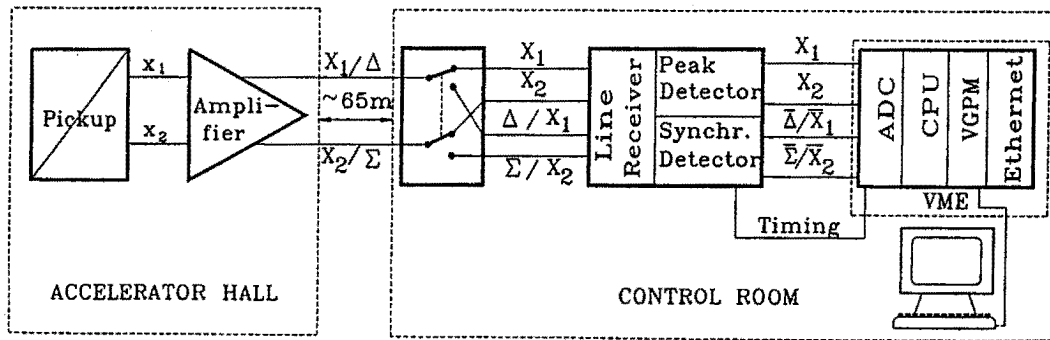The input signals are amplified by high bandwidth (.01-50 MHz) amplifiers and the difference and sum signals are created in a passive circuit consisting of three power splitters. Frequency analysis of these signals can yield the q-value of betatron oscillations and the momentum spread of the beam, as indicated in the figure.

The same result, but much faster, can be obtained by processing the signals in a DSP consisting of a flash-ADC and a FFT processor. It is considered to include this type of module in the VME system. The VME computer is equipped with a GPIB controller which allows control and read-out of auxiliary instruments.

Fig 5 Schottky detector.

References

[1] C. J. Herrlander, K.-G. Rensfelt and J. Starker, "CRY-RING—a heavy-ion synchrotron and storage ring," in *EPAC 88* Rome, June 7–11, 1988, p. 350; K.-G. Rensfelt, "Status of the CRYRING project," in *EPAC 90* Nice, June 12–16, 1990, p. 623; C. J. Herrlander, "CRYRING— a Low Energy Heavy Ion Facility", in *Cooler Rings and Their Applications*, Tokyo, November 5–8 1990, to be published.

[2] U. Tallgren et al., "The distributed computer control system for the CERN low energy antiproton ring (LEAR)," CERN PS/85-27 (LI) 6.5.1985.

[3] P. Lienard, "The LEAR GFD," CERN/PS (to be published).

[4] J. Knott, "The LEAR control timing system," CERN PS/LI/Note 83-12.

[5] S. Borsuk, K. Agehed, W. Klamra, and Th. Lindblad, "A peak detecting pulse height ADC system for beam diagnostic pick-up detectors in the CRYRING accelerator," *Nucl. Instrum. Methods*, vol. A284, p. 430, 1989.

[6] S. Borsuk, "The fast digital peak detector for the beam diagnostic system in the CRYRING accelerator," (to be published).
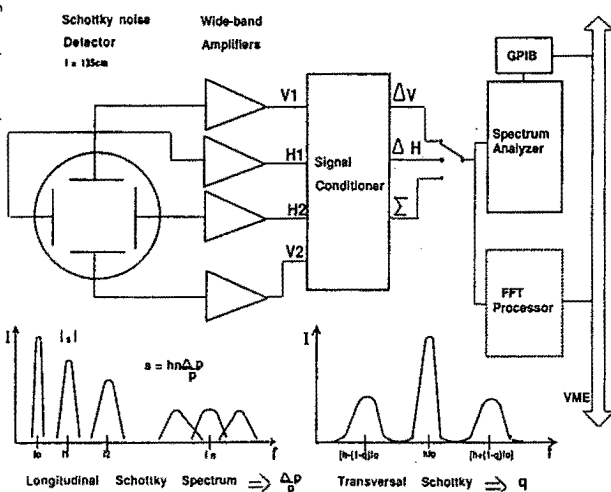
MAGNET TEST FACILITY CONTROL SYSTEM
FOR SUPERCONDUCTING MAGNETS OF UNK

A.I.Agueev, V.N.Alferov, K.F.Guertsev, V.I.Gridassov,
A.A.Gussak, A.F.Dunaitsev, V.A.Krendelev, A.F.Lukyantsev,
V.M.Proshin, V.E.Solovyov, A.N.Sytin, E.A.Ustinov.

IHEP, PROTVINO, USSR.

1. INTRODUCTION

An UNK Magnet Test Facility (MTF) is being constructed to provide cryogenic, electrical and magnet tests of superconducting (SC) magnets of UNK. The main parts of it are:
- The cryogenic system consisting in its turn of the central liquefier, ten satellite refrigerators, two compressors, purification system and transfer lines. The central liquefier supplies the satellite refrigerators with liquid helium. The liquefier is manufactured according to the scheme incorporating precooling by liquid nitrogen, two turbine expanders and a wet expander.
- Four 8 KA, 24 V, ramped Power Supplies (PS) for cold testing of SC magnets, two 3 KA PS's for instrumentation testing and calibration.
- Test facility in its turn consisting of:
a) two dipoles and one quad benches for warm measurements;
b) eight dipoles and two quad benches for cold measurements;
c) two benches for instrumentation.
Relevant parameters and technique are given on table:

	Item	Parameter	Technique
1	Reference and calibration dipoles	Axis field	NMR – method in the central region and Hall – method at the end parts
		Multipoles	Rotating coils
2	Dipoles and quads measurements	Effective length Multipoles	Rotating coils NMR
		Field angle Magnetic axis	Stretched wires
		Dynamic multipoles	Stepwise coil rotation Measurement of transition process after dB/dt→0

The systems for warm measurements of SC coils and cold measurements of quads, including measurements of the magnetic axis location and alignment of the reference target, are being developped jointly with Saclay (France).

Total production rate of facility intended to be 2,4 SC magnets per day.

Such a complex of equipment requires a Control system which provides automatic monitoring, control of equipment, data aquisition and storage into files of magnets. Taking into account the difference between 3 pieces of an equipment (cryogenics, PS's and electrical/magnet measurement stations) Control system is designed as a mixture of 3 different subsystems with different phylosophy, but connected by LAN, sharing the same Host Computers, and based on the same hardware and computers.

2. HARDWARE CONFIGURATION

The hardware configuration is shown on   Figure. PC/AT's were chosen as a suitable computers for real time control of groups of equipment and measuring benches.

PC's present the middle level of computers hierarchy. The upper level is formed by two Host Computers of DEC - family. All computers are connected with Ethernet.

The interface electronics is based on IHEP version of CAMAC. In cryogenics the groups of CAMAC crates communicate with PC's through RS-232 lines, connected to crate controllers. CAMAC crate houses:
- intelligent crate controller with 16 bit, LSI-11-compatible processor;
- convertors for Allen Bradley thermometers;
- convertors for vacuummeters;
- 12 bit ADC;
- I/O - registers;
- restart memory module after power failure/restore;
- thyristor and relay drive amplifiers for Motors and valves;
- intelligent (16 bit, LSI-11-compatible) module for 4 control loops, aimed to provide reliability of the Control system itself as well as the cryo-complex.

Such a distributed processing means that all processors provide local control algorithms, gather local data pool of all related equipement, transmitting information to upper level only on request.

Control of PS's is provided with 16 bit computer, connected to CAMAC. Electrical/magnetic measurement station consists of 2 CAMAC crates, connected to PC, CAMAC crates house:
- 18 and 16 bit ADC, 20 bit DAC, timers, function generators for PS's control;
- amplifiers, filters, comparators for quench detection;
- NMR - convertors, voltage/frequency convertors, step motor drivers and so on for magnetic measurements.

3. SOFTWARE

The purpose of software is to maintain functionality of all subsystems connected with production, testing and filing of magnets. In addition (independently of the serviced subsystems) software must maintain:
- automation of programming that includes the programming of low-level computers;
- communication between all computers that includes loading and start-up of programmes on low level, data exchange, subsystems interface through a middle-level computer;
- filing of the produced superconducting magnets and preparation of all necessary documentation about them;
- support of the bank of programmes that maintain execution of the system tasks.

All software may be devided into the system software and the application one.

The application software carries main functional load and includes algorithms of control, data acquisition, mathematical processing and representation of data that are individual for each technological subsystem. The software toolkit necessary for quick and qualitative writing of application programs is maintained by the system facilities.

The system software is all software that doesn't depend on the object of control and one that is common for various subsystems. It maintains the operational environment for application programs interfacing with operator and equipment and includes the following components:
- operating systems of high- and middle-level computers and their utilities;
- systems of automation of programming;
- software for intercomputer communications (TCP/IP) that includes the communication of high-level computers with the computers of IHEP central complex;
- automation system data base and its utilities;
- the operational bank of superconducting magnets tests results;
- packages of subroutines for organization of the operator dialog with computers, format data conversion and data representation on the information representation equipment;
- multiprogram real-time monitor for low-level computers;
- the tests for system equipment components.

The features of problems solved by high-level computers permit us to use the VAX/VMS operating system that maintains a parallel execution of some tasks in multiprogram and multiaccess modes. We want to use the mode of the RSX-11M operating environment emulation especially in the system of automation of programming of tasks for low-level microcomputers.

We have selected C and FORTRAN as the programming languages for high-level computers. The features of the programming of low-level computers are defined by two reasons: small amount of main storage (56 Kb) and the short response time requirements especially with respect to control programs. The first reason requires elaborate and accurate programming. Second one requires the use of specially developed monitors.

The middle-level computers may be used as the local console for subsystems that need a control.

4. INTENTIONS

a) Analog electronics interlock is being constructed to protect the most important cryogenic equipment in case of Control system default.
b) Cooling of CAMAC crates with help of vortical air refrigerator will be tested.

5. ACKNOWLEADGEMENTS

The author list of this status paper contains only names of group leaders. Much more people really take part in the designing of the Control system.
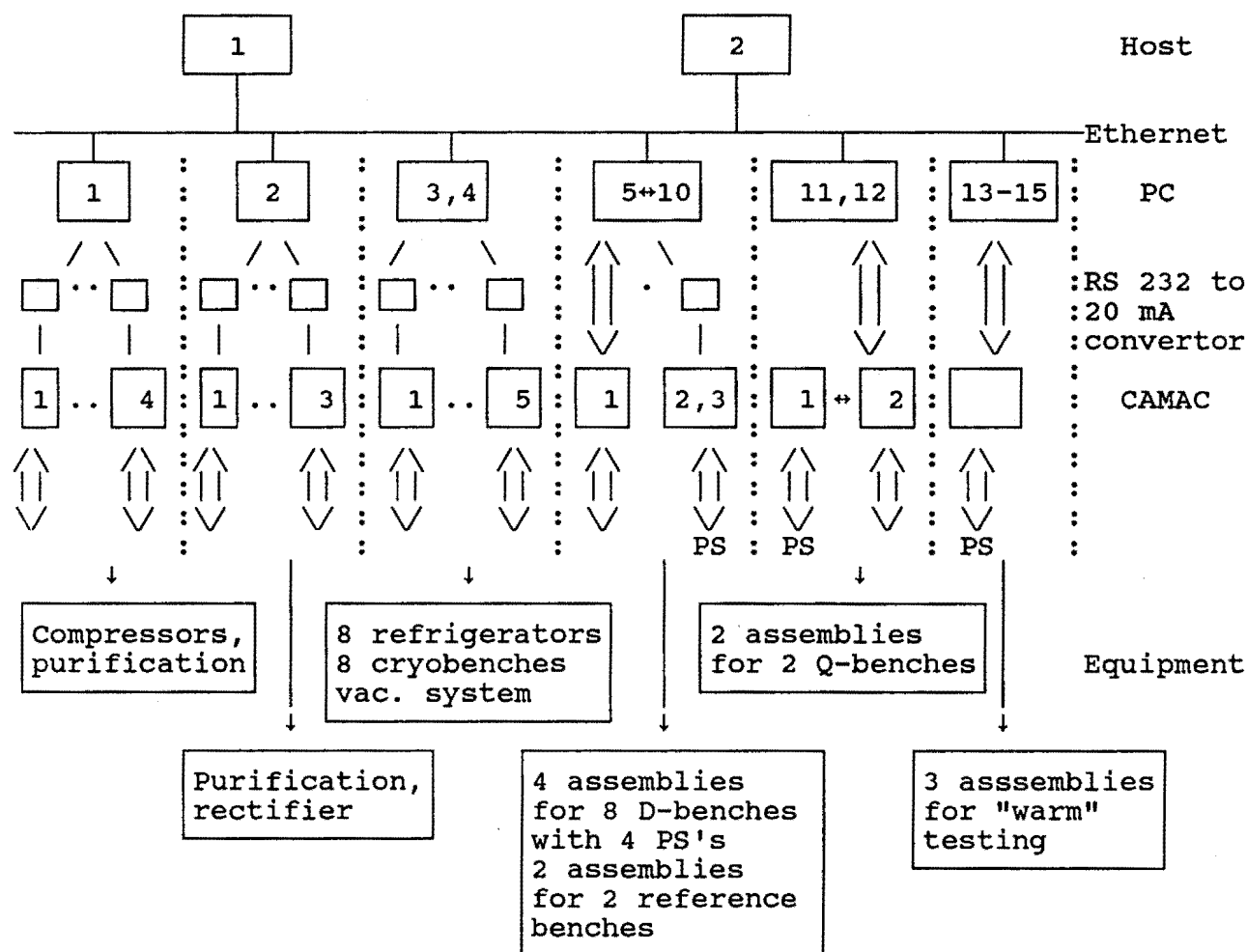
BEAM EXTRACTION CONTROL SYSTEMS
OF THE FAST-CYCLING SYNCHROTRON

A. Toumanian, N. Zapolski, V. Nickogosian, A. Ananian,

A. Kazarian, M. Khoetsian, A. Agababian, A. Matevosian

Yerevan Physics Institute, Armenia

Abstract

A compact system controlling the extraction of different beams (gamma, electron, synchrotron radiation) in single and simultaneous operation modes at high electromagnetic disturbances level based on using one computer of IBM PC/AT type is described.

Introduction

Physical research program at the Yerevan synchrotron pursues the realization of the experiments generally with the use of the slow extraction of primary and secondary beams in single and simultaneous operation modes at 4.5 GeV energy with the 4-8 μs magnetic field top. The most complicated process of the extraction, requiring the precision tuning of the beam extraction devices and not having analog in the world is the mode of simultaneous beam guidance to the two internal targets, one of which is a thin crystal, the other one is of thick tungsten and is put in the neighbouring focusing interval of the synchrotron. At the same time it is necessary to provide a significant decrease of the beam pass factor through the thin target by screening from the particles, once passed through it by means of the thin target [1].

Due to the developed and described below the control system of the synchrotron extraction devices it was managed to increase the ratio of the pick of coherent bremsstrahlung radiation from the thin crystal target to the amorphous part almost 2.5-3 times with keeping unchanged the common requirements to the extracted beam parameters, that is to say to the stable uniformity and duration of the extraction, effectiveness of the extraction and so on.

