Available online at www.sciencedirect.com

SCIENCE DIRECT®

Nuclear Instruments and Methods in Physics Research A 499 (2003) 138–166

NUCLEAR INSTRUMENTS & METHODS IN PHYSICS RESEARCH
Section A

www.elsevier.com/locate/nima

KEKB accelerator control system

Nobumasa Akasaka, Atsuyoshi Akiyama, Sakae Araki, Kazuro Furukawa, Tadahiko Katoh*, Takashi Kawamoto, Ichitaka Komada, Kikuo Kudo, Takashi Naito, Tatsuro Nakamura, Jun-ichi Odagiri, Yukiyoshi Ohnishi, Masayuki Sato, Masaaki Suetake, Shigeru Takeda, Yasunori Takeuchi, Noboru Yamamoto, Masakazu Yoshioka, Eji Kikutani

KEK, High Energy Accelerator Research Organization, Oho 1-1, Tsukuba Ibaraki 305-0801, Japan

Abstract

The KEKB accelerator control system including a control computer system, a timing distribution system, and a safety control system are described. KEKB accelerators were installed in the same tunnel where the TRISTAN accelerator was. There were some constraints due to the reused equipment. The control system is based on Experimental Physics and Industrial Control System (EPICS). In order to reduce the cost and labor for constructing the KEKB control system, as many CAMAC modules as possible are used again. The guiding principles of the KEKB control computer system are as follows: use EPICS as the controls environment, provide a two-language system for developing application programs, use VMEbus as frontend computers as a consequence of EPICS, use standard buses, such as CAMAC, GPIB, VXIbus, ARCNET, RS-232 as field buses and use ergonomic equipment for operators and scientists. On the software side, interpretive Python and SAD languages are used for coding application programs. The purpose of the radiation safety system is to protect personnel from radiation hazards. It consists of an access control system and a beam interlock system. The access control system protects people from strong radiation inside the accelerator tunnel due to an intense beam, by controlling access to the beamline area. On the other hand, the beam interlock system prevents people from radiation exposure by interlocking the beam operation. For the convenience of accelerator operation and access control, the region covered by the safety system is divided into three major access control areas: the KEKB area, the PF-AR area, and the beam-transport (BT) area. The KEKB control system required a new timing system to match a low longitudinal acceptance due to a low-alpha machine. This timing system is based on a frequency divider/multiply technique and a digital delay technique. The RF frequency of the KEKB rings and that of the injector Linac are locked with a common divisor frequency. The common divisor frequency determines the injection timing. The RF bucket selection system is also described.

PACS: 29.20

Keywords: Accelerator control; Control computer system; EPICS; Safety control system; Timing system

---

*Corresponding author.

E-mail address: tadahiko.katoh@kek.jp (T. Katoh).

0168-9002/03/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.
PII: S0168-9002(02)01786-2

Overview

The KEKB accelerator control system [1–6] consists of:

(1) KEKB accelerator control computer system,
(2) KEKB timing distribution system [7,8],
(3) KEKB safety control system, and
(4) KEKB communication system. The TRISTAN communication system [9] is used as the KEKB communication system without any significant changes.

The system design started in 1993. During the years 1993 and 1994 we have designed system and evaluated several existing control computer software environments, such as the EPICS [10–12], Vsystem [13] and NODAL [14]. We have concluded that EPICS is a feasible and cost-effective solution for the KEKB control system. The advisory committee in the KEKB accelerator department also supported this conclusion. The first portion of the control computer system, including the server workstation, the network, and 15 VMEbus-based computers were installed in March 1997. The full system installation finished in March 1998 and the KEKB accelerator commissioning was completed by the end of that year.

While the KEKB accelerators were being mounted in the tunnel where TRISTAN Main Ring (MR) was installed, there was a control computer system for TRISTAN distributed in the local control rooms. We had many CAMAC modules connected to various TRISTAN equipment. In order to reduce the cost and labor for constructing the KEKB control system, we decided to reuse as many CAMAC modules as possible. For RF and vacuum monitoring and controls, we are now using CAMAC modules after overhauling them. However, we had changed electrolytic capacitors in the power supplies of CAMAC crates, and also changed all of the cooling fans mounted in the chases and racks. Some groups, such as the Beam Monitor, Magnet, and Beam Transport groups, decided to use a completely new configuration and new equipment for monitoring and controlling their hardware.

