ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S02SRU09

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

ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S02SRU09

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

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.

ISBN: 978-3-95450-254-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS1991-S02SRU09

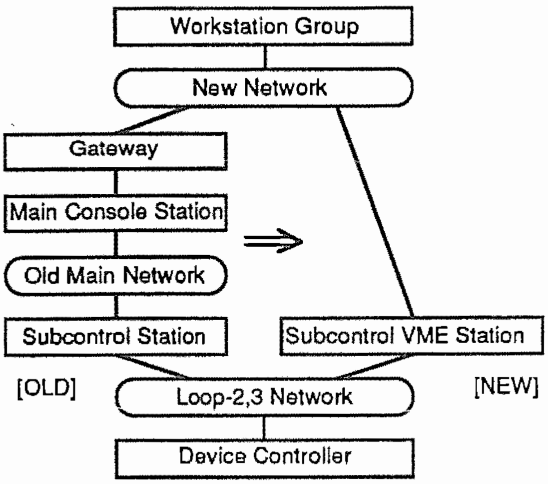

Figure 1. Transition of the system connection and the control message flow from the old system to the new system.

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.

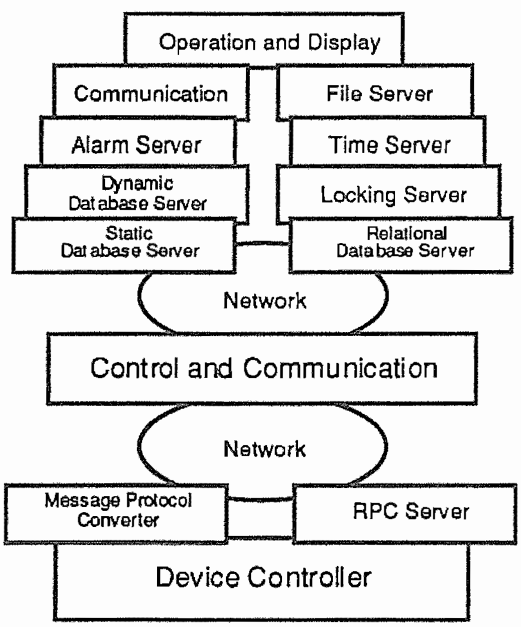

Figure 2. Logical structure of the new linac control system.

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

The authors wish to thank Professor A. Asami for his support during this work. They also would like to thank the staff and operators of the linac for their valuable discussions during the system design and development. One of the authors (K.F) is grateful to Dr. K. Kostro who allowed him to use the NC/RPC code of CERN.

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.