Secondary beam extraction of the Yerevan synchrotron is based on the local disturbance of the orbit with using the additional electromagnetic coils of the guiding magnetic field. At the beam guidance simultaneously onto two internal targets there is also used a system of changing the betatron oscillation frequencies of the circulating particles with the help of the lenses set on the orbit. To realize the slow extraction of the primary beams to the vicinity of the nonlinear resonance of the third order the conventional system of magnetic elements (quadrupole and sextupole lenses; septum and bending magnets) is used. Magnetic elements and additional coils of the electromagnet are supplied by the current pulses of the complicated shape, produced by the resonance forming lines with the use of the thyristor switches. The tuning of the form and amplitude of the current pulses is realized by means of the face control of thyristor switches with the use of the synchronizing pulses from the synchrotron timer device. The control of the current pulse form and the intensity changes during the beam extraction is carried out by many pickups.

1. Architecture and the control system construction principles

The first control system of the synchrotron extraction device was based on the control computers EC1010 and EC1011 (Videoton firm, Hungary) [2]. But the lack of reliability in their work and the relatively expensive maintenance showed the necessity of replacing them by the modern computers. The computer PC/AT was chosen for that. It determined the architecture of the control system from the one hand and from the other the requirements of reliability and flexibility at high level of the electromagnetic noises were satisfied by having an intensive information flow, a large number of the control parameters and so on. That's why a mixed 3-level architecture of
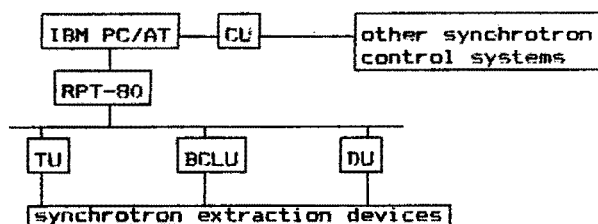
the computing systems was chosen (fig.1).

Fig. 1

A set of specialized microprocessor modules of KP580BM80 type, allowing to solve the following tasks was developed for operation at the low level:

- continuous measuring, tolerance parameters control and formation of the actions for controlling accelerator extraction devices;
- buffering, preliminary processing, information conversion and transmission to the computer of the higher level;
- synchronization of measurement and control processes with the synchrotron cycles.

The wish to minimize and get less expensive apparatours from the one hand and to achieve the sufficient universality of its functional possibilities from the other - was the main reason of the development of these modules and of not using the nucleus electronics apparatours.

The microcomputer RPT-80 (Hungary) used on the middle level with the processor of INTEL 8080 type runs system terminal functions in the separate control subsystems at its off line work and as a peripheral processor at the controlling through the higher level. In the first case it solves the user tasks providing a standard interface to all the modules of the low level and in the second one it solves the same problems as well as the other ones but under the control of the computer higher level.

At the higher level a personal computer IBM PC/AT is used, the main functions of which are the following:

- creation and maintenance of the parameter data base of the main operation modes of the beam extraction devices;
- realization o the local control algorithms with the feedback;

- statistic processing of measurement results at the normal system operation;
- information exchange with the other synchrotron control system.

For information exchange with the other synchrotron control systems, specially with rf systems, electromagnet supply system and others, the second serial port of IBM PC/AT is used, as well as a non-standard communication unit (CU).

1.1. Timer Unit (TU)

The timer unit is developed on the base of the microprocessors and is used for synchronization of all the extraction devices and equipment of physics-experimenters with the synchrotron acceleration cycles and carries out the following functions:

- control of the synchronization main pulse and its selection on the false signals background measurement and tolerance control of frequency; in case of mode violation of the main pulse forming timer unit automatically switches off the controlling of the extracted beam channels for elimination of the break-downs in the thyristor devices;
- program distribution of the synchronization main pulse in the devices of different extraction beam lines depending on the operator given sequence;
- time pulse delay formation in the given devices for running the phase control.

Main technical features
- pulse distribution channels number — 8;
- the range of the programmed pulse time delay is within $0.5 \mu s - 32 \mu s$;
- pulse distribution periodicity in the beam lines is arbitrary — up to 256 cycles.

1.2. Beam lines control units (BLCU)

Eight units of the beam control lines are based on the unified microprocessors for control all the parameters of the magnet extraction devices and have the following functions:

- measuring with the help of the different ADC — the current control pulse in the magnets and beam intensity signal from the

S04SRS05

scintillator pickup and realization of their tolerance control (the number of sampling points at analog signals measurements is up to 256, sampling step is 55 μs, measurement accuracy is 12 digits);

- the phase control of the amplitude and shape of the control current in the magnets by means of the 6 control time intervals for each forming current pulse;

- measuring and tolerance control of time intervals between synchronization and controlling pulses (up to 16 time intervals);

- information exchange with the higher level computer.

1.3. Diagnostics unit of the relay signals (DU)

This unit realizes registration, control, diagnostics of the state signals of the "switch on - switch off" type (number of channels -64, block time response to the state change - not more than 100 μs).

1.4. Software

The low level microprocessor units software is based on the program-monitor, realizing the main cycle of unit operation and organizing communications with the subprograms as well as on the asynchronous lines, drivers including subroutines of the data exchange, control words receive determining equipment operation mode and status-words transmission programs of the corresponding unit.

Middle level software [3] realized in RPT-80 consists of command monitor, global control table and command table, input-output dispatcher of the driver external devices, manager of the asynchronous communication lines and the interrupt handlers.

Command monitor realizes the direct interaction with the operator through menu, which gives the list of all available tasks and the ways of access to them. Monitor also supports the global control table and the command table.

The information of the external device operation mode, the interrupt bytes and also

the pointer to system area, which is given to each external device, are kept in the control table. Command table contains the addresses of the functional tasks.

The input-output dispatcher is created for a common access to different external devices.

Manager of asynchronous communication lines gives an alternative command input source which realizes the information exchange between the computers.

The described software is written in the assembly language in the CP/M OS environment and occupies 2 kB of ROM.

Conclusions

The created system realizes the following possibilities:

- continuous control of the extraction devices state;

- measuring and display of the current values of all the measured parameters in the digital and graphical form;

- monitoring the extraction beam quality and fault diagnostics;

- manual and automized control of the extraction devices through the computer while tuning the extraction modes and stabilization of their parameters.

More than one year operation of the system proved the reliability of its work and the convenient maintenance.

REFERENCES

1.   A.R.Toumanian,   Kh.A.Simonian, N.A.Zapolski. The way of getting the secondary beam in the high energy cyclic accelerators. (in Russian).

2. A.G.Agababian, and others. - The structure and organization of automized control subsystems by the beam extraction from the fast cycling synchrotron. Proc. of the II Sov.Union conf. on charged particles accel.V.1,p.125,Dubna,1989. (in Russian).

3. A.G.Agababian and others. Software subsystems control organization of the Yerevan synchrotron. Preprint YerPhy 1284(80)-90, Yerevan 90. (in Russian).

Instrumentation & Control System For PLS-IM-T 60 MeV LINAC

D.K.Liu, K.R.Yei, H.J.Cheng, L.P.Yang, C.Z.Chuang
W.Yue, P.Lu, Y.H.Su, Q.Lin, H.Huang
Institute of High Energy Physics
P.O.Box 918 Beijing

Abstract

The PLS*IMT* is a 60 MeV LINAC as a preinjector for 2 GeV LINAC of PLS project. The instrumentation and control system have been designed under the institutional collaboration between the IHEP (Beijing, China) and POSTECH (Pohang, Korea). So far, the I&C system are being set up nowadays at the POSTECH of Pohang. This paper describes its major characteristics and present status.

I. INTRODUCTION

The Concept Design Research(CDR) of PLS 60 MeV LINAC has been completed in 1989. The construction of PLS 60 MeV started in July, 1991. The accelerator column and electron gun have been installed earlier. The gun pulser has been tested with 3.5A 2ns pulse width with success and Modulator, microwave system and I&C system will be set up soon. The commissioning of whole system would be completed around the end of this year or next spring.

The I&C system of PLS 60MeV is a compact and complete hierarchical distributed control system. Therefore it is small system and it includes all of the essential control structure and various beam monitor, high speed electronics modules etc. for LINAC operation.

II. SYSTEM STRUCTURE

In a centralized control system, computer failure will cause a failure that will shut down the entire system. However, a distributed system is more costeffective and becomes easily modified.

According to the requirements of physics and our previous experience, and considering the entire budget, schedule of I&C of PLS 60MeV, we compared various structure of control system [1], and adopted the Intel BITBUS architecture. The major reasons are as follows:

* BITBUS distributed control system is a commercial product

* High performance microprocessor could be useful for local station.

* Powerful software support such as RMX286 and RMX51 are an excellent developing environment. The function that have to be explicitly coded can be greatly reduced by making system calls. A BITBUS drive can be run under the RMX286 which allows messages passing across the SBX interface on down the BITBUS network.

* More second source: We should consider the situation that developing this system is in China, and commissioning and maintenance is in Korea. So we must get these products easily from the market of both country.
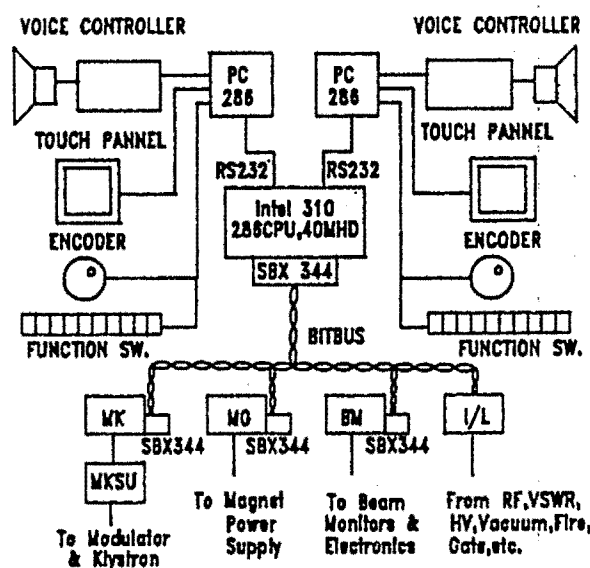
Fig.1 I&C archtecture for Preinjector of PLS

According to the considerations above, the system architecture is illustrated in Fig.1.

There are four stations linked with BITBUS network, each station has its own resource and tasks respectively. Those local stations are Modulator-Klystron Station(MK), Magnet power station(MG), Beam diagnostics station (BM) and Interlock station(IL).

In general, entire task are hierarchically managed. Each local station completes data acquisition and data control during the 5ms period. the details of MK local station

3rd Int. Conf. Accel. Large Exp. Phys. Control Syst.   ICALEPCS1991, Tsukuba, Japan (partial, pages 194 of 611)   JACoW Publishing

ISBN: 978-3-95450-254-7   ISSN: 2226-0358   doi:10.18429/JACoW-ICALEPCS1991-S04SRS06

will be discussed as example. Intel 310 (CPU 286, HD 40M) can be used for task management, data sort, data processing and BITBUS communication control. Two sets of industrial level console computer (CPU 286, HD 40M, RAM 2M) which can be used for humanmachine interaction.
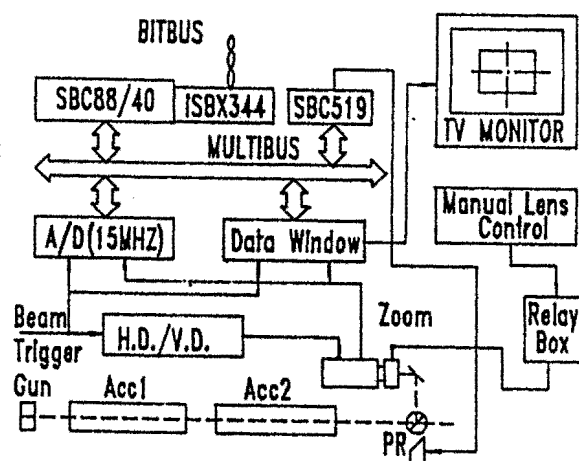

Fig.2 Beam Station Schematic

III. HARDWARE SYSTEM

A. Local station

Each local station has own iSBC 88/40 microprocessor and various I/O interface boards which are linked together on the Intel's MULTIBUS.

1. Modulator and Klystron Support Unit (MKSU) [2] is a powerful interface between MK station and Klystron & Modulator. It was developed by SLAC in 1985. and contains various interface circuit linked with intelligent PIOP CAMAC module. In order to keep this powerful function in our system, a dedicated bus adapted from MULTIBUS to MKSU bus have been developed successfully.

2. MG station

The PLS 60 MeV has 29 sets of power supplies to be monitored and controlled for serving the solenoid coil, steering coil, analyzing magnet and quadruple magnet. A digital remote control model using data modulator and demodulator with Manchester code has been adopted. By the way, a remote D/A controller could be mounted at the magnet power supply as close as possible.

3. BM station is designed for measuring various beam characteristics such as beam profile, beam energy spread and beam emittance. This is a multifunction image processing system, illustrated in Fig.2.[3]

It consists of a profile monitor, a video signal synchronizer, a high speed A/D converter with 15 MHz inserted into the Intel's MULTIBUS. The video signal of beam spot from the camera was transmitted to the TV monitor of control room. It is easy to correct the target haircrossing line using the movable data window by the computer control. The 4000 points signal could be collected in less than 4ms. After data processing, a beam profile distributed picture and three dimensional distribution will be shown to operator immediately.

The major function is follows:

* Measuring resolution better than 0.2mm

* Data window Size: 50x80(256x25) would be possible)

* Sample rate: Max. 15MHz

* Multipurpose: profile, energy spread, emittance measurement.

* High anti-interference:

When the beam intensity is so weak that the beam profile can not be observed on TV monitor, therefore it may be clearly seen on the graphic display after eliminating background noise from the image data taken repeatedly.

In order to record the 2ns beam intensity data which is important data for accelerator operator ,high speed sample hold circuits are being developed and we intend to use it instead of 7104 oscilloscope.

B. Timing system

PLS 60MeV LINAC timing system is a small system with three triggers to electron gun, travel wave tube, and modulator. In general, we refer to the LINAC timing system of KEK because the same kind system had been running for 5 years in the Beijing Election Positron Collider(BEPC) without trouble.

C. Beam monitors

According the physical requirements and our running experience in the BEPC, three categories of monitor are adopted.

The short pulse current monitor consists of a ceramic solid resistor in the shape of a disk, magnese-zinc ferrite aluminum case with BNC connector. Its features are:

* Measuring min. limit: 0.2 ma without amplifier

* monitor sensitive: 3mv/ma

* Frequency response: >1.5 GHz

The fluorescent target typed AF955 has been mounted in the profile monitor and can be movable by console computer. The beam loss monitor is not necessary for the short distance of 60 MeV LINAC, but it could be a prototype as reference for 2 GeV LINAC beam loss system.

IV. SOFTWARE SYSTEM

A. System software

The main control software on the Intel's 310 is the real time multitasking control software which is based on the BIT-BUS network. According to the LINAC physical requirements, it can carry out the control to each local station and make data processing. It owns its multitask scheduler. When the scheduler receives the command from the console, the related application tasks will be activated at any time. The control system uses fully operating functions, and the tasks will be put into operation in order of their priority level. The real-time data base is built in; it always holds refreshed data (over 300 signals) of whole control system. The data adjustment and command sending task is running forever after the control software is set up. It acquires the datum from each local station via BITBUS and updates the DB continuously in rate of 2-3 times/second.

B. Application software

In normal times, the control software in the local station is continuously acquiring the datum and monitoring from/to the accelerator's equipments. The major application software include as follows:

* Modulator/Klystron package [4] such as control and monitoring to modulator, drive power control, waveform digitalized control

* Magnet power control and monitoring

* Beam FWHM calculation and emittance processing etc.

The Human-Machine interactive software have been designed for those physicist and specialist who are not familiar with system software. It easy to operate and configure various control system. Please refer to "Human-Machine interface software Package" in this conference proceeding.

