Chapter 13

Accelerator Control System

13.1 System Requirements

The KEKB accelerator complex has more than 50,000 control points along the 3 km circumference of the two rings, the LER and HER. Some control data must be updated at a high rate up to 100 Hz. A fast and powerful control system is required to implement sophisticated tuning procedures and feedback loops for the stable operation of KEKB. In addition, such a control system has to be readied in a timely fashion, *i.e.* it has to be operational at the first commissioning time of KEKB, which is expected to begin near the end of 1998. The schedule for the control system development is summarized in Table 13.1.

13.1.1 Functional Requirements

The functional requirements for the KEKB accelerator control system can be categorized into three groups as follows. Some of the important issues are itemized below:

1. *Hardware Control*:

   (a) It should be that the graphical user interface (GUI) to read/set/monitor the status of a hardware device can be constructed without resorting to complicated programming.

   (b) It should be that the data logging can be configured without complicated programming.

   (c) The low level access to individual hardware components should be supported for developing and debugging the hardware interface. The interface of this low level access should be user-friendly and flexible.

FY1995	April	Decision to use EPICS.
		Start development of software tools.
		Start user education on EPICS.
	July	Complete First stage of Relational Database Design.
	December	Complete Steering Power supply/ BPM interface design.
FY1996		Develop Application programs for AR/KEKB.
	December	System test of new TRISTAN AR control system.
FY1997	July	New TRISTAN-AR control system commissioning
		New TRISTAN/KEKB beam transport line control system
	December	Starts installation of KEKB accelerator control system equipment.
FY1998		
	December	Commissioning of KEKB.

Table 13.1: Schedule of the KEKB accelerator control system development.

(d) A powerful yet easy-to-handle development system should be provided for the device driver level software.

(e) Stand-alone operation of the hardware subsystem should be allowed, without depending on other part of the accelerator.

2. *Accelerator Operation*:

   Application programs required for this category include: automated machine operation software, monitors of the machine performance, feedback loops to optimize the machine performance, alarm notification system and others.

   (a) The user interface of these programs should be operator-friendly.

   (b) Operator interfaces should have a quick response; typically less than a few seconds.

   (c) An alarm notification system should be included.

   (d) All key operations should be recorded for later inspection.

   (e) Powerful yet easy-to-handle development system for the development of application programs with GUI.

3. *Accelerator Development*:

   Accelerator development refers to activities where various machine errors are identified and fixed, new tuning techniques are attempted, better operating conditions are searched for, or unresolved accelerator physics issues are investigated, and so on. Its guiding principles may be summarized as:

   (a) All data that can possibly be collected should be collected.

   (b) All collected data should be saved for later analyses.

   (c) All operations should be recorded for later inspection.

   The requirements on the data analysis tools are:

   (d) Stored data can be easily retrieved.

   (e) The data can be easily analyzed.

   (f) The data analysis tools should be able to handle both data from the accelerators and the results from computer simulation in a similar way.

   (g) The development system for application programs should be programmer-friendly.

In a particle accelerator for high energy physics, hardware modifications and upgrades to the machine configuration are common events in its life cycle. The following features should be incorporated so that the control system will be able to keep pace with the changes of the accelerator.

1. All machine parameters and data of individual machine components should be saved in a database. This is so that changes in the accelerators are automatically reflected in the control system.

2. Incremental upgrades and the replacement of components of the control system should be allowed.

3. Integration of control sub-systems within the accelerator control system should be done in a easy and seamless way.

13.1.2 Design Guideline

With the requirements stated above, the design considerations for the KEKB accelerator control system are summarized as follows. The keywords of the guidelines are "Openness", "Standard" and "Scalability":

1. For future maintenance and upgrades, the control system should have a layered structure. The layers described here refer to such concepts as: the device interface layer, the equipment control layer and the presentation layer. The interface protocol between these layers should be a well-defined one based on international standards. This layered structure will allow us to modify components in one layer without affecting other layers.

2. As the interface between the control system and the equipment, international standards such as CAMAC, VME, VXI and GP-IB should be used where their performance is found adequate.

3. To maintain good responsiveness of the control system, a high-speed network should be used to connect the computers. The network architecture should be chosen from well established, standard technologies.

4. To reduce the man-power requirement and efforts to build the control system, international collaboration and/or commercially available products should be taken advantage of.