The guiding principles of the KEKB control computer system are as follows:

(1) use EPICS as the controls environment,
(2) provide a two-language system for developing application programs,
(3) use VMEbus as frontend computers as a consequence of EPICS,
(4) use standard buses, such as CAMAC, GPIB, VXIbus, ARCNET, RS-232 as field buses, and
(5) use ergonomic equipment for operators and scientists.

On the software side, we adopted interpretive languages, Python [15,16] and SAD [17], for writing application programs. They are suitable not only for making programs for operations but also for accelerator studies. By their nature of interpretive languages, we can easily obtain higher reliability and flexibilities in modifications, ease of syntax checking, and short turn-around times.

The whole system was developed by close relationships with link-persons. The link-persons are interface people in the hardware groups who are responsible for controlling the equipment of their groups. For software design and programming, we have hired 4–5 engineers from a company outside of KEK. They are very powerful in designing, coding, and making documents concerning the software. Accelerator operations are also performed by out-sourcing.

1. System design

1.1. Functional requirements

The control-system design must be looked at from as many points of view as possible. Usually, the suppliers of the system and the users tend to have different points of view. Therefore, we first started to review the functional requirements of the control system for the KEKB accelerators. The hardware groups that install the accelerator components to be controlled want to monitor the magnet current values, the voltages of the power supplies and so forth, and if a failure occurs they want it to be reported and recorded as soon as possible. The operations group wants to record the operations sequence of the accelerators and to have tools to analyze previous operations and to make correlations between the parameters and the measured values for tuning purposes. Accelerator physicists need connections between the beam-monitoring, simulation and beam-correction systems.

The principal functional requirements are as follows:

1. All the data that can be taken should be taken.
2. All the data taken should be saved for later analysis.
3. All the operations should be recorded for later inspection.
4. All the machine parameters and data about the machine components should be saved in the database.
5. The control system should be operator friendly.
6. The programming system for application programs should be programmer friendly.
7. The overall response time to an operator's request should be less than a few seconds, ideally 1 s, unless progress of the process is indicated.

1.2. Constraints

The KEKB accelerators are reconstructed machines; there were many constraints in constructing the control system economically and efficiently. The first of them was CAMAC. We had already accumulated a large number of CAMAC modules in the TRISTAN control system, and CAMAC was still thought to be the only well-defined standard for interface modules that could be used for process input/output. A few new CAMAC modules were added. On the other hand, it was necessary to replace mini computers more than 10 years old with the latest technology. There was no constraint to introducing a new software environment, because all of the application programs had to be newly developed.

Because the final commissioning of KEKB was scheduled to take place in 1998, the control system was installed about 1 year before that. The basic system for hardware and software development was installed at the beginning of 1997. It included a workstation, VME computers and X-terminals as operator consoles. Another constraint was that there were only a dozen people in the KEKB controls group.

1.3. Basic design concepts

Considering the requirements and the constraints stated above, the basic guidelines are to use:

1. the so-called Standard Model for the accelerator control system,
2. international standards for the interfaces between the three layers, to facilitate later upgrading and maintenance,
3. international standards such as CAMAC, VME, VXI and GPIB as the interfaces between the control system and the equipment to be controlled,
4. products either from international collaborations or that are commercially available, to minimize the manpower and effort required,
5. the object-oriented technique or abstraction to hide the hardware from application programmers,
6. high-speed networks to connect the computers with each other to get a quick response,
7. and the link-persons system, as in the construction of the TRISTAN control system, in which the link-persons make the equipment database and code device drivers for the application programs because they know the equipment best.

Finally, we decided to use EPICS as the KEKB control system environment, and joined the EPICS collaboration.

1.4. System architecture

The control computer system is divided into three layers—presentation layer, equipment control layer and device interface layer—as shown in Fig. 1. The first two layers are divided functionally, but are connected with each other through an