V. CONCLUSION

During the configuration of I&C system of 60 MeV LINAC, some technology, experiments and equipments such as beam monitors, MKSU, and timing are transmitted from KEK and SLAC. We believe that international collaboration has speeded up the progress of PLS 60 MeV LINAC.

Instrumentation and control system of PLS 60 MeV is designed for PLS's preinjector. So far, its commissioning with the whole machine will be in November, 1991.

It is a compact and complete control system for PLS 60MeV LINAC.

VI. ACKNOWLEDGMENT

VII. REFERENCES

[1] K.Nakahara, I.Abe, R.P.Bissonnette, A.Enomoto, Y. Otake, T.Urano and J.Tanaka, "Control System for the Photo Factory 2.5GeV Electron LINAC", Nucl. Instr. and Meth. A251 (1986) 327.

[2] R.K.Jobe, M.J.Browne and K.P.Slattery, "Hardware Upgrade for Klystrons in the SLC", SLAC-PUB-3666 poster paper presented at the 1985 Particle Accelerator Conference

[3] D.K.Liu, K.Nakahara, I.Abe, T.Urano, A.Enomoto and J.Tanaka. "Image Processing System for Electron LINAC Beam Diagnosis", Contributed to the 1986 Linear Accelerator Conference, 2-6 June 1986, SLAC, U.S.A.

[4] R.K.Jobe, K.Thompson, and N.Phinney, "Klystron control software in the SLC", SLAC-PUB-3667

Multi-Microprocessor Control of the Main Ring Magnet Power Supply of the 12 GeV KEK Proton Synchrotron

T. Sueno, K. Mikawa, M. Toda, T.Toyama and H. Sato
National Laboratory for High Energy Physics

and S.Matsumoto,
Physics Division, Dokyo University School of Medicine

Abstract

A general description of the computer control system of the KEK 12 GeV PS main ring magnet power supply is given, including its peripheral devices. The system consists of the main HIDIC-V90/25 CPU and of the input and output controllers HISEC-04M. The main CPU, supervised by UNIX, provides the man-machine interfacing and implements the repetitive control algorithm to correct for any magnet current deviation from reference. Two sub-CPU's are linked by a LAN and supported by a real time multi-task monitor. The output process controller distributes the control patterns to 16-bit DAC's, at 1.67 ms clock period in synchronism with the 3-phase ac line systems. The input controller logs the magnet current and voltage, via 16-bit ADC's at the same clock rate.

1.INTRODUCTION

The main ring magnet power supply consists of 10 twelve-pulse thyristor rectifiers with dc filters, of 2 reactive power compensators [1] with tuned ac harmonic filters [2] and of an analog and digital hybrid control system [3]. A schematic diagram of the power supply is given in Fig. 1. Fig.2 shows the principle layout of the hybrid control system. Eight rectifiers feed the bending magnets and the other two excite the horizontally and vertically focusing quadrupole magnets. Fine adjustment of the current at injection and of the ratio between currents of the bending and the quadupole magnets is required to tune the acceleration. The current of the quadrupole magnets must be tracked separately from the current of the bending magnets for precise Q-tuning and optimum beam acceleration.
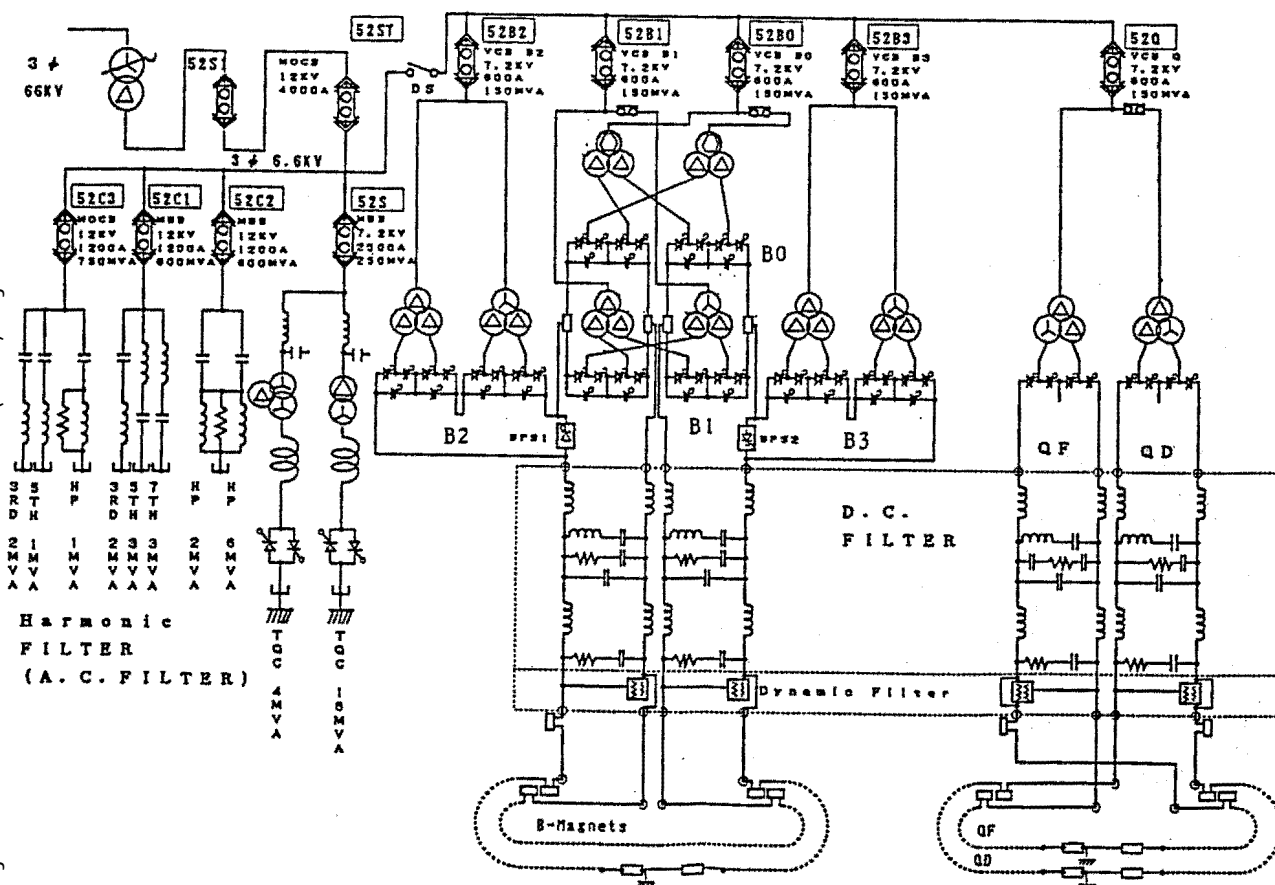

Fig.1 Schematic Diagram of the KEK 12 GeV PS Main Ring Power Supply.

The desired magnet excitation currents are obtained by controlling the output voltage of the thyristor power converters through the SCR gate firing pulse. The rectifier voltage reference patterns are implemented by the common action of two negative feedback loops, i.e. a low-gain automatic voltage regulator (AVR) and a high gain automatic current regulator (ACR). These patterns are elaborated by the control computer and fed to the regulation through a DAC, synchronized at 600 Hz on the zero crossing of the two 3-phase ac line systems. While providing the voltage reference patterns the computer implements a repetitive control algorithm on the base of measured deviations of the magnet current from reference . The digital system is in charge of the fast feedforward pattern control and of the repetitive control via the ACR.
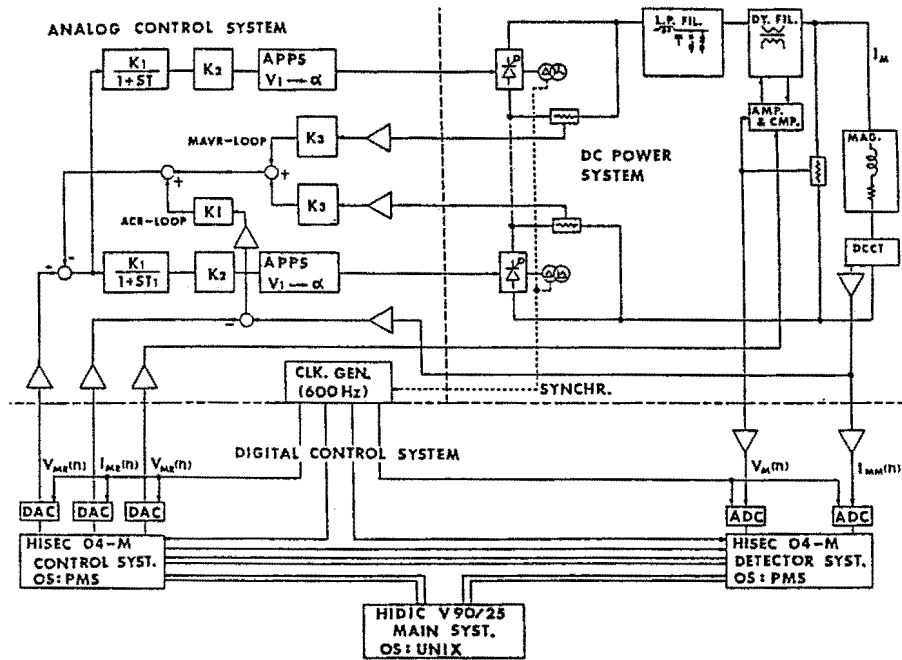

Fig.2 Schematic Diagram of the Hybrid Control System.

2. MULTI-MICROPROCESSOR CONTROL

2.1 General Layout

The multi-microprocessor control system has been introduced in 1985 [4]. Initially the main CPU was a V90/5 (8MHz without cash-memory); this was then upgraded to a V90/25 (16MHz with cash-memory) in order to improve the speed of the main system and of the communication loop. The main parts of the digital system are based on a 16-bit-microprocessor and on LSI components, connected to an industrial standard bus and to standard peripherals, supported by an universal operating system. Consequently a high level language and powerful utilities facilitate development and maintenance of flexible software for the pattern control system which consists of the main CPU HIDIC-V90/25 and of the input and output controllers HISEC-04M. The three distributed systems have no hierarchical software but are rather independent even at assembler level because of the difference between the CPU families and in particular of different addressing for memory access. The main components are LSI of the MC-68020 CPU family. The direct digital control system, as main part of the controller, consists of I-8086 and home-made LSI modules. Fig.3 shows a layout of the multi computer system.

2.2 HIDIC-V90/25 system

The system, equipped with memory management unit and 16MB DRAM on the internal bus, has a floating processor. Two local area network (LAN 1 and 2) loops provide the interconnection between main system and standard peripherals, i.e. 5-inch 80 MB hard disc and 8-inch 1 MB floppy disc drive, two CRT terminal stations , a typewriter and a printer. These resources are supervised by the UNIX compatible main OS.   One of the network-loops, LAN 1,
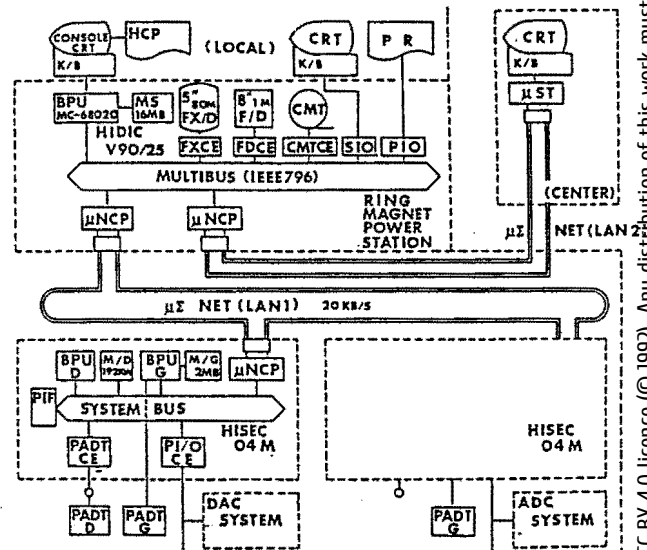
Fig.3 Layout of the Control Computer Netwok System.

exclusively   devoted to   input or   output network communication, transfers patterns or logged data between the main and the input or output controllers. The fact that this communication is rather slow, due to time sharing operation on the same system bus, limits the speed of response for fine adjustments.

The local terminal is supported by a serial I/O full duplex link. The remote terminal, located in the Center Control Room (CCR) of the 12 GeV PS, is linked by an optical cable to cope with the 350 m distance between CCR and power station, where the main system is installed.

The application programs are written in the system language C and FORTRAN 77 [5]. The parameter files of

the programs displayed on a CRT terminal can be communicated to the central control system. Powerful UNIX utilities are used not only for program development but also for maintenance of the application programs, controlling and monitoring the whole system by file management, screen editor and shell command.

Pattern generation can be done while the power supply is running by using the main CPU and storing the new pattern in its memory. Therefore the operating pattern can be changed without interrupting power supply operation. Fine adjustment of the injection current and of the tracking ratio between bending magnet and quadrupole current is done from either terminal in the CCR or the local power supply control room. The main tasks are control operation, e.g. start-stop and status monitoring, calculating the correction patterns of the repetitive control and fine adjustment of pattern data. As far as additional supporting tasks concerns, the system works on pattern generation, processing of pattern and operational data, control program development and background processing. The main operator commands are shown in Table 1.

Table 1.

**** MR-PS OPERATOR COMMANDS****

f1 [pattern No.]	: run
f2	: stop
f3 pattern No.	: pattern exchange (with repetitive control)
f4	: IQ tracking adjust
f5	: B inj. adjust
f7	: pattern generation
f8	: PS status display
f9	: pattern No. display
f10 [pattern No.]	: pattern save
f11 pattern No.	: pattern remove
f33 pattern No.	: pattern exchange (without repetitive control)
menu	: command menu display
menu2	: command menu next page display
pc20	: repetitive control start
cpcstop	: periodic control stop

Footnote: [default pattern number] can be neglected

Generated pattern files are used for fine adjustment of injection level of the bending magnet current (without tune shift) by additional smoothing corrections calculated in the same pattern generation algorithm. Tracking offset calculations are done similarly to injection current
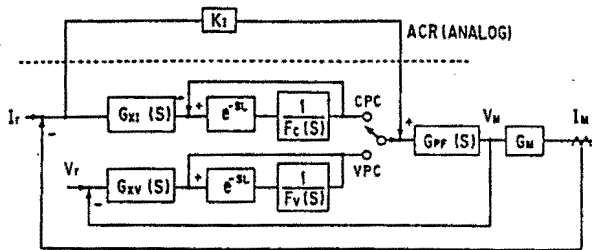
GX(n) : Transfer function for compensation.
F(s)   : Transversal finite impulse response of low pass filter.
L=mT : Dead time of m times period T, m : integer.

Fig.4 Block Diagram of the Repetitive Control.

corrections. Concerning the B2 and B3 rectifiers, the reference voltage pattern is subdivided and distributed to each of the 12 pulse thyristor converter groups in order to obtain the desired magnet voltage with minimum reactive power generation [6]. In both fine adjustment cases a step variation is smoothed out by applying optimum polynomials in a fixed interval. The main and the controller systems are linked in a LAN with an effective transfer rate of 20 kB/s. Typical response time, on fine adjustment of tracking or of injection current for beam tuning, was 20 to 50 s for the V90/5, even after the control programs have been optimized by fixed point calculation, but becomes less than 10 s in case of the V90/25. Table 2 shows as an example display of injection current adjustment by function f5 (see Table 1).