5. At times, adopting object-oriented techniques or data abstraction for hiding the details of the hardware helps in the development of application programs

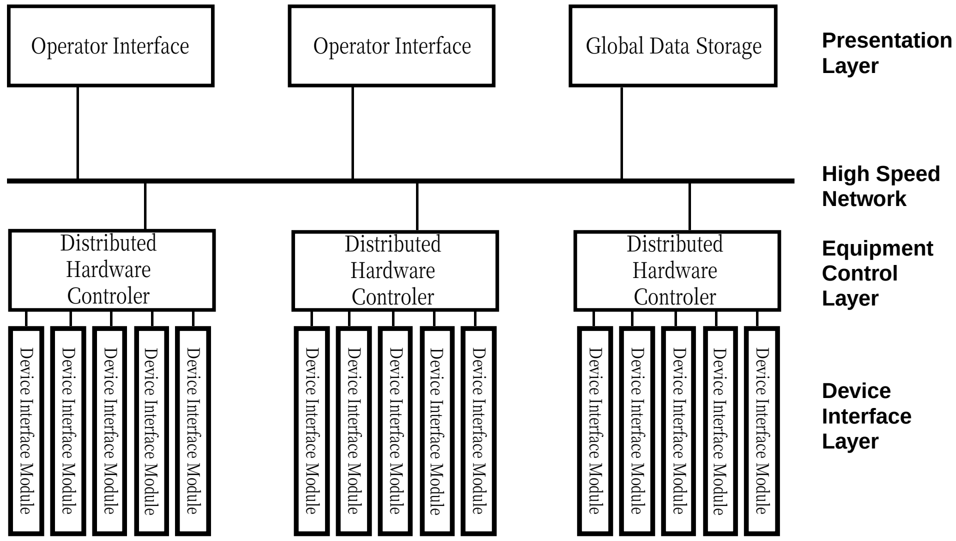

Figure 13.1: Schematic view of the KEKB accelerator control system.

13.2 System Architecture

The KEKB accelerator control system will be built based on the EPICS[2] toolbox. The control system is divided into three layers:

- Presentation layer,
- Equipment control layer, and
- Device interface layer,

as shown in Figure 13.1. The network system which connects the first two layers is another key component of the control system. In EPICS terminology, a computer in the presentation layer is called an OPI (Operator Interface) and one in the equipment control layer is called an IOC (Input/Output Controller).

13.2.1 Device Interface Layer

The device interface layer, which sits at the lowest level of the three, sends and receives signals from or to individual hardware equipment. This layer communicates such signals and data with computers in the equipment control layer. A schematic view of the device interface layer is shown in Figure 13.2.

KEK has a large inventory of CAMAC modules from the TRISTAN control system. The CAMAC crates and modules inside them are used as a device interface layer of the KEKB control system. The CAMAC crates are connected by CAMAC serial highways or CAMAC branch highways, depending on specific requirements. An equipment