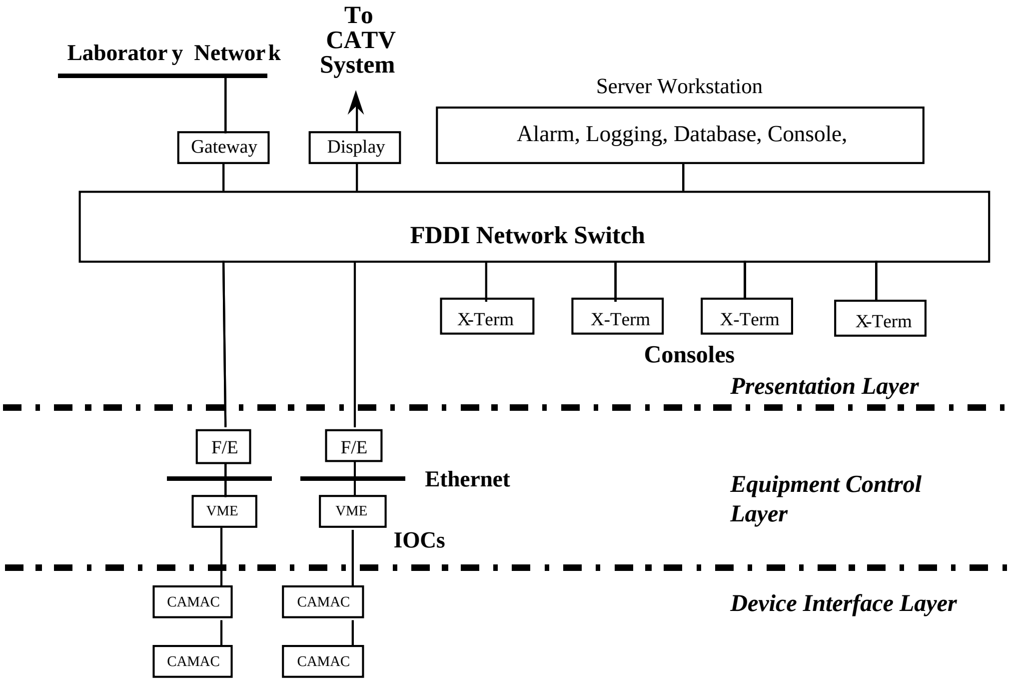

Fig. 1. System configuration.

FDDI switched network. This is because the network traffic between the presentation layer and the equipment control layer computers is expected to be dominant. There is also a possibility of adopting a distributed shared-memory network to obtain fast data transfer, and event transmission among the computers. The presentation layer which is called an Operator Interface (OPI) in EPICS is composed of several workstations and about 20 X-terminals used as operator consoles. The main network is an FDDI-switched network that has a node at each local control building, where an Ethernet segment is extended. The equipment control layer consists of about 100 VME Input/Output Controllers (IOC) distributed around the KEKB rings. Each IOC is allocated to a hardware group so as to avoid any conflicts among groups or users. The final layer, the device interface layer, has standard modules, such as CAMAC, GPIB and VXI. There is a special interface for controlling the magnet power supplies.

1.4.1. Presentation layer

OPI, the presentation layer, includes operators consoles, a database manager, a simulation computer, alarm generation/recording, a data logging, display and a gateway to the KEK in-site network. All of the functions are performed by the server workstation. A relational database management system, ORACLE 7, runs on the server workstation and keeps all information concerning KEKB, including the machine parameters, equipment specifications and location. The simulation is performed by the accelerator design workstation for such purposes as orbit correction. Faults in the equipment are monitored by each IOC and reported to the server for broadcasting and recording purposes. The data logging programs collects data from various IOCs for later analyses. Some display output from the server programs are converted to the standard TV signal format and transmitted over the KEKB site through the CATV network and other media. Most of the X-terminals are PCs running X-server emulation software. Macintoshes and Microsoft Windows PCs are used for this purpose (Table 1).

1.4.2. *Equipment control layer*

The equipment control layer consists of IOCs that functionally control the equipment of each hardware group. Each IOC is a VME computer equipped with CAMAC serial highway drivers and other standard field-bus driver modules. The operating system for the equipment control computer is VxWorks.[1] Several types of processors on the VME processor boards are used. MC68040, MC68060, PowerPC603 and PowerPC750 are used according to the requirements. The software programs for these IOCs were cross-developed and generated on an EPICS server workstation.

1.4.3. *Device interface layer*

There are several field buses for the lowest device interface layer. There are CAMAC crates and CAMAC modules which are re-used from TRISTAN. The CAMAC crates are connected by CAMAC serial highways. There are also other standard field-bus equipment, such as GPIB. VXI modules are used for beam-position monitors and fast signal measuring systems, like beam-feedback systems. There are about 800 BPMs, and the electronics for them are in 20 sub-control rooms around the ring, where the VXI system is used. More than 2200 magnet power supplies are installed in eight power-supply rooms around the ring. These power supplies are controlled through ARCNET, which was specially designed for them. A Power Supply Interface Controller Module (PSICM) is plugged into each power supply and exchanges signals and data with the power supply, and the VME IOC. Schematics of the power supply control configuration are shown in Fig. 2.