Table 2.

MR-PS injection tuning (on line)
page - 1/1   proton/q05066

Ib	injection	: 198.88
Iqf	injection	: 116.73
Iqd	injection	: 116.29

Binj. [G] 1450.0

UPPER LIMIT > INITIAL > LOWER LIMIT
Injection [G]
1598.9 > 1449.0 > 1285.5

Repetitive magnet current control has been performed to suppress the deviations from a given current reference pattern and repetitive voltage control has been used to reduce the parasitic voltage ripple [7]. The principle is based on the control method applied to a repetitive reference input [8]. The excitation current in the respective magnets is to be controlled according to periodical patterns. The frequency response of the correcting transfer function has been confined in a lower frequency region, about 15 Hz or less, to track a periodic input and assure the stability of the power system. Fig. 4 shows an algorithm of the repetitive control of current and voltage. In Fig. 5 the convergence of the repetitive control, from the initial pattern to the corrected one, for an ACR deviation of the bending magnet current pattern is shown over eight correction cycles. (Timing pulses, P1,P2,P3 and P4 are injection start, acceleration start, flat top start and flat top end, respectively.)
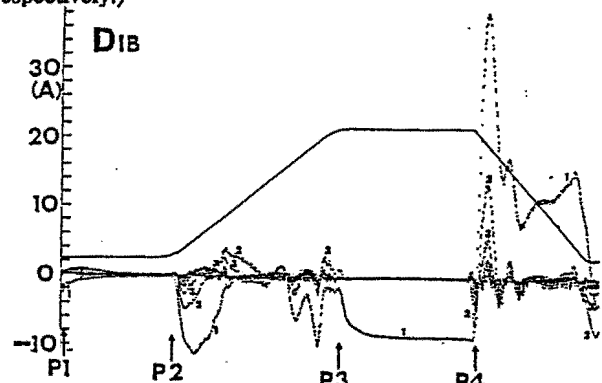

Fig.5 Converging of Current Deviation by Repetitive Control.

2.3 Controller System

The HISEC-04M/D and G controllers, belonging to different families i-8086 and HD-68000, are subdivided into two sub-system working on the same system-bus. The first one works as direct digital control and feeds operational patterns through 16-bit DAC's at 600 Hz clock. The other one acts as a data processing system and a support to the direct digital control; it logs the magnet current from the DCCT and the group voltage through a sets of 16-bit ADC's working at the same clock. It communicates through LAN 1 with the V90/25 and reads or writes data or messages on the memory of the HISEC-04M/D.

The output controller, supervised by the process monitoring system, executes application programs such as start-stop as well as process timing and sampling synchronization. The routine output processing function distributes 15 data patterns in memory through parralell I/O to the 16-bit DAC's.

Eleven sets of DAC's serve the bending magnet power supply: eight of them give the voltage reference patterns to the thyristor converter groups, two are for the ripple detectors of the dynamic filters and one gives the current reference for the analog ACR. Each focusing and defocusing magnet power supply has three sets of DAC's. Two serve as voltage and current reference to the analog loops and one is used for the dynamic filter. The system ouputs the pattern data to the seventeen sets of DAC at every 1.67 ms clock period and the data conversion is synchronized to the zero-crossing of the six phase ac power line. Control signals of by-pass thyristors and gate pulse suppress signals are distributed by the system.

The input controller is dual with respect to the output controller as far as hardware and system software concern, except the digital input and output. Its main task is to collect the data from the ADC's through paralell I/O and to accumulate and save them at every control clock.

Simultaneously the system reads data from six sets of 16-bit-ADC's through a Sample/Hold amplifier at the same clock as the DAC system. Three sets serve for the DCCT current signals and the others for the dc voltages applied to the B, Qf and Qd magnets. The clock is sychronized on ac voltage zero-cross pulses but has a constant delay of 100 microsec corresponding to about twice the conversion time. The DCCT current data serve in the repetitive control loop for calculation of corrections to the voltage references.

3. CONCLUSION

The hybrid control scheme of an analog and a digital system and the multi-microcomputer control system HIDIC-V90/25 and twin HISEC-04M have been implemented in the main ring magnet power supply of the 12 GeV PS. The HIDIC-V90/25 and HISEC-04M system perform fast feedforward pattern control at 600 Hz clock and slow but high gain feedback control of current pattern for steady deviations according to the repetitive control method. The analog system works as real time feedback control loop of the voltage and current reference pattern fed by the digital system.

The repetitive current control is not yet used routinely, despite its effectiveness, due to the cumulative effect of small errors produced by the intrinsic ripple of the present DCCT. It is used for initial pattern correction and

fine adjustment of the injection current and of the tracking ratio between bending and quadrupole magnet currents. It is easy to change the operating patterns without stopping the power supply. During the extension of the flat top duration [9], memory and hard disc capacity has been increased and the application software modified. When operation is performed with a long magnet current flat top, the control clock is synchronized at 300Hz to save memory space. At present the multi-microcomputer control system allows to perform stable operation of the PS and to achieve effective utilization of the slow extracted beam spill.

References:

[1] S.Matsumoto, H.Baba, H.Sato, T.Sueno and K.Mikawa, "Improved Control System of the Thyristor Flicker Suppressor for the KEK 12GeV PS," IEEE Transactions on Nuclear Science, NS30(1983)2932

[2] T.Shintomi, M.Masuda and S.Matsumoto,"Harmonic Current AC Filters at a Large Accelerator," Particle Acc. 8(1978)87

[3] S.Matsumoto, T.Sueno, K.Mikawa and N.Kumagai, "Control of the Main Ring Magnet Power Supply for the 12 GeV PS," Proceedings of the 6th Symp. on Acc. and Tech., 1987, Tokyo, Japan, pp.158.

[4] S.Matsumoto, T.Sueno, K.MIkawa, M.Toda and H.Baba,"New Multi-Microprocessor Control System of the Main Ring Magnet Power Supply for the KEK 12 GeV PS," Proceedings of the 1987 IEEE Particle Accelerator Conference, 1987, Washington D.C., pp.689.

[5] T.Sueno, M.Toda, S.Matsumoto and K.Mikawa, "Software Systems of the Main Ring Magnet Power Supply for the 12 GeV PS," Proceedings of the 6th Symp. on Acc. and Tech., 1987, Tokyo, Japan, pp.155.

[6] H.Sato, T.Sueno, T.Toyama, M.Mikawa, T.Toda, S.Matsumoto and M.Nakano, "Performance of the Main Ring Magnet Power Supply of the KEK 12 GeV Proton Synchrotron", Proc.of the 1991 IEEE Nuclear Science Symposium, 1991, Santa Fe, to be published.

[7] T.Sueno, S.Matsumoto, K.Mikawa, M.Toda, H.Sato, N.Kumagai and H.Baba,"Repetitive Voltage Control of the Main Ring Magnet Power Supply for the KEK 12 GeV PS," Proceedings of the 14th Int. Conf. on High Energy Accelerators, 1989, Tsukuba, Japan, Particle Accelerator. 29(1990)133.

[8] T.Inoue, M. Nakano, T.Kubo, S.Matsumoto and H.Baba,"High Accuracy Control of a Proton Synchrotron Magnet Power Supply," Contl.Sci. & Tech. for Prog. of Soc. (IFAC 82) Vol.6, 3137, Pergamon Press(1982)

[9] H.Sato, T.Sueno, T.Toyama, M.Mikawa, T.Toda and S.Matsumoto, "Upgrade of the Main Ring Magnet Power Supply for the KEK 12 GeV Proton Synchrotron", Proc.of the 1991 IEEE Particle Accelerator Conference, 1991, San Francisco, to be published.

S04SRS07

VME COMPUTER MONITORING SYSTEM OF KEK-PS FAST PULSED MAGNET CURRENTS AND BEAM INTENSITIES

T.KAWAKUBO, A.AKIYAMA, T.ISHIDA*, E.KADOKURA

National Laboratory for High Energy Physics, 1-1, Oho, Tsukuba-shi, Ibaraki-ken, 305, Japan

*Mitsubishi Electric Company, 1-7-4, Iwamoto-cho, Chiyoda-ku, Tokyo-to, 101, Japan

Abstract

For beam transfer from the KEK-PS Linac to the Booster synchrotron ring and from the Booster to the Main ring, many pulse magnets have been installed. It is very important for the machine operation to monitor the firing time, rising time and peak value of the pulsed magnet currents. It is also very important for magnet tuning to obtain good injection efficiency of the Booster and the Main ring, and to observe the last circulating bunched beam in the Booster as well as the first circulating in the Main. These magnet currents and beam intensity signals are digitized by a digital oscilloscope with signal multiplexers, and then shown on a graphic display screen of the console via a VME computer.

1. INTRODUCTION

There are many pulsed magnets and beam monitors which concern beam injection and extraction of the KEK-PS-Booster as well as beam injection of the Main ring. In order to tune the machines and to search trouble points, it is very important to display these signals using proper trigger timing. Because we must select a proper signal and trigger among many connectors and choose a proper time scale, voltage range and trigger level, only a few trained crew members had been able to observe the expected signals within a short time. By using signal multiplexers, a digital oscilloscope with GPIB and a VME computer system, however, we can now observe the expected signals without any great effort using a touch panel of a console desk in the PS-control room.

---

*When you observe rapid changing figures as a kicker current and a fast beam intensity in the control room, the figure deterioration through a long co-axial cable becomes problem. We have re-shaped the deteriorated figure to the original by the "equalizer" made by Dr.S.Ninomiya. We would like to acknowledge him for his offering of his instrument.

2. PURPOSE OF THIS SYSTEM AND REQUIRED SIGNALS

A. Observing pulsed magnet current

In order to observe the operating conditions of the pulsed magnets, the following magnet currents should be observed with proper time scale:

- For Booster Injection:
  - four Bump magnets in series

- For Booster Extraction:
  - Bump(#1,#2)
  - Septum(#1,#2)
  - Kicker(#1~#4)

- For Main Injection:
  - Septum(#1,#2)
  - Kicker(#1~#5)

B. Checking the magnets' firing timing

For beam transport from the Booster to the Main ring, just after firing the Booster extraction septums and carrying out time-matching of an RF bucket of the Booster ring with one of the Main ring, Booster extraction bumps are fired; after about 20μsec, four kickers are fired at the same time. After a transfer time from the Booster to the Main, firing of five Main injection kickers follows. In order to check these timing, it is convenient to display the concerning magnet currents and a bunched beam intensity with a "mountain view"*.

- For Booster extraction:

  - a mountain view of currents of four kicker magnets and a pulsed beam measured by a wall current monitor (see Fig.1)

- For Main injection:

  - a mountain view of currents of five kicker magnets and a pulsed beam measured by a wall current monitor

- For all pulse magnet fire timing:

  - a mountain view of currents of septums, bumps and a kicker (see Fig.2)

Figure 2. A mountain view of all kinds of pulsed magnets in PS-BT (from top figure to bottom)
Booster Extraction Bump (#2, #1)
Main ring Injection Septum (#2, #1)
Booster Extraction Septum (#2, #1)
Main ring Injection Kicker (#1)
Booster Extraction Kicker (#1)
Booster Extraction Bump (#2, #1)

Figure 1. (from top figure to bottom)

A mountain view of currents of a pulsed beam measured by a wall current monitor, currents of four kicker magnet and averaged those four kicker currents

C. Tuning machine

After tuning positions and emittance figures of a beam at the transport line, final tuning should be carried by observing the injection efficiency of the Booster and the Main. By dividing the Booster circulating beam particle number at injection by the Linac beam particle number, which is calculated by integrating the Linac beam current with time duration, the injection efficiency of the Booster is obtained. And by dividing the circulating particle number at the Main injection by that at Booster extraction, the injection efficiency of the Main is obtained. The fire timing of the kicker magnet should also be adjusted by observing the height of the Booster

bunched beam at extraction and of the Main at injection. The trigger used to observe these monitors can be selected among nine successive Booster extraction beams which inject to the Main ring in the Main injection porch:

- For Booster injection efficiency:

    - Linac beam intensity and Booster particle number measured by a slow intensity monitor (see Fig.3)

- For Main injection efficiency:

    - Booster particle number measured by a slow intensity monitor and the Main particle number

- For Booster kicker firing timing:

    - fast intensity monitor at Booster extraction

- For Main kicker firing timing:

    - fast intensity monitor at Main injection

Thu Nov 07 10:37:09 1991
CH1 100mV/div 10us/div

LINAC   8.98e+11ppp
BOOSTER 8.56e+11ppp
Injection
Efficiency **95%**

CH2 100mV/div 10us/div
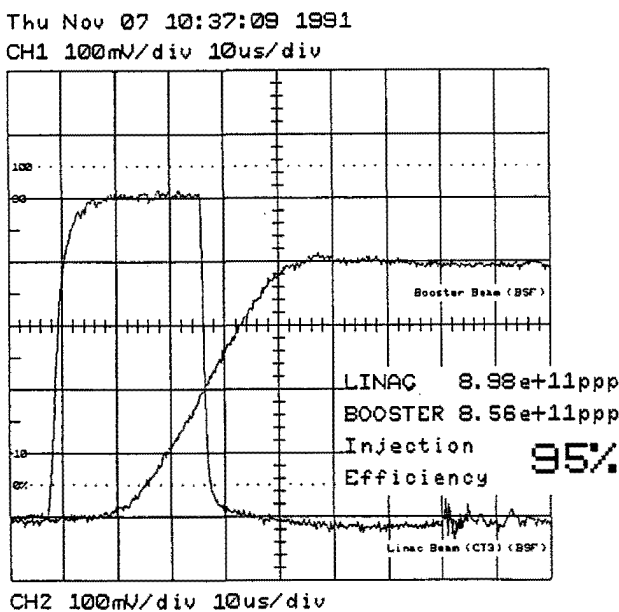
Figure 3. Linac beam intensity, Booster particle number at injection and Injection efficiency

D. Searching for trouble points of the bump and kicker systems

The thyratron used in a bump and kicker power supply has a lifetime when the turn-on timing becomes delayed. When one of the two Booster extraction bump magnets happens to show such a deterioration, the betatron amplitude arising from the bump firing increases. Therefore, two bump currents and ΔR monitor signal in the Booster are needed to check the problem.

Every kicker has a separate delayed trigger circuit, which has low reliability, and sometimes becomes out of order. When one of the kicker currents begins to be delayed, we can distinguish which causes the trouble (thyratron or delayed module) by changing the following trigger:

- For Booster extraction bump trouble (signal):

    - Booster extraction bump currents and ΔR monitor signal

- For kicker trouble (trigger):

    - origin of the trigger to fire all kickers

        * for Booster extraction kicker
        * for Main injection kicker

    - output of delayed trigger module after branching from the origin trigger

        * Booster kicker($\#1 \sim \#4$)
        * Main kicker($\#1 \sim \#5$)

3. BLOCK DIAGRAM OF THIS SYSTEM

In order to obtain the injection efficiency precisely, the Linac and Booster intensity, or the Booster and Main intensity, should be evaluated at the same time. Therefore, the Linac and Main intensity are connected to different inputs of an oscilloscope from that of the Booster intensity. The figures changing rapidly as a kicker current are connected to the input via the "equalizer" mentioned in the footnote. All of these input figures are displayed by using a proper trigger, as shown in Fig.4.

4. FUTURE IMPROVEMENT

We are using an oscilloscope with a sampling rate of 250MS/sec; the memory number is 1K words. This number is too small to display the synchrotron oscillation by taking the envelope of the height of the bunched beam train (because of "areasing" of digital oscilloscope).