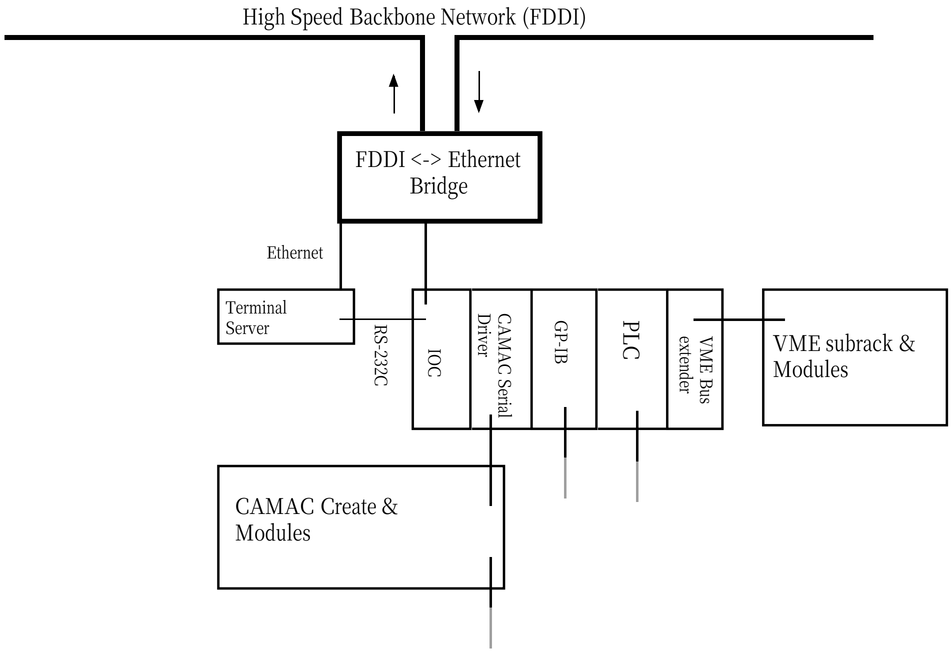

Figure 13.2: Schematic of the device interface layer

control computer in the VME sub-rack controls the serial/branch CAMAC highway driver in the same VME sub-rack .

The standard CAMAC modules used in TRISTAN are supported by the KEKB accelerator control group, i.e. the control group takes the responsibility for maintaining proper functionality of these modules. It is considered that standard CAMAC modules can cover most of the needs for the device interface modules.

However, CAMAC is not the only field bus that is used in the device interface layer for KEKB. Drivers of other field bus architecture, such as GP-IB, can be placed in a VME sub-rack . Another example of the field bus used in the KEKB accelerator control system is a multi-drop serial line, with which about 1600 power supplies for steering correction magnets are controlled. A device interface using a multi-drop serial line for magnet power supplies is now being studied in collaboration with the magnet group and the beam transport group for KEKB. The high-speed BPM electronics system will be implemented on VXI. The VXI is yet another example of the field bus to be used at KEKB. In this case, it will be natural to control these modules directly from IOC through the VME/VXI bus .

13.2.2 Equipment Control Layer

The equipment control layer consists of computers that functionally control the equipment in each hardware group, e.g. magnet, RF systems, beam monitor and others. The

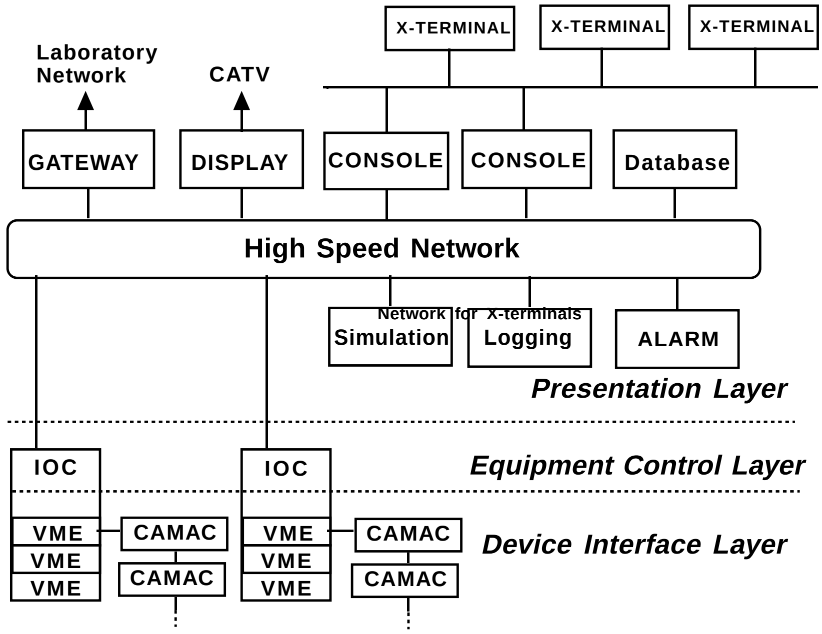

Figure 13.3: KEKB Accelerator Control System Architecture

equipment control computer, IOC, is a VME board-computer. The current version of EPICS system supports VxWorks as the real-time operating system for IOC.

One of the core elements in EPICS is the distributed runtime database, which resides in IOC. A database record (also called a process variable) in the runtime database corresponds to each channel in the device interface layer. Those records include all the device-specific information, such as hardware address, calibration table, upper/lower limits of action variables, and others.

Software components to support CAMAC are already included in EPICS. To simplify the use of standard CAMAC modules for KEKB, device support routines for these standard CAMAC modules will be developed.