2. Control consoles

The operator consoles of the KEKB accelerators will be described in comparison with the former TRISTAN accelerator control consoles.

Table 1
Number of VME modules in the KEKB control system

	Linac	AR	Beam monitor			Beam transpt.	Control			Magnet			RF	Vacuum	Physics exp.	Total
			BPM	Feedback	SOR		Control	Timing	Operation	Magnet	Mover	QCS				
CPU/subrack	3	11	27	11	2	8	1	1	1	9	10	1	8	13	1	107
ARCNET	—	6	—	—	—	8	—	—	—	44	—	—	—	—	—	58
CAMAC	3	10	—	1	—	6	1	—	1	—	—	—	8	13	—	43
DIO	—	—	—	—	—	—	—	—	—	—	9	—	—	—	—	9
Event trans.	—	—	—	—	—	—	—	1	—	—	—	—	—	—	—	1
Event recvr.	—	—	20	—	—	—	—	1	—	—	—	—	—	—	—	21
GPIB	3	7	4	6	2	7	—	—	—	8	11	2	9	18	—	77
Modbus+	—	1	—	—	—	5	—	—	—	1	—	—	—	—	—	7
MXI	—	—	37	1	—	4	—	—	—	—	—	—	—	—	—	42
RAS	3	11	26	9	1	8	1	1	1	9	10	1	8	13	1	103
Total	12	46	114	28	5	46	3	4	3	71	40	4	33	57	2	468

---

[1] VxWorks is the name of a real-time operating system and a registered trade mark of Wind river Systems, Inc.

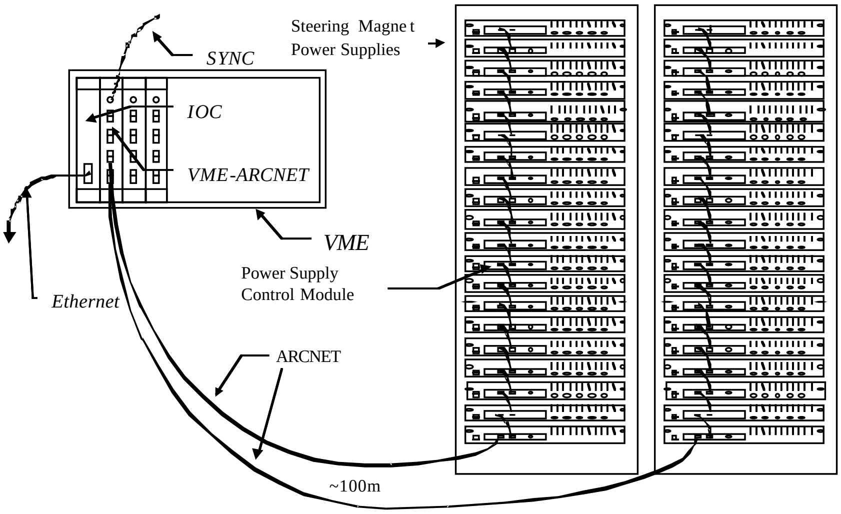

Fig. 2. System configuration of the steering magnet power supplies control.

The construction policy of the KEKB operators' console is flexibility [18,19]. For TRISTAN, there were several sets of identical console desks made of iron frames in which two touch-panels, two graphic display monitors, and ten TV monitors were packed. Each set of console was directly controlled by a mini computer in the TRISTAN control computer network. On the contrary, in the KEKB control system, all the man–machine interface devices are based on the X-window system and application software runs on a UNIX server workstation. Therefore, the number of X-window devices, such as X-terminals, Macintosh and PC/AT compatible PCs with X-server software, is limited only by the capacity of the UNIX server and space. Low-cost Network Stations [20] are introduced as X-terminals with a single screen. A Macintosh or a PC can have up to four multi-screens, and works as multiple X-terminals with only a set of a mouse and a keyboard. For space and ergonomic reasons [19], we adopted TFT flat-panel displays, which are thin and light in weight; they reduce the reflection of light. For the same reason, a cordless keyboard and a mouse were also introduced.

In the EPICS environment, the system is centralized as far as operation is concerned; X-terminals are used as man–machine interface devices. Application programmers using MEDM, SAD and Python run on the server workstation in the system, which output information on the X-terminals.