We will purchase an oscilloscope with greater memory, so that we can display not only the synchrotron oscillation, but also a "mountain view" of a bunched beam train at the Main ring injection with an interval of a quarter of the synchrotron period.
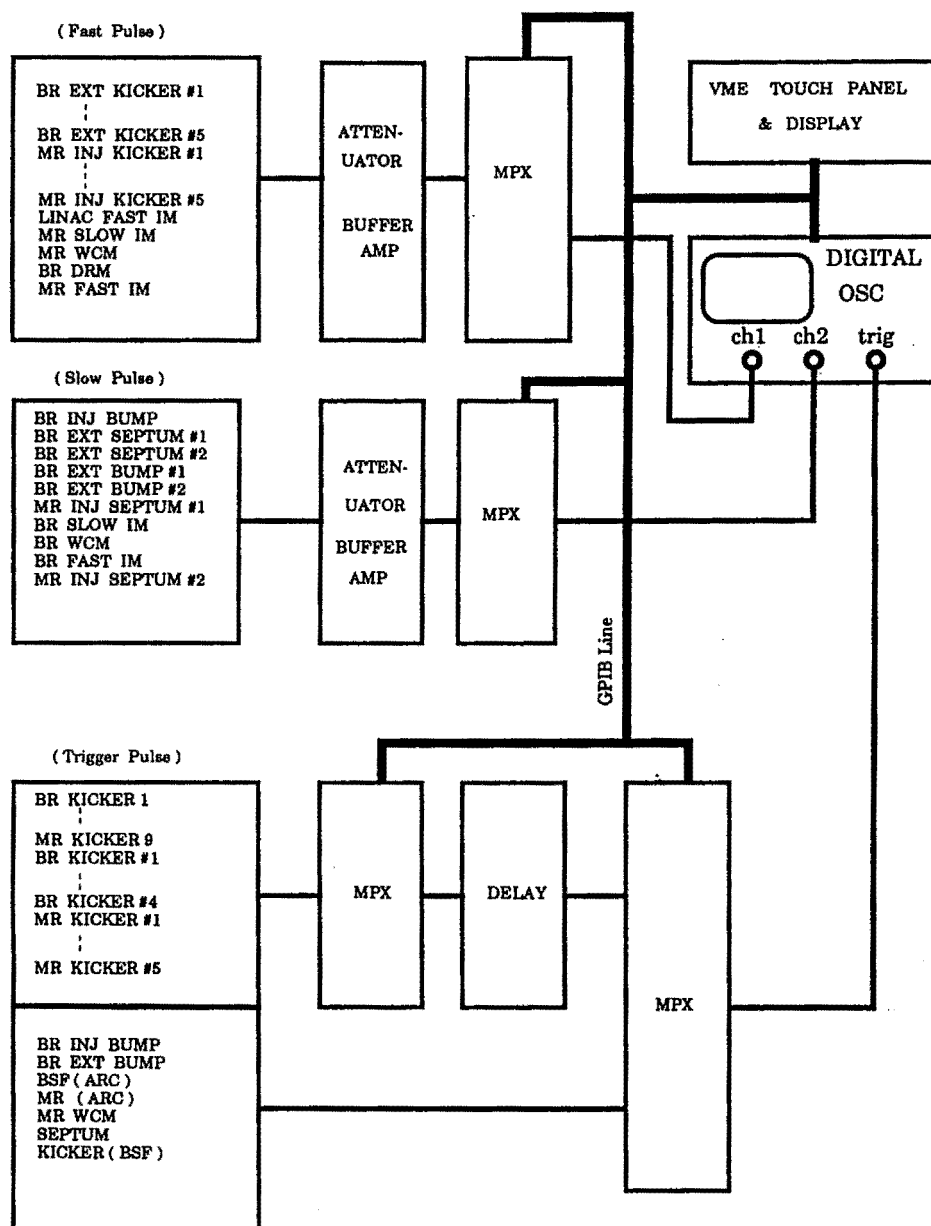

Figure 4. Block diagram for observing system of pulse currents
(BR:booster ring, MR:main ring, IM:intensity monitor, WCM:wall current monitor,
DRM:delta R monitor, BSF:booster facilities, MPX:multiplexer)

Magnet Power Supply and Beam Line Control

for a Secondary Beam Line K6

Y.Suzuki, M.Takasaki, M.Minakawa, H.Ishii,
Y.Kato, M.Ieiri, K.H.Tanaka, H.Noumi,
and Y.Yamanoi

KEK National Laboratory for High Energy Physics
1-1 Oho, Tsukuba, Ibaraki 305, Japan

Abstract

K6 is a secondary separated-beam line with momentum range up to 2.0 GeV/c in the north experimental hall at the KEK 12 GeV Proton Synchrotron (KEK-PS). On the construction, newly developed magnet power supplies (MPSs), in each of them a microprocessor is embedded, are introduced. The features of the MPS are as follows:
1, The MPS is connected to an upper-level beam line controller (BLC) by GPIB highway for exchanging simple messages.
2, All the operations of the MPS are supervised by the microprocessor, which has its individual parameters and fault messages. It reduces the load of the upper-level controller.
3, The MPS has functions to inspect itself and to report the result. It saves much time and labor of maintenance.

INTRODUCTION

On the KEK-PS site, there are two experimental halls for high energy physics experiments. The one is the East experimental hall (E-hall), that has been servicing since 1977. The other is the North experimental hall (N-hall) built in 1990, where the construction of the new beam line K6 construction is under going.

In N-hall, the beam lines were designed for high-intensity proton beams against high radiation field. The R&D for the beam line components has made during the last few years. The design of the magnets and vacuum system were reported in [1]. On the other hand, magnet power supply (MPS) and control system have also been developing [2].

On the construction of the K6 beam line, newly developed MPSs were introduced. Each MPS has a microprocessor, ADC(analog to digital converter), DACs (digital to analog converter), relay I/O(input and output), and GPIB(IEEE-488) interface. The digital processing unit i.e. magnet power supply controller (PSC) is incorporated into the MPS to have functions; ON, OFF, reset of interlocks circuit, polarity switch, current/ voltage control mode, current setting with appropriate speed, and checking the health of the MPS without help of upper-level beam line controller. These functions are invoked by a simple message, for example; current setting message; "A 1234.0" means to set output-current to 1234.0 ampere. This message is sent to MPS from BLC through a single coaxial cable; GPIB highway.

These design concepts were reported several years ago, [3], [4]. Though effective, those design concepts have been applied to few devices on the accelerator field up to present.

We introduced this design concept in to beam line control system, and developed MPS. Now we are saving cost, time, and trouble.

This paper report this MPS's PSC and beam line control for K6. The details on soft program and hardware will be reported elsewhere.[5]

POWER SUPPLY CONTROLLER (PSC)

Hardware

PSC consists of five boards (STD: IEEE-961):

(1) CPU board: Z-80, GPIB communication interface.

(2) DAC board: 16-bit DAC, for reference voltage.

(3) ADC board: 16-bit ADC, for monitoring DCCT (output current).

(4) ADC board: 12-bit ADC, 16-channel multiplexer.
This board is used for monitoring the following values:
    1, Reference voltage (16-bit DAC).
    2, MPS's DC output voltage
    3, MPS's output voltage for monitoring Wave Form.
    4, MPS's AC input current .
    5, Input voltage of firing module.
    6, Seven points on low-voltage power supplies.

(5) Relay I/O board: for control and monitor.
 The control points are:
    1, ON/OFF
    2, Reset of interlock logic.
    3, Polarity switch.
    4, Regulation mode (voltage or current).
    5, Remote or Local

Input points for monitoring status are:
    1, Control power ON/OFF.
    2, Remote/ Local.
    3, Main switch, ON/OFF.
    4, Polarity +/-.

5, Magnet #1 Ready/ Trouble.
6, Magnet #2 Ready/ Trouble.
7, Fault; DC over current.
8, Fault; DC current leak,
9, Fault; MPS over heat #1.
10, Fault; MPS over heat #2.
11, Fault; cooling fan
12, Fault; MPS's door open.
13, Fault; magnet #1; over temperature.
14, Fault; magnet #1; cooling water flow trouble.

Software

All the PSC's soft programs are written in assembler language. They mainly consist of :
(1), GPIB communication program.
(2), MPS operation program.
(3), Watching program.
(4), Fault or error message list for diagnosis of
MPS.

Messages [1] BLC--> PSC

The messages sent to PSC in MPS from BCL are the followings:

1),   "ST$"
requests PSC to send status message of MPS. This message includes the following:
ON/ OFF,
REMOTE/ LOCAL,
READY/ NOT READY,
+/ - ,   : polarity
CC/ CV,   : regulation mode
Reference value,
DAC value,
DCCT value,
Voltage value,
AC input current value.

2),   "ST1$" , or "ST3$".
These messages request PSC to send MPS's status as follows;
DCCT value, or absolute output current [A].
Voltage value, or absolute output voltage [V].
+/ - ,   : polarity
CC/ CV,   : regulation mode
Ac input current value [A].

3),   "ST2$"
requests to send
Absolute output current [A],
Absolute output voltage [V].

4)   "FL"
requests to send fault messages to help diagnosis of the MPS. Its message is text-format message.

5),   "AC"
requests to send AC power source input current [A].

6),   "OW"

requests to send wave form data of MPS's output voltage; 255 words binary data with EOI, its sampling period is about 25 millisecond. Figure 1 shows an example of the wave form.
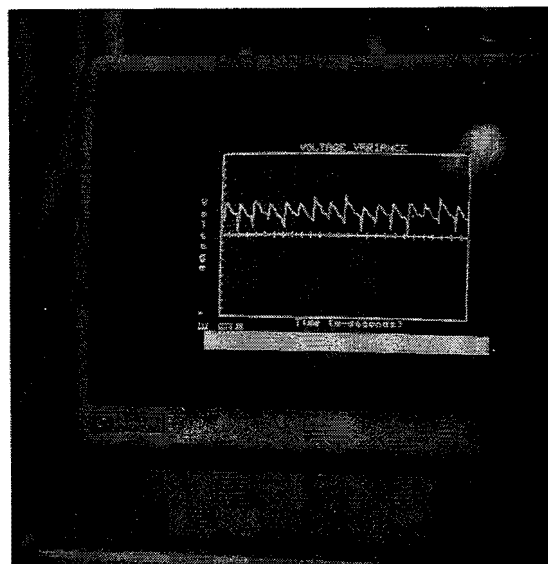
Figure 1: Wave form of MPS's output voltage

7),   "ID"
requests to send MPS's identifying message;
It includes MPS's rating, factory name fabricated the MPS, etc.

8),   "Y0"
requests to send DCCT output voltage for monitoring.

9),   "X0","X1","X2",, , "X15"
request to send the value of ADC(with 16 channel multiplexer) for monitoring.

10),   "A xxx.xx"
requests MPS to set output current xxx.xx [A] smoothly.

11),   "V xxx.xx"
requests MPS to set output voltage xxx.xx [V] smoothly.

12),   "D x"   : (-10000<= x <= +10000)
requests PSC to set xxxx data to DAC smoothly.

13),   "T1", "T2", "T3"
are setting speeds of output current or voltage.
T1: fast.
T2: middle.
T3: slow.

14),   "L x"
sets the limit value of output current for watching.

15),   "CC"
sets MPSs to constant current regulation mode.

16),   "CV"

sets MPS to constant voltage regulation mode.

17),   "CK".

It invokes check program to check MPS. When MPS is OFF state, the check program checks Interlock's fault signal, low voltage power supplies, DAC output voltage, and function of main switch and polarity switch. When MPS is running, it initializes check list and fault flag, then check program starts.

18),   DC, SDC   : GPIB command.

initialize PSC and MPS. Main SW is off, and all data is clear.

Message [2] PSC--> BLC

The messages sent to BLC form PSC have described already in the above section without SRQ.
The SRQ is a signal of PSC to request BLC. One status byte is sent to BLC.
The bit assignment is listed below.

Bit 0,   0:   MPS is OFF state.
         1:    MPS is ON.
Bit 1,   1:   Local control mode.
Bit 2,   1:   Fault on interlock.
Bit 3,   1:   Fault on ON, OFF, polarity SW procedure.
Bit 4,   1:   Fault on Values; output current, low
               voltages, or DAC: reference voltage.
Bit 5,   1:   Message error.
Bit 6,   1:   SRQ
Bit 7,   1:   This bit is set after the check program runs,
and indicates that the fault-list is available for read-out.

MPS operation

All the operation of MPS is done by PSC in MPS. When current setting message is received, the MPS's operation is as follows.

Step 1,   Receive "A xxx.xx".
Step 2,   Invoked MPS operation program.
Step 3,   Reset MPS interlocks.
Step 4,   Check Interlocks fault signal.
Step 5,   Check MPS's low voltage power supplies,
            MPS output current, and DAC output
            voltage for reference.
Step 6,   Set MPS to current or voltage regulation
            mode.
Step 7,   Check polarity, and turn polarity switch.
Step 8,   Main power switch ON, and check.
Step 9,   Current setting and check trouble loop.
Step 10,   Current setting end, and SRQ.
Step 11,   Set limit values for watching program.
Step 12,   Watching: check status (ON/OFF, interlock
signal, remote or local), output current, reference DAC, and low voltage power supplies.

On this state, when MPS receives "A-xxx.xx" message, the following steps are done.

Step 13,   Current setting starts to 0 [A].
Step 14,   Main power switch OFF.
Step 15,   Jump to the above step 2.
Final state is negative (-) xxx.xx [A].

On the step 15, when MPS receives "V+ xx.x" message, the following steps are done.
Step 16,   Current setting starts -xxx.xx[A] to 0 [A].
Step 17,   Main power switch OFF.
Step 18,   Jump to above step 2.
Final state is positive(+) xx.x [V] on voltage

On the OFF state, when MPS receives "CK" message, the following steps are done.
Step 19,   Reset interlocks, check fault signal.
Step 20,   Check output voltages of low-voltage power
                supplies, DAC, and DCCT(output current).
Step 21,   Turn polarity switch, check its status.
Step 22,   Turn main switch ON, check its status.
Step 23   Check output voltages of low-voltage power
                supplies, DAC, and DCCT.
Step 24,   Turn main switch OFF, check its status.
Step 25,   Send SPQ, bit 7 added in order to indicate
                check-end.

BEAM LINE CONTROL

MPSs and control system for K6

The control system for K6 using new MPSs is shown in figure 2. All the MPSs are connected through GPIB highway to the BLC by a coaxial cable. A terminal display connected to the BLC is offered to a user group to operate the beam line. The BLC computer is dedicated for the beam line control and the MPS's maintenance work.
On construction stage of the beam line, this new MPS's function is effective for checking MPSs. All the function of MPS is performed through the microprocessor embedded in the MPS. Then commands or messages to the MPS is simple, those MPSs can be operated easily by direct command of personal computer with interpretive language (BASIC) too. The diagnostic information of the MPSs are able to get without checking program on the BLC, for its checking or test program is stored in each of MPSs.
On this configuration BLC programing becomes simple, and no checking program is running constantly on BLC. As the result, the GPIB communication line between MPSs and BLC becomes quiet, there is no problem with speed.

The equipments for K6 beam line control system are:
BLC: Personal computer with BASIC language   HP-300
TRM: Terminal for user of K6 beam line.
EX: GPIB bus extender for long distance up to 1250m,
        60k bytes/s. HP 37204A
MPSs: listed bellow.