Although EPICS supports a rich collection of VME modules, some modules that are introduced for KEKB may not be readily supported by EPICS at the moment. For these modules, it is necessary to develop a set of software, device driver, device support and record support. This task will be carried out by personnel from the control group and the hardware group which uses such modules.

13.2.3 Network

The equipment control layer and the presentation layer are connected through a high-speed network, such as FDDI. To separate the network traffic between these layers from the traffic between console computers and X-terminals, another network and/or

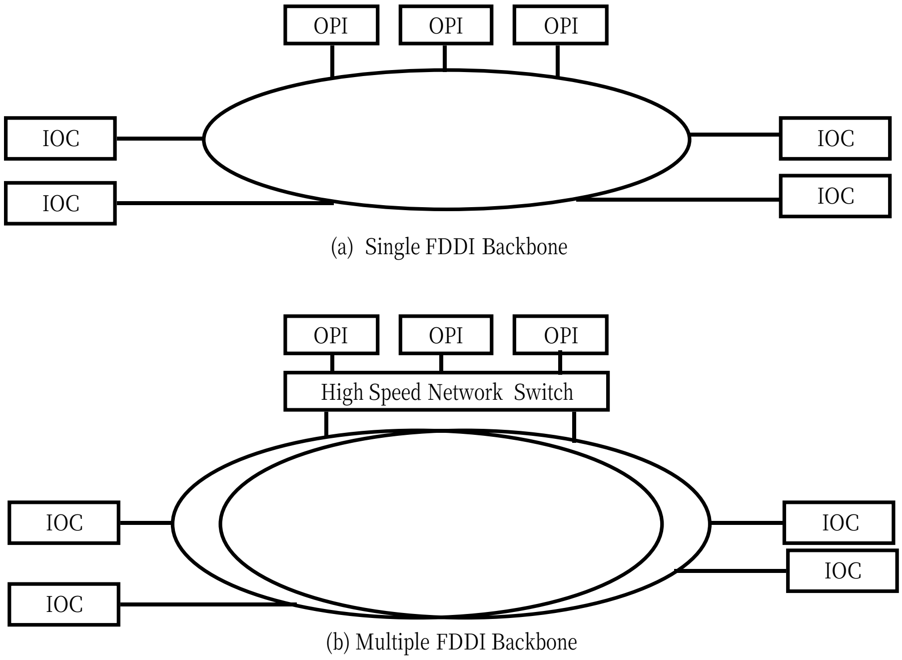

Figure 13.4: Throughput of the network will be enhanced by duplicating the FDDI backbone.

a network switch will be used, as shown in Figure 13.3.

In EPICS, the OPI and IOC communicate through a protocol called "Channel Access" (CA). The CA protocol hides the detail of physical configuration of lower layers from the application program on OPI. The CA protocol is built on the standard TCP/IP network protocol.

If the network system is not scalable as other parts of the system, the network can become a bottleneck when the system grows. When the backbone FDDI is saturated by the network traffic, a high-speed network switch and multi FDDI backbones will be introduced, as shown in Figure 13.4. It enhances the network throughput up to the speed of network switch, which is currently $\sim$ 1 GBit/sec.

There is also the possibility of adopting a distributed shared-memory network to obtain fast data transfer and event transmission among computers. The high-speed orbit feedback is a candidate to utilize this technology.

13.2.4 Presentation Layer

The presentation layer includes the operator's consoles, database manager, simulation computer, alarm generation/recording, data logging, displays, and a gateway to the KEK local-area network. Workstations and/or CPU servers running UNIX[1] operating

[1]UNIX is a trademark of Unix Lab.

systems are used in this layer. The X-window system is used as the graphical user interface standard.

The software environment used in this layer is very important for the whole system. The fundamental functions that have to be supported from the beginning of the installation include: the controls of each hardware equipment and graphical displays of the hardware status and various monitor values from the accelerator. Programmer-friendly interfaces must be provided for the accelerator physicists to develop application programs painlessly. It is also important to have very efficient data retrieval tools and analysis tools. The use of EPICS solves some of these problems as discussed later.

Operator's console

X-terminals with multiple displays will be used as operator consoles. The X-terminals are connected to the UNIX machines in the presentation layer. By using the tools in EPICS, personnel who are not from the control group will be able to create graphical operator interfaces without difficulties. Tcl/Tk and other languages which support CA access can also be used to develop application programs.