2.1. Console design concepts

In the KEKB control system, consoles are provided as flexibly as possible to reply to any requests from the KEKB accelerator commissioning team. Only several large tables are distributed in the central control room, as shown in Fig. 3. PC-based X-terminals are placed on large tables (4.0 m × 1.6 m) and the latest 15- and 18.1-in. Thin-Film Transistor (TFT) Liquid Crystal Display (LCD) monitors and keyboards are placed on them. Because these LCDs are light in weight and thin, they require only a little space. Wireless keyboard and mouse sets are also used partially, which makes the table clean and simple. Because of the nature of a new accelerator like KEKB, many people want to operate it and conduct studies during the commissioning period. All of them want to use their own X-terminals in the CCR. There are more than 30 general-purpose X-terminals, including Macintoshes and Windows PC running X-terminals.

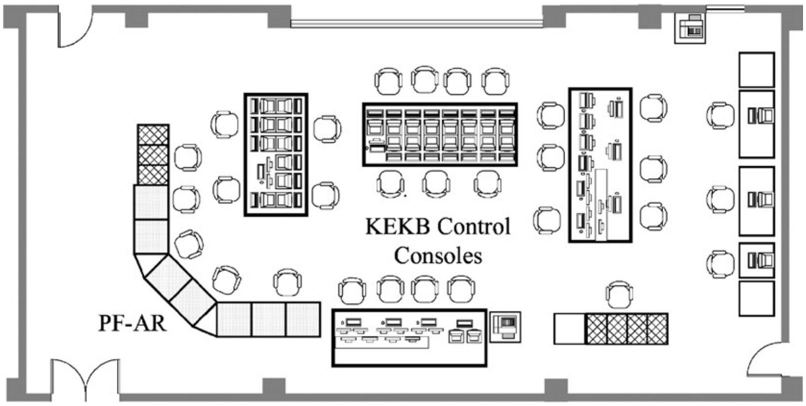

Fig. 3. Layout of the KEKB control consoles.

2.2. Ergonomic design

Previous TRISTAN consoles were made of iron-framed racks, and various devices were mounted in them. In order to keep the operators from any reflected images of the light sources on the round surface of the monitors, the lighting in the control room was limited in both strength and direction. It was so dark and the operator needed to have a lamp stand when he wrote the logbook. By using flat display monitors, like LCDs, the reflection problem could be eliminated and we could make the room as bright as in an office room. The brightness gives us benefits of easiness and comfort when reading and writing.

2.3. Console equipment

There are many PCs (IBM PC/AT compatibles and Macintoshes) used as X-terminals. Most of the Macintoshes are equipped with 2–4 display cards to support multiple screens. Multiple-video-display controllers can be installed into Windows PCs and one can use 2–4 screens. By using multiple-screens, you can display much more information than single-screen, and are not overlapped, but on separated windows. The most beneficial merit which we obtain is that the use of multiple-screens decreases the number of necessary keyboards and mice, thus allowing more space for logbooks, etc. Low-cost X-terminals are realized by using a Network Station. It is a thin PC, which allows emulation software of various protocols, such as X-Window and IBM 3270, can be downloaded and operated. A Network Station has no disk drive or fan, and is completely maintenance free. Various types of TFT color LCD monitors are used. An 18.1-in. monitor displays an SXGA (1280×1024 pixels) screen, and a 15-in. monitor displays an XGA (1024×768 pixels) screen. There are NTSC color TV monitors as used to display the usual TV signals. The latest LCD monitors have a characteristic of wide viewing angles of more than 120°.

For IBM PC/AT compatibles, we adopted a PCI-bus graphic display controller board. It can display 1–4 SXGA screens at one time, and it is possible to have a Windows screen with 2560×2048 pixels with true color. For a Macintosh, conventional PCI graphic cards can be added to obtain more screens. Because we have many keyboards and mice along with connecting cables, they are sometimes tangled or tied together. We adopted sets comprising a wireless keyboard and mouse for IBM PC/AT compatibles. For a Macintosh, a USB-to-PS/2 converter can be employed to utilize the same keyboard and mouse set. For the common display use in the central control room, we had been using 27-in. color TV monitors for years. Some of them were damaged as well as obsolete. Therefore, we replaced them by 40-in. plasma display monitors. A plasma display monitor is thin (about 15 cm thick) and light (about 30 kg) compared to a CRT display of 27 in. (about 60 cm deep and 50 kg in weight).