Name	Address	KW	A	V
D1	1	260	2000	130
Q1	2	65	1300	50
Q2	3	105	1600	65
Q3	4	200	2000	100

Q4	5	160	2000	80
CM1	6	85	1300	65
CM2	7	85	1300	65
Sext	8	50	1000	50
Q5	9	65	1300	50
Q6	10	85	1300	65
Q7	11	105	1600	65
Q8	12	120	2000	60
D2	13	500	2500	200
Q9	14	120	2000	60
Q10	15	160	2000	80

Figure 2. Configuration of K6 beam line control

The soft program for K6 beam line is written by BASIC language, which does not include the operation procedures or the fault messages of MPSs, therefore it becomes simple.

The operation procedure or the specifications or the fault messages of MPS are MPS's own. It is better that they are not separated from MPS's body, because they form MPS's character. If they were separated, in case of worst case of MPS exchanging, it needs more work, e.g. exchanging of procedure program and fault messages for individual MPS in BLC.

CONCLUSION

We have developed new MPSs, and introduced the MPSs in K6 beam line. Both the specifications and the source program of the PSC were offered at free to factory or workshop for MPS fabrication. By using this type of MPS, we rationalized the work on control wiring, check and maintenance work of MPSs, and BLC programing.

No longer on BLC programing the programer need to be concerned with the specifications or the fault messages or codes of MPSs.

REFERENCES

[1]   K.H. Tanaka, etc all, The Beam-Handling Magnet System of the KEK-PS New Experimental Hall: MT-12, June 23-28, 1991, Leningrad, U.S.S.R.
[2]   Y. Suzuki, M. Takasaki, Development of a Computer-controlled Magnet Power Supply for KEK PS Beam Lines: NIM, A293, pp. 253-257, 1990.
[3]   M. Crowley-Milling, Control Problems in Very Large Accelerators: IEEE NS-32, No. 5, pp. 1874-1876, Oct 1985.
[4]   J. H. Humphrey, The Isabelle Control System-design Concepts: Distributed Computer control systems, Proceedings of the IFCA Workshop, Oct 1979.
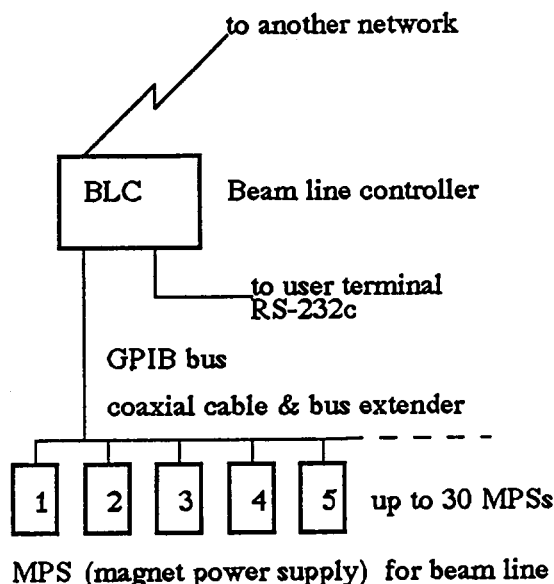[5]   Y.Suzuki, M. Takasaki, PSC hardware and soft program, KEK internal paper.

SPECIFIC BEAM DELIVERY SYSTEM OF MEDICAL ACCELERATOR HIMAC

S.Minohara, T.Kohno, M.Sudou, M.Endo
T.Kanai, F.Soga and K.Kawachi

Division of Accelerator Research
National Institute of Radiological Sciences
Chiba , JAPAN

Abstract

A specific beam delivery system for radiation therapy in HIMAC is being designed. This report describes an outline of the beam delivery control system and its operation.

I. INTRODUCTION

HIMAC is a heavy ion accelerator facility designed for radiation therapy[1]. Beam delivery system of HIMAC is very specific and different from the ordinary facilities for experiments of general physics. The treatment control system for irradiation of patients is closely linked with operation of accelerator and beam transport. We report an overall idea of HIMAC beam delivery system and its operation for radiotherapy.

II. CLINICAL REQUIREMENTS

The clinical requirements for radiation therapy are described as follows.

1) At the end of irradiation, three dimensional dose distribution at the tumor volume in the patient must be achieved with an error of less than a few % compared to the precalculation of the dose distribution by the physician. Above all, overdose to the patient must be absolutely avoided.

The tumor of the patient as a target is set at the beam iso-center with an accuracy of less than 1mm. In case of the abdominal organ as the target, it is subject to move by breathing, and the margin of irradiated field should be considered in the treatment planning. Since the shape, volume and position of patient's target and the planned dose distribution are different for each patient, setting of many kinds of devices in the beam port varies at the time of each irradiation. The size of most treatment is satisfactory within a maximum field of 22cm in diameter which is based on clinical experiences at NIRS. On the other hand, small fields such as less than 1cm are often required. Hence, devices of beam port must be accurately adapted for wide range of field size.

2) Irradiation time per patient must be less than a few minutes. The reason is that the patient is immobilized on the couch by the shell or capsule, and immobilization of longer time gives much stress to the patient with illness. Now we are estimating that it takes about ten minutes to set a patient for positioning on the couch. Therefore, treatment time that a patient stays in the treatment room is about fifteen minutes. HIMAC has two synchrotron rings and three treatment rooms (Fig. 1). In the room B, horizontal and vertical beams can be utilized at the same time, and the room A and the room C have the vertical and the horizontal beam course respectively. Accordingly, two beams are delivered to four beam ports alternately. The course of each beam is changed at interval of about ten minutes, and the beam should be immediatly adjusted in compliance with medical requirement for each patient. To realize such a rapid change, all magnets along the beam transport lines are actuated by corresponding treatment schedule and the beam course can be changed by setting only one switching magnet in HEBT(high energy beam transport) line. For these reason, switching magnet must have accurate reproducible setting of field strength and high stability.
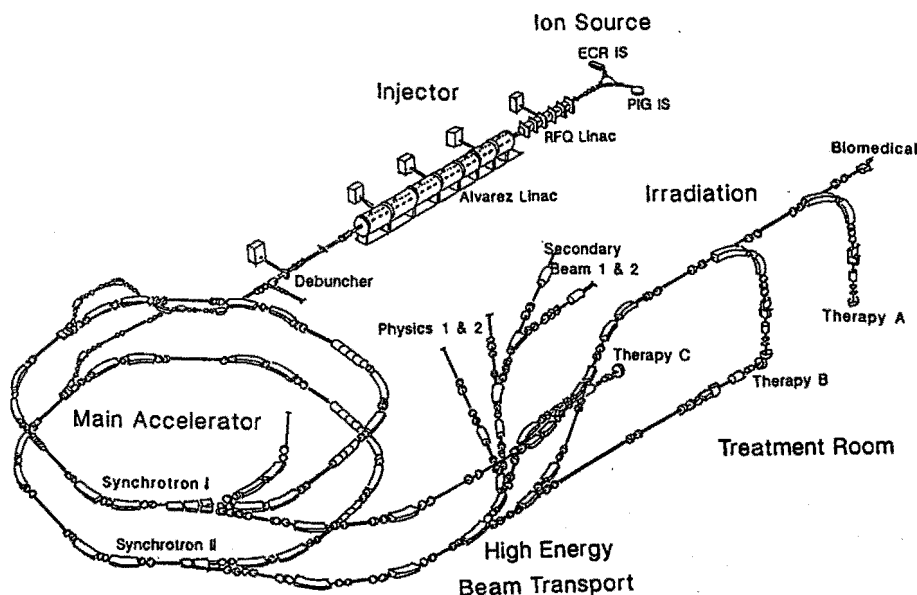

Fig. 1 A schematic view of accelerators and beam lines

3) HIMAC is expected hopefully to be utilized for more than fifty patients everyday for clinical treatments. In order to realize this, the treatment schedule and the flow of patients must be managed to run smoothly.

4) All systems in HIMAC are designed under the consideration of primary importance for keeping safety of patients and collaborating staffs. Especially, interlock system associated with the beam must be carefully designed from the view point of reliability.

III. TREATMENT SYSTEM

A schematic view of the treatment control network that we designed is shown in Fig. 2. The treatment system of HIMAC consists of the following components.

A. Treatment Planning System

In order to make the best use of heavy ion beam's characteristics, we are developing a three dimensional treatment planning system using a super graphics workstation (TITAN750) [2]. This planning system consists of:

1) defining target volume and critical organs by interactive contouring on Xray-CT, MRI and PET images,
2) determination of directions and shapes of irradiation fields using beam's eye view graphics,
3) calculation and display of three dimensional dose distributions,
4) designing collimators and compensators,
5) generating digitally reconstructed radiographs which are used for patient positioning.

In addition, the treatment supporting computer converts the planning data to the treatment control data, which consist of beam parameters, port data and treatment couch parameters.

B. Patient Positioning system

For patient positioning, we usually use the laser pointers, light localizer and digital Xray TVs that are incorporated in the beam port. The three dimensional coordinates of the target are estimated by coordinates of anatomical landmarks in the process of reconstruction by two projected Xray images perpendicular to each other[3]. Referencing the digitally reconstructed radiographs that are generated at the planning, the transfer of treatment couch is determined. The couch is to be operated by the treatment control computer(HP9000/380) linked to the patient positioning computer(HP9000/730) with image devices. Further, CT scanner can be used for the check of patient verification.

C. Irradiation system

Irradiation managing computer(HP9000/380) communicates with HIMAC central supervisor computer. It gives a requirement of the beam course, beam energy and ion species to HEBT controller via supervisor along the schedule. Concerning the responsibility of beam irradiation, we use the name "RIGHT" which means the initiative of opening the neutron shutter and the FCN(one of the Faraday cup monitors) shutter. The irradiation , that is the on/off of the beam, should be under the control of treatment side. Usually, irradiating at the treatment room starts after RIGHT is transfered to the treatment control.

Devices of irradiation beam port (Fig. 3) comprise a pair of wobbler magnets, a beam scatterer, a range shifter, a ridge filter, a multileaf collimator and monitors. They are controlled by the treatment control computer via interface unit.

To protect from accidental irradiation to patients and collaborating medical staffs, interlock systems are carefully built against such occasions as probable overdose, various kinds of troubles in each instrument, change of beam intensity and change of patient's condition including his unexpected movement. Further, quick stop and restart of irradiation are required repeatedly for medical use. In consideration of these things, operation of opening and closing the neutron shutter and the FCN shutter are handled either automatically or manually. They are synthesized in the form of global interlock system which is driven with threefold safety through hardware and software.

IV. OPERATION OF IRRADIATION

A layout of treatment room floor is shown in Fig. 4. Chief radiation therapy technologist(RTT) sitting by the irradiation managing computer can always look over the treatment hall and watch the ITVs coming from treatment rooms. In addition, the status of patients and beam lines are displayed on his console. So he manages the schedule of irradiations for smooth running.
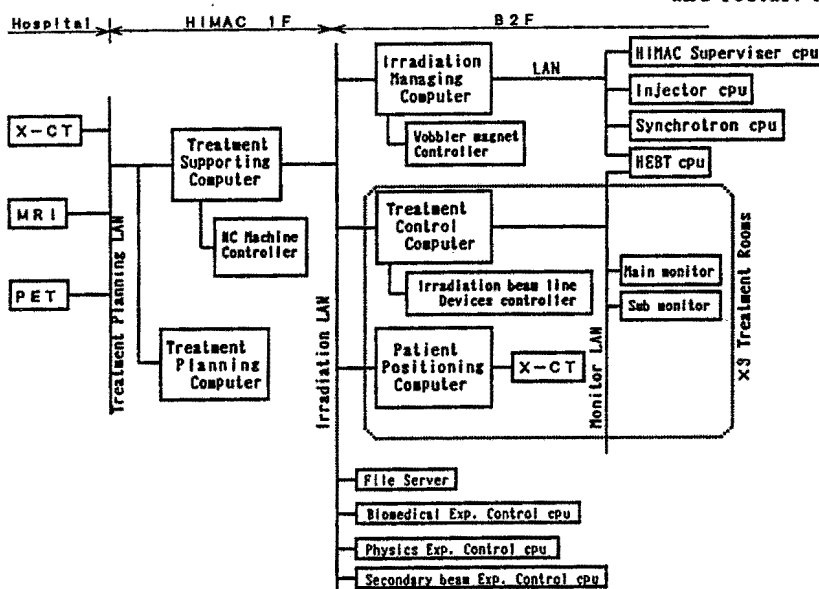

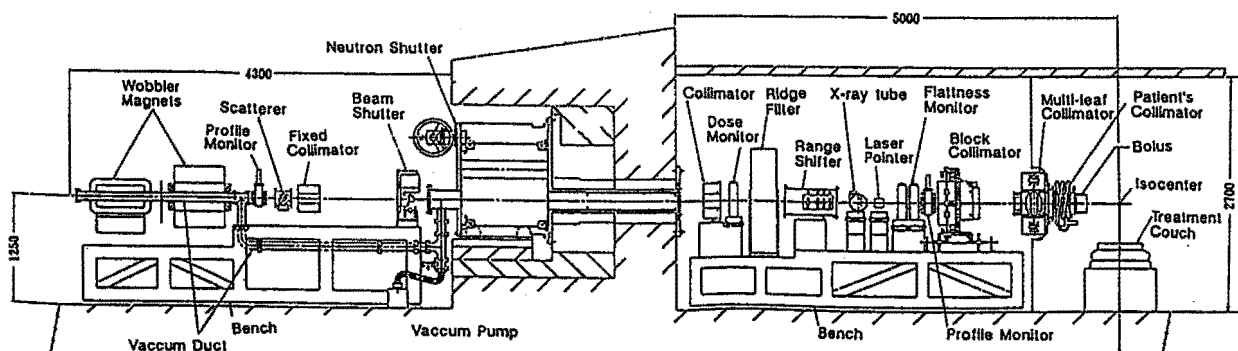Fig. 2 HIMAC treatment computer network

Fig. 3 A layout of beam port in the horizontal line

In ordinary clinical use, every morning, at first, RTT calibrates monitors in the same conditions of various devices just as the irradiation for each patient. Each parameter file which contains beam course, beam energy, ion species, setting parameters of magnets and so on is saved in the treatment control computer. This file name is checked before corresponding patient's irradiation, and the irradiation starts in the same condition at the time of calibration. The patient's irradiation starts after these.

Following the check of patient's ID-card, devices of beam port are automatically set on the starting status. While the RTT is setting the patient on the treatment couch, operators of HEBT adjust the beam up to the position before the neutron shutter, and keep waiting as the beam is stopped by FCN shutter. After the patient's setting, the RTT goes out from the treatment room and closes the shield door. The status of shield door is connected to global interlock system. And next, the RTT requests the RIGHT to HEBT. After the RIGHT is transferred to the treatment control, he opens the neutron

shutter. It takes about ten seconds to open. Then RTT opens the FCN shutter which takes less than one second, and the irradiation starts. Besides global interlock system, whenever RTT wants to suspend irradiation depending on the patient's condition, he can shut the beam and restart quickly afterward. Once the dose counting of main-monitor reaches preset counts, kicker magnet kicks off the beam, and then FCN and neutron shutter are closed through the interlock system. A main-monitor system is always backed by sub-monitor system. Then RIGHT returns to HEBT with the data of request to next beam course. The data of irradiation records such as final counts of monitors, times of suspension, positioning images and status of port devices are preserved for each patient.

Now, we are designing and developing the software and hardware to control the beam for radiation therapy in HIMAC. The clinical trial will start in 1994.