Database manager

The database manager keeps all information concerning the KEKB accelerator complex, such as machine parameters, equipment specifications, component locations, and others. Logged data, archived data and configuration data should also be accessible from a database system with the same interface. Commercially available DBMS(Database Management System) will be used as a framework of the database manager. Two commercial relational database management software, ORACLE[2] and SYBASE[3], are currently being evaluated. Recent progress in OODB(Object Oriented Database) technology should be also taken into account for the choice of DBMS.

Each runtime database record in EPICS has a unique name in the system. This unique name may not be easy enough to decipher for application programmers. The database manager also works as a channel name server. An application program can send a query on channel names and get a list of the channel names from the database manager.

[2]ORACLE is a registered trademark of ORACLE Corporation.

[3]SYBASE is a registered trademark of SYBASE,Inc.

Simulation computer

The simulation computer is used for accelerator physics calculations for such purposes as orbit correction and optical matching optimization. Computer code SAD[4] has been heavily used in the design work of KEKB, and will be used as a modeling/simulation program in the KEKB control system as well.

The ultimate goal of the development of SAD is to build a virtual accelerator on a digital computer. The KEKB control system will be built such that the SAD users should be able to control this virtual accelerator through the *same* operator interface that is used for the real accelerator, like a "flight simulator." To realize this idea, an interface program between SAD and a portable CA server will be developed, which will run on UNIX workstations. The CA clients can access the virtual accelerator through this CA server using an ordinary CA protocol.

The concept of a virtual accelerator is also useful in the development of application programs. Application programmers can test their software without risking operation of the real machine. It is even possible to develop an application program without a real machine.

System Alarm/Data logging

Equipment failures are monitored by each IOC and are reported to the alarm computer for broadcasting and recording purposes. The alarm computer notifies such conditions to the operator with various methods including: alarm sound, flashing or pop-up display, automatic pocket-beeper calls, and so on. The data logging computer collects and saves data from IOCs for later analyses.

System status broadcast and the local network gateway

The display computer broadcasts and displays data and information throughout the KEK site through the CATV network and other media. There is also a gateway computer which connects the KEKB control computer network with the KEK laboratory network. Through the KEK lab network, users who are away from the control room can monitor the status of each device and the status of KEKB overall. Privileged users are allowed to control their equipment from their office. To avoid a network jam caused by access through the laboratory network, a proxy channel access (CA) server will be used on the gateway computer. It will intercept and bundle CA requests from the laboratory network.

13.2.5 Timing system

Magnet synchronization

Since the magnet setting in the KEKB rings needs to be changed without loosing the beam, the control system has to change the strength of the magnets step by step in a synchronized way. For this operation, a timing system will be used to send a "heart beat" to the entire system. A dedicated signal line will be used to send a timing signal. A module in a device interface layer receives this signal and triggers an action.

Synchronized data acquisition

The beam orbit in the beam transport line are measured by using one-pass beam position monitors (BPMs). The KEKB control system needs to be able to associate this data with data from the injector linac BPMs or from the beam loss monitors and others within KEKB. This is important for studying and optimizing the beam injection setup.

When orbit distortions are intentionally introduced at KEKB for studying magnet misalignment and other effects, the BPM data need to be tagged appropriately such that sensible analyses are made possible.

Bibliography

[1] W-D. Klotz et al., "Experience with a *Standard Model '91* based Control System at the ESRF", ICALEPCS '93, Berlin(FRG), 1993.

[2] L.R. Dalesio, et al., "EPICS Architecture", ICALEPCS 91, KEK Proceedings 92-15, (1992) pp.278-282., Dalesio, L., et al. "The Experimental Physics and Industrial Control System Architecture," ICALEPCS, Berlin,Germany, Oct. 18-22, 1993.

[3] "Vsystem" is a product of Vista Control Systems Inc., Los Alamos, NM.

[4] SAD is the accelerator design tool program developed at KEK.

[5] Shin-ichi Kurokawa, et al., "The TRISTAN Control System", Nucl. Instr. and Meth., A247, (1986) pp. 29-36.

[6] Shin-ichi Kurokawa, " Status of TRISTAN-II Project", Proceedings of the 1993 Particle Accelerator Conf, pp. 2004-2006, S-I Kurokawa, "KEKB status and Plans", 1995 Particle Accelerator Conference, Dallas, Texas.