References
1) K.Kawachi et al., Proc. 2nd Int. Sympo. Advanced Nuclear Energy Research, p.362 (1990)
2) M.Endo et al., Proc. of Int. Conf. on MBE, vol.29, p.317 (1991)
3) S.Minohara et al., Proc. of Int. Conf. on MBE, vol.29, p.859 (1991)
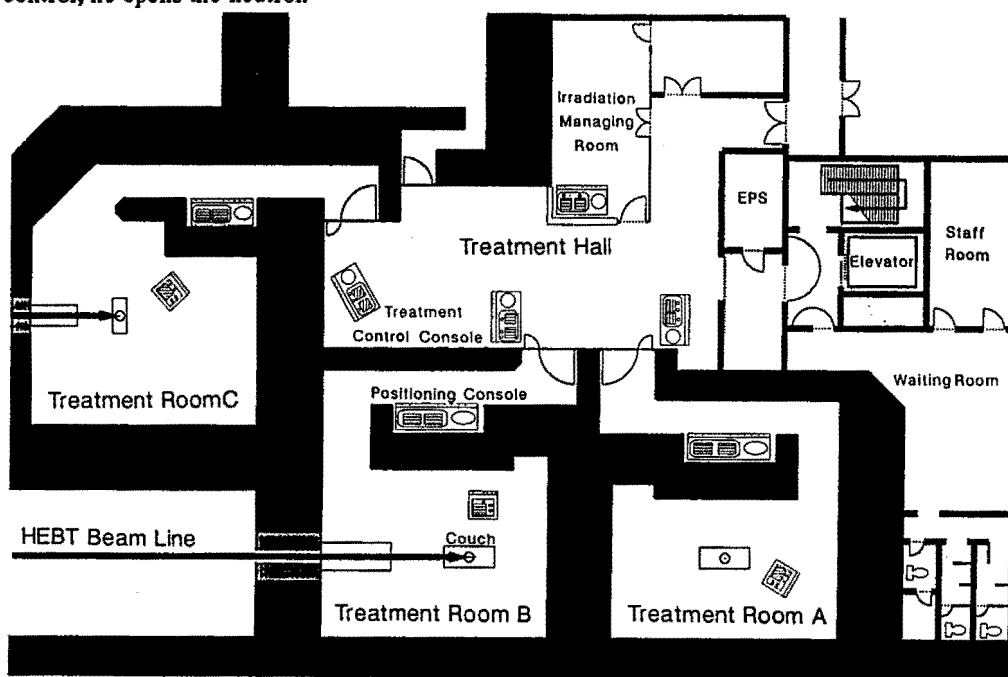
Fig. 4 A schematic view of treatment floor (HIMAC B2F)

List of Authors

Bold papercodes indicate primary authors; ~~strike through/black~~ papercodes indicate **no submission received**

— A —	
Abe, I.	S02SRU09
Agueev, A.I.	**S04SRS04**
Akiyama, A.	S02SRU08, S04SRS08
Alferov, V.	**S03SRD08**, S04SRS04
Anderson, J.B.	S03SRD04
Anderson, M.D.	S03SRD04
Aoki, K.	S02SRU10
Arnold, N.D.	S03SRD04
Asami, K.	S03SRD14
Azumaishi, R.	S03SRD14

— B —	
Bailey, R.	**S01SRA01**
Barton, C.J.	S02SRU05
Bassato, G.	**S01SRA06**
Battistella, A.	S01SRA06
Baumann, R.	S01SRA05
Bellato, M.A.	S01SRA06
Billinge, R.	**S04SRS02**
Björklund, E.	S01SRA02, S02SRU01
Bret, A.	S04SRS02
Brook, V.L.	S03SRD08
Brownless, D.M.	S02SRU05
Burns, M.J.	S01SRA02, S02SRU01

— C —	
Caccia, B.	S03SRD06
Callaway, T.	S01SRA02, S02SRU01
Canella, S.	S01SRA06
Carr, G.P.	S01SRA02, S02SRU01
Cavallari, G.	**S04SRS01**
Cha, B.-C.K.	S03SRD04
Chang, S.S.	S03SRD11
Chen, C.J.	S03SRD10
Chen, J.	S03SRD10
Cheng, H.J.	S04SRS06
Chepurnov, A.S.	**S03SRD09**
Chin, M.J.	S01SRA03
Chu, Z.S.	S01SRA11
Chuang, C.Z.	S04SRS06
Ciapala, E.	S04SRS01
Cohen, S.	S01SRA02, S02SRU01
Cuttone, G.	S03SRD07

— D —	
Daly, R.T.	S03SRD04
Dante, V.	S03SRD06
David, L.	S02SRU04
Deloose, I.	S04SRS02
Di Pirro, G.	**S03SRD06**
Dunaitsev, A.F.	S03SRD08, S04SRS04

— E —	
Endo, M.	S04SRS10

Engstrom, M.	S04SRS03

— F —	
Fahmie, M.P.	S01SRA03
Franck, A.R.	S02SRU11
Furukawa, K.	**S02SRU09**

— G —	
Gajewski, K.J.	**S02SRU06**
Giove, D.	**S03SRD07**
Goloborodko, S.G.	S03SRD08
Gribov, I.V.	S03SRD09
Gridassov, V.I.	S04SRS04
Guertsev, K.F.	S04SRS04
Gunderson, G.R.	S03SRD04
Gussak, A.A.	S04SRS04

— H —	
Harrington, M.	S01SRA02, S02SRU01
Hirao, Y.	S03SRD14
Huang, H.	S04SRS06
Huang, J.	S03SRD11
Huang, T.H.	S01SRA11
Humphrey, J.	**S01SRA04**
Hunt, S.M.	**S03SRD02**

— I —	
Ieiri, M.	S04SRS09
Ishida, T.	S04SRS08
Ishii, H.	S04SRS09
Itano, A.	S03SRD14
Itoh, Y.I.	S03SRD13
Izgarshev, S.V.	S03SRD08

— J —	
Jan, G.J.	**S03SRD10**
Jiao, T.S.	**S01SRA11**
Johansson, O.	S02SRU06

— K —	
Kadokura, E.	S04SRS08
Kamikubota, N.	S02SRU09
Kanai, T.	S04SRS10
Kanazawa, M.	S03SRD14
Kapps, E.	S01SRA05
Karonis, N.T.	S03SRD04
Katayama, T.	S02SRU10
Kato, Y.	S04SRS09
Kawachi, K.	S04SRS10
Kawakubo, T.	**S04SRS08**
Kawamoto, T.	S02SRU08
Kazarian, A.	S04SRS05
Kerr, J.C.	S02SRU05
Khoetsian, N.	S04SRS05
Kirn, J.H.	S03SRD11
Kissler, K.H.	**S02SRU03**, **S03SRD01**

Klotz, W.D.	**S03SRD05**	Nawrocki, G.J.	S03SRD04
Knaebel, R.	S01SRA05	Noda, K.	S03SRD14
Knott, M.J.	S03SRD04	Noumi, H.	S04SRS09
Kohno, T.	**S03SRD14**, S04SRS10		

— O —

Komada, K.	S02SRU08	Ohnishi, K.	S02SRU10

Komarov, V.V.	S03SRD08

— P —

Kotov, V.M.	**S01SRA12**	Pace, A.	S04SRS02
Kraimer, M.R.	S03SRD04	Peck, S.B.	S03SRD03
Krause, U.	**S01SRA07**	Perriollat, F.	S03SRD01
Krendelev, V.	S04SRS04	Persigny, J.	S01SRA05
Kubicek, D.	S01SRA02, S02SRU01	Poore, R.	S01SRA02, S02SRU01
Kudoh, K.	S02SRU08	Pose, R.A.	S01SRA12
Kuiper, B.	S03SRD08	Proshin, V.M.	S04SRS04
Kumahara, T.	S03SRD13, S03SRD12		

— R —

Kurokawa, S.-I.	S02SRU08	Rabany, M.	S03SRD01

— L —

		Rausch, R.	**S02SRU02**, S02SRU03
Lackey, S.L.	S02SRU11	Rawlinson, W.R.	S01SRA08
Lancaster, H.	S01SRA03	Rice, D.H.	S03SRD03
Lécorché, E.	S02SRU04	Robb, A.	S01SRA03
Lee, J.	S03SRD11	Rovelli, A.	S03SRD07

Lee, J.W.	S03SRD11

— S —

Lenkszus, F.	S03SRD04	Sakharov, V.P.	S03SRD08
Li, T.Y.	S01SRA11	Sato, K.	S03SRD14
Lin, Q.	S04SRS06	Schaa, V.RW.	S01SRA07
Liu, D.K.	**S04SRS06**	Schaller, S.	**S01SRA02**, S02SRU01
Liu, S.-Y.	S02SRU07	Scherbakov, E.D.	S03SRD08
Lomoro, R.	S03SRD06	Schultz, D.	S01SRA02, S02SRU01
Low, K.	S03SRD02	Seino, K.	S02SRU11
Lu, P.	S04SRS06	Serio, M.	S03SRD06
Lukyantsev, A.	S03SRD08, S04SRS04	Serre, Ch.	S02SRU02
Luong, T.T.	**S02SRU04**	Shen, Z.	S01SRA11
Lutz, J.R.	**S01SRA05**	Shering, G.	S03SRD01, S04SRS02
		Shumakov, A.V.	S03SRD09

— M —

		Smolucha, J.	**S02SRU11**
Ma, S.	S01SRA11	Soga, F.	S04SRS10
Magyary, S.	**S01SRA03**	Solovyov, V.E.	S04SRS04
Mannix, R.P.	**S02SRU05**	Spiriti, E.	S03SRD06
Marsaudon, J.C.	S01SRA05	Starker, J.	**S04SRS03**
Martlew, B.G.	**S01SRA08**	Stecchi, A.	S03SRD06
Masuda, T.	S03SRD12	Steiner, R.	S01SRA07
McCarthy, M.	S01SRA08	Strohman, C.R.	**S03SRD03**
McDowell, W.P.	**S03SRD04**	Stuewe, R.	S01SRA02, **S02SRU01**
Mikawa, K.	S04SRS07	Su, Y.H.	S04SRS06
Mikheev, M.S.	S03SRD08	Sudou, M.	S03SRD14, S04SRS10
Milardi, C.	S03SRD06	Sueno, T.	**S04SRS07**
Mimashi, T.	**S02SRU08**	Suzuki, Y.	**S04SRS09**
Minakawa, M.	S04SRS09	Sytin, A.	S04SRS04
Minohara, S.	**S04SRS10**		

— T —

Molinari, P.	S01SRA03	Takada, E.	S03SRD14
Montoya, T.	S04SRS07	Takasaki, M.	S04SRS09
Morii, Y.	S03SRD14	Takeda, S.	S02SRU08
Morozov, S.Yu.	S03SRD09	Tanaka, K.H.	S04SRS09

— N —

		Thuresson, L.	S02SRU06
Naitoh, T.	S02SRU08	Timossi, CA.	S01SRA03
Nakahara, K.	S02SRU09		
Narusaka, H.	S03SRD14		

List of Authors

Toda, M.	S04SRS07		Winans, J.R.	S03SRD04
Toumanian, A.	**S04SRS05**		Won, S.C.	**S03SRD11**
Trasatti, L.	S03SRD06		— Y —	
Trofimov, N.N.	S03SRD08		Yamanoi, Y.	S04SRS09
Tsuzuki, N.	S03SRD14		Yang, L.P.	**S02SRU07**, S04SRS06
— U —			Yao, C.	**S01SRA10**, **S01SRA09**
Ulrich, M.	S02SRU04		Yei, K.R.	S04SRS06
Urakawa, J.	S02SRU08		Yonehara, H.	S03SRD12
Ustinov, E.A.	S04SRS04		Yoshikawa, H.	**S03SRD13**, S03SRD12
— V —			Yoshizawa, J.	S02SRU10
Vaguine, A.I.	S03SRD08		Young, J.	S01SRA03
Valentini, S.	S03SRD06		Yourpalov, V.D.	S03SRD08
Voevodin, V.P.	S03SRD08		Yue, W.	S04SRS06
Vong, F.C.	S03SRD04		— Z —	
— W —			Zapolski, N.	S04SRS05
Wada, T.	**S03SRD12**		Zelepoukin, S.A.	S03SRD08
Wang, C.S.	S03SRD10		Zhen, W.	S03SRD12
Wang, L.Z.	S02SRU07		Zhou, X.	S01SRA11
Wang, Z.	S01SRA11		Zieman, R.C.	S03SRD04
Watanabe, S.	**S02SRU10**		Zinoviev, A.V.	S03SRD09

List of Authors

List of Authors

Institutes List

ANL
Lemont, Illinois, USA
- Anderson, J.B.
- Anderson, M.D.
- Arnold, N.D.
- Cha, B.-C.K.
- Daly, R.T.
- Gunderson, G.R.
- Karonis, N.T.
- Knott, M.J.
- Kraimer, M.R.
- Lenkszus, F.
- McDowell, W.P.
- Nawrocki, G.J.
- Vong, F.C.
- Winans, J.R.
- Yao, C.
- Zieman, R.C.

CERN
Meyrin, Switzerland
- Bailey, R.
- Billinge, R.
- Bret, A.
- Cavallari, G.
- Ciapala, E.
- Deloose, I.
- Kissler, K.H.
- Kuiper, B.
- Pace, A.
- Perriollat, F.
- Rabany, M.
- Rausch, R.
- Serre, Ch.
- Shering, G.

CNRS/IN2P3
Strasbourg Cedex, France
- Baumann, R.
- Kapps, E.
- Knaebel, R.
- Lutz, J.R.
- Marsaudon, J.C.
- Persigny, J.

Cornell University (CLASSE), Cornell Laboratory for Accelerator-Based Sciences and Education
Ithaca, New York, USA
- Peck, S.B.
- Rice, D.H.
- Strohman, C.R.

DEC-Japan
Tokyo, Japan
- Narusaka, H.

ESRF
Grenoble, France
- Klotz, W.D.

Fermilab
Batavia, Illinois, USA
- Franck, A.R.
- Lackey, S.L.
- Seino, K.
- Smolucha, J.

GANIL
Caen, France
- David, L.
- Luong, T.T.
- Lécorché, E.
- Ulrich, M.

GSI
Darmstadt, Germany
- Krause, U.
- Schaa, V.RW.
- Steiner, R.

Hitachi, Ltd.
Ibaraki-ken, Japan
- Asami, K.
- Azumaishi, R.

IHEP
Moscow Region, Russia
- Agueev, A.I.
- Alferov, V.
- Brook, V.L.
- Cheng, H.J.
- Chuang, C.Z.
- Dunaitsev, A.F.
- Goloborodko, S.G.
- Gridassov, V.I.
- Guertsev, K.F.
- Gussak, A.A.
- Huang, H.
- Izgarshev, S.V.
- Komarov, V.V.
- Krendelev, V.
- Lin, Q.
- Liu, D.K.
- Liu, S.-Y.
- Lu, P.
- Lukyantsev, A.
- Mikheev, M.S.
- Proshin, V.M.
- Sakharov, V.P.
- Scherbakov, E.D.
- Solovyov, V.E.
- Su, Y.H.
- Sytin, A.
- Trofimov, N.N.
- Ustinov, E.A.
- Vaguine, A.I.
- Voevodin, V.P.
- Wang, L.Z.
- Yang, L.P.
- Yei, K.R.

- Yourpalov, V.D.
- Yue, W.
- Zelepoukin, S.A.

IMP
Lanzhou, People's Republic of China
- Chu, Z.S.
- Huang, T.H.
- Jiao, T.S.
- Li, T.Y.
- Ma, S.
- Shen, Z.
- Wang, Z.
- Zhou, X.

INFN/LASA
Segrate (MI), Italy
- Giove, D.

INFN/LNL
Legnaro (PD), Italy
- Bassato, G.
- Battistella, A.
- Bellato, M.A.
- Canella, S.

INFN/LNS
Catania, Italy
- Cuttone, G.
- Rovelli, A.

INS
Tokyo, Japan
- Katayama, T.
- Watanabe, S.
- Yoshizawa, J.

ISS
Rome, Italy
- Caccia, B.
- Dante, V.
- Lomoro, R.
- Spiriti, E.
- Valentini, S.

JAERI
Tokai-Mura, Naka-Gun, Ibaraki-Ken, Japan
- Itoh, Y.I.
- Kumahara, T.
- Yoshikawa, H.

JAERI-RIKEN/Spring-8 Project Team
Tokyo, Japan
- Kumahara, T.
- Masuda, T.
- Wada, T.
- Yonehara, H.
- Yoshikawa, H.
- Zhen, W.

JINR
Dubna, Moscow Region, Russia
- Kotov, V.M.
- Pose, R.A.

KEK
Ibaraki, Japan
- Abe, I.
- Akiyama, A.
- Furukawa, K.
- Ieiri, M.
- Ishida, T.
- Ishii, H.
- Kadokura, E.
- Kamikubota, N.
- Kato, Y.
- Kawakubo, T.
- Kawamoto, T.
- Komada, K.
- Kudoh, K.
- Kurokawa, S.-I.
- Mikawa, K.
- Mimashi, T.
- Minakawa, M.
- Montoya, T.
- Naitoh, T.
- Nakahara, K.
- Noumi, H.
- Sueno, T.
- Suzuki, Y.
- Takasaki, M.
- Takeda, S.
- Tanaka, K.H.
- Toda, M.
- Urakawa, J.
- Yamanoi, Y.

LANL
Los Alamos, New Mexico, USA
- Björklund, E.
- Burns, M.J.
- Callaway, T.
- Carr, G.P.
- Cohen, S.
- Harrington, M.
- Kubicek, D.
- Poore, R.
- Schaller, S.
- Schultz, D.
- Stuewe, R.

LBNL
Berkeley, California, USA
- Chin, M.J.
- Fahmie, M.P.
- Lancaster, H.
- Magyary, S.
- Molinari, P.
- Robb, A.
- Timossi, CA.
- Young, J.

LNF-INFN
Frascati, Italy
- Di Pirro, G.
- Milardi, C.
- Serio, M.

Institutes List

- Stecchi, A.
- Trasatti, L.

Mitsubishi Electric Corporation
Tokyo, Japan
- Ishida, T.

MSL
Stockholm, Sweden
- Engstrom, M.
- Starker, J.

MSU
Moscow, Russia
- Chepurnov, A.S.
- Gribov, I.V.
- Morozov, S.Yu.
- Shumakov, A.V.
- Zinoviev, A.V.

NIRS
Chiba-shi, Japan
- Asami, K.
- Azumaishi, R.
- Endo, M.
- Hirao, Y.
- Itano, A.
- Kanai, T.
- Kanazawa, M.
- Kawachi, K.
- Kohno, T.
- Minohara, S.
- Morii, Y.
- Narusaka, H.
- Noda, K.
- Sato, K.
- Soga, F.
- Sudou, M.
- Takada, E.
- Tsuzuki, N.

NSRRC
Hsinchu, Taiwan
- Chen, C.J.
- Chen, J.
- Jan, G.J.
- Wang, C.S.

NTUT
Taipei, Taiwan
- Wang, C.S.

PAL
Pohang, Kyungbuk, Republic of Korea
- Chang, S.S.
- Huang, J.
- Kirn, J.H.
- Lee, J.W.
- Lee, J.
- Won, S.C.

RAL
Chilton, Didcot, Oxon, United Kingdom

- Barton, C.J.
- Brownless, D.M.
- Kerr, J.C.
- Mannix, R.P.

SERC
Daresbury, Warrington, Cheshire, United Kingdom
- Martlew, B.G.
- McCarthy, M.
- Rawlinson, W.R.

SHI
Tokyo, Japan
- Aoki, K.
- Ohnishi, K.

SLAC
Menlo Park, California, USA
- Humphrey, J.

SSCL
Dallas, USA
- Hunt, S.M.
- Low, K.

TMEIC
Tokyo, Japan
- Morii, Y.

Toshiba Mitsubishi Electric Industrial Systems Corporation
Tokyo, Japan
- Tsuzuki, N.

TSL
Uppsala, Sweden
- Gajewski, K.J.
- Johansson, O.
- Thuresson, L.

USTC/NSRL
Hefei, Anhui, People's Republic of China
- Yao, C.

YerPhI
Yerevan, Armenia
- Kazarian, A.
- Khoetsian, N.
- Toumanian, A.
- Zapolski, N.

Institutes List

Participants List

Abe, Isamu *(KEK)*
Akiyama, Atsuyoshi *(KEK)*
Alferov, Vladimir *(IHEP-Protvino)*
Allen, Mike *(SSCL)*
Amtey, Sharad *(SSCL)*
Araki, Sakae *(KEK)*
Bailey, Roger *(CERN)*
Baribaud, Guy *(CERN)*
Barton, Carole Joan *(RAL)*
Barton, Donald S. *(BNL)*
Batskikh, G. *(MRI)*
Beetham, Christopher Gary *(CERN)*
Blumer, Thomas *(PSI)*
Bork, Rolf *(SSCL)*
Bourianoff, George I. *(SSCL)*
Briegel, Charles I. *(Fermilab)*
Bulfone, Daniele *(INFN-Trieste)*
Burns, Alan James *(CERN)*
Busse, Winfried *(HMI)*
Cahill, Kevin *(Fermilab)*
Canella, Stefania *(INFN-LNL)*
Cavallari, Giorgio *(CERN)*
Chang, Suk Sang *(POSTECH)*
Chepurnov, Alexander S. *(MSU)*
Chohan, Vinod *(CERN)*
Ciapala, Edmond *(CERN)*
Ciriani, Paolo *(CERN)*
Clausen, Matthias *(DESY)*
Clifford, Tom S. *(BNL)*
Clout, Peter Norman *(VISTA)*
Coffman, Lindsay *(SSCL)*
Crowley-Milling, Michael
Cuttone, Giacomo *(INFN-LNS)*
Dalesio, Leo Robert *(LANL)*
Daneels, Axel Jean *(CERN)*
Dasgupta, Subrata *(VECC)*
Dimaio, Franck *(CERN)*
Di Pirro, Giampiero *(INFN-LNL)*
Dickey, Carl Edward *(SSCL)*
Dohan, Don A. *(SSCL)*
Dunaitsev, Anatoly F. *(IHEP-Serpukhov)*
Emura, Katsuji *(Sumitomo E.I.-Osaka)*
Engström, Mats *(MSI)*
Ermolin, Yuri *(IHEP-Serpukhov)*
Frolov, I. *(MRI)*
Furukawa, Kazuro *(KEK)*
Gajewski, Konrad Jan *(TSL)*
Giove, Dario *(INFN-LASA)*
Götz, Andy *(ESRF)*
Grant, Peter Angus *(TRIUMF)*
Gribov, Igor V. *(MSU)*
Gurd, David P. *(SSCL)*
Hacker, Uli *(KFA)*
Haenni, David Richard *(SSCL)*
Hama, Hiroyuki *(IMS)*
Hanashima, Susumu *(JAERI-Tokai)*

Harada, Shunji *(MELCO-NFDD)*
Heinze, Wolfgang *(CERN)*
Hendricks, Brian *(Fermilab)*
Hendrickson, Linda Joyce *(SLAC)*
Heubers, Wim *(NIKHEF)*
Hien, Pham Zuy *(CANT)*
Hsu, K.T. *(SRRC)*
Huang, Jung-Yun *(POSTECH)*
Humphrey, John (Rusty) *(SLAC)*
Hunt, Steve *(SSCL)*
Iselin, Christoph F. *(CERN)*
Ishii, Kazuhiro *(KEK)*
Jan, Gwo-Jen *(SRRC)*
Jiao, Tian-shu *(IMP)*
Joshel, Robert H. *(Fermilab)*
Kabe, Seiji *(KEK)*
Kadokura, Eiichi *(KEK)*
Kamikubota, Norihiko *(KEK)*
Kanaya, Noriichi *(KEK)*
Kanazawa, Mitsutaka *(NIRS)*
Kaneko, Hiroshi *(NIFS)*
Kase, Masayuki *(RIKEN)*
Katoh, Tadahiko *(KEK)*
Katsura, Tomotaro *(KEK)*
Kawakubo, Tadamichi *(KEK)*
Kawamoto, Takashi *(KEK)*
Kijima, Yuko *(MELCO-NFDD)*
Kimura, Toyoaki *(JAERI-Naka)*
Kimura, Yoshitaka *(KEK)*
Kishiro, Jun-ichi *(KEK)*
Kitamura, Masaharu *(LNS)*
Klotz, Wolf-Dieter *(ESRF)*
Knott, Martin John *(ANL)*
Kohno, Toshiyuki *(NIRS)*
Komarov, Vladimir *(IHEP-Protvino)*
Kouno, Satoshi
Kraimer, Martin R. *(ANL)*
Kubota, Tetsuya *(Kawasaki H.I.)*
Kudo, Kikuo *(KEK)*
Kuiper, Berend *(CERN)*
Kumada, Masayuki *(NIRS)*
Kumahara, Tadashi *(JAERI-Tokai)*
Kuper, Edward *(INP-Novosibirsk)*
Kurokawa, Shin-ichi *(KEK)*
Lécorché, Eric *(GANIL)*
Lee, Martin Joe *(SLAC)*
Le Goff, Jean-Marie H. *(CERN)*
Liu, Shi-Yao *(IHEP-Beijing)*
Lucas, Peter Wayne *(Fermilab)*
Ludgate, George Arthur *(TRIUMF)*
Lukyantsev, Alexander F. *(IHEP-Serpukhov)*
Luong, Tam T. *(GANIL)*
Lutz, Jean-Robert *(IN2P3-CNRS)*
Magyary, Steven Bela *(LBL)*
Mannix, Rbert Patrick *(RAL)*
Marsaudon, Jean-Claude *(IN2P3-CNRS)*

Participants List

Masuda, Takemasa *(RIKEN)*
Matsutomi, Shokichi *(Toshiba-Fuchu)*
Matsuzaki, Yoshimi *(JAERI-Naka)*
McDowell, William P. *(ANL)*
Mertens, Volker *(CERN)*
Miki, Tetsushi *(PNC-Oarai)*
Mimashi, Toshihiro *(KEK)*
Minohara, Shinichi *(NIRS)*
Miura, Takeo *(Nippon Eng.)*
Morita, Koh-Ichiro *(NAO-NRO)*
Mozin, Igor *(Efremov)*
Mueller, Klaus-Dieter *(KFA)*
Murai, Katsuji *(Hitachi-Omika)*
Murin, Boris P. *(MRI)*
Mutoh, Masakatsu *(LNS)*
Nakabayashi, Shiro *(Kawasaki H.I.)*
Nakagawa, Hidetoshi *(KEK)*
Nakamura, Takeshi *(RIKEN)*
Navratil, Jiri *(CTU)*
Nguyen, Huan T. *(SSCL)*
Nigorikawa, Kazuyuki *(KEK)*
Ninane, Alain H. *(UCL)*
Nishimura, Hiroshi *(LBL)*
Nishio, Masanori *(NAO-NRO)*
Ogata, Hiroshi *(RCNP)*
Okumura, Susumu *(JAERI-Takasaki)*
Otake, Yuji *(KEK)*
Owen, Edward Charles *(Daresbury)*
Oyamada, Masayuki *(LNS)*
Pace, Alberto *(CERN)*
Pak, Cheol On *(KEK)*
Peters, Franz *(DESY)*
Pose, Rudolf *(JINR)*
Potepan, Franco *(INFN-Trieste)*
Protasov, Yuri S. *(MSU)*
Rabany, Michel *(CERN)*
Raffi, Gianni *(ESO)*
Raich, Ulrich *(CERN)*
Rausch, Raymond *(CERN)*
Rawlinson, William Robert *(Daresbury)*
Reinhard-Nickulin, Petr I. *(INR)*
Riedel, Larry Dean *(SSCL)*
Saban, Roberto *(CERN)*
Sarkar, Surajit *(SSCL)*
Saw, Sor-Heoh *(Univ. Malaya)*
Schaa, Volker R.W. *(GSI)*
Schaller, Stuart *(LANL)*
Schekochihin, Arkady V. *(INP-Protvino)*
Schimmel, Fred *(NIKHEF)*
Schmidt, Volker *(IGI)*
Serre, Christian *(CERN)*
Serre, Claudia *(IN2P3-Paris)*
Shea, Michael F. *(Fermilab)*
Shering, George Craig *(CERN)*
Shibasaki, Yoshinobu *(LNS)*
Shibata, Shinkichi *(TCT)*
Shirakawa, Akihiro *(KEK)*
Skarin, Igor A. *(INP-Protvino)*

Skelly, Joseph Francis *(BNL)*
Starker, Josef *(MSI)*
Stecchi, Alessandro *(INFN-LNF)*
Steiner, Rudolf *(GSI)*
Strohman, Charles Ralph *(Cornell)*
Stuewe, Robert B. *(LANL)*
Sugimoto, Masayoshi *(JAERI-Tokai)*
Suwa, Shigeki
Suzuki, Yoshihiro *(KEK)*
Takada, Eiichi *(NIRS)*
Takahashi, Minoru *(JAERI-Naka)*
Takahashi, Mitsuyuki *(IHI)*
Takebe, Hideki *(RIKEN)*
Takeda, Shigeru *(KEK)*
Tanabe, Eiji *(AET-Japan)*
Tanaka, Shigeru *(Sumitomo E.I.-Yokohama)*
Tanaka, Hitoshi *(RIKEN)*
Trahern, Charles Garrett *(SSCL)*
Trasattil, Luciano *(INFN-LNF)*
Trofimov, Nikoly Nik. *(IHEP-Serpukhov)*
Tsybin, Alexander S. *(MSTU)*
Tumanian, Arsen Rafael *(YPI)*
Tyrrel, Mark William *(CERN)*
Urakawa, Junji *(KEK)*
Utsumi, Masafumi *(MELCO-NFDD)*
Vaguine, Alexei *(MRI)*
Vande Vyvre, Pierre *(CERN)*
Wada, Takeshi *(RIKEN)*
Wang, Zhen *(RIKEN)*
Watanabe, Shin-ichi *(INS)*
Weng, C.S. *(SRRC)*
Wilkinson, Neil Adam *(TRIUMF)*
Won, Sangchul *(POSTECH)*
Xu, Weimin *(HESYRL)*
Xue, Jing-Xuan *(IHEP-Beijing)*
Yagi, Takao *(Toshiba-Fuchu)*
Yamada, Kohji *(NKK)*
Yamamoto, Noboru *(KEK)*
Yamazaki, Kozo *(NIFS)*
Yao, Chih-Yuan *(HESYRL)*
Yonekawa, Izuru *(JAERI-Naka)*
Yasunobu, Seiji *(Hitachi-Ozenji)*
Yoshikawa, Hiroshi *(JAERI-Tokai)*
Zagel, James R. *(Fermilab)*
Zaitsev, V.I. *(Kurchatov)*
Zlelepoukin, Sergei *(IHEP-Protvino)*