3. Computers and network

3.1. System configuration: client/server architecture

IOCs located in the local control rooms collect the raw data. They, as a data server, also reply to requests mainly from the client programs on the server workstations in the central control room. Operations and data analyses are performed by these server workstations; the outputs are displayed on the screens of the X-terminals.

3.2. Network

In the KEKB control computer system, an FDDI switched network is used to connect the IOCs at local control rooms to the central control room. The distance between the local control rooms and the server workstations in the central control room is from 50 to 2050 m. Multi-mode optical fiber cables are used for connections of less than 2 km, and single-mode ones are used for more than 2 km. In the local control rooms, there are 100 base TX or 10 base T Ethernet segments. At each local control room, there is a terminal server for serial ports of the IOCs.

3.3. UNIX server workstations

In the KEKB control computer system, there is a UNIX server workstation of PA-RISC architecture with 4 CPUs. The clock frequency of the server is 100 MHz and the main memory capacity is 2 GB. The system hard disk has a capacity of 4 GB and an external RAID disk has a 20 GB capacity.[2] A server workstation for accelerator simulations is also used for accelerator operations, communicating with the EPICS server. During the year of 2000, another server workstation was introduced as an extension of the control system to cover the Photon Factory Advanced Ring (PF-AR) storage ring as shown in Fig. 4. This server has two 440 MHz PA-8500 processors, 1 GB of memory and a 36 GB system disk. Two server workstations work together to control KEKB and PF-AR simultaneously.

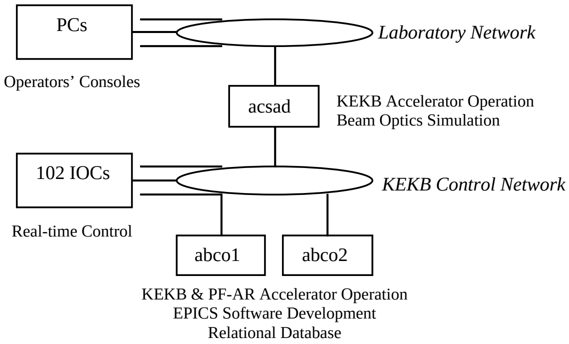

Fig. 4. New system configuration of server workstations.

3.4. VME IOCs

There are about 100 IOCs distributed around the rings in the local control rooms. Each IOC is equipped with a VME subrack, a plug-in power-supply module, a plug-in cooling fan unit, a system monitoring module and a VMEbus board computer module. Two CPU architectures are used in this system, namely the MC68k and Power PC architectures. Old MC68040 CPU and MC68060 CPU boards are still used and provide sufficient computing power for some applications. New Power PC modules of PPC 603e and PPC 750 are used for more computing power-consuming applications. The console port of the CPU module is connected to the terminal server and can be accessed through the KEKB control network.

The IOCs were carefully designed to realize high maintainability. The electric power for the VME subrack is supplied by a 250 W power supply module which is plugged-in from the front side. The system monitoring module monitors the power-supply voltages on the backplane, and sends alarm messages through a dedicated serial communication line connected to the terminal server. This module can generate a system reset

---

[2] Recently, we also added 9 GB internal disk.

signal on the backplane for re-booting the IOC in the VME subrack upon a request to the re-boot command from the serial communication line. This function is very effective and helps the link-persons to develop and test new software or EPICS database. The fan unit seems to be the weakest point of the IOC system. This is the only component that has moving mechanical parts. The rotating speed of the fan is controlled by the temperature of the air coming out through slots. Further, the fan unit can be removed and exchanged with a new one from the front side in a few seconds.

4. Field buses

The VME computers provide field buses. They are used to obtain more suitable interfaces to the equipment. In other words, the VMEbus is used only for providing field buses. The VMEbus is designed to transfer digital signals between the CPU and peripheral modules, and is not suitable for handling high-precision analog signals. We decided to use field buses for isolation, both functionally and electrically.

The field buses that we have adopted are: a CAMAC serial highway, ARCNET, GPIB, Modbus Plus, and RS232. We had many CAMAC Modules used for TRISTAN accelerators. RF acceleration system uses almost all the CAMAC modules used for TRISTAN RF control system. Some parts of the beam-transport control system and vacuum-control system also use CAMAC modules. ARCNET is used mainly to control the magnet power supplies. GPIB is used mainly for magnet current measurements and to control the vacuum equipment. Modbus Plus is used to monitor the status of the safety control system. MXIbus is used for interfacing VME IOC with VXI modules.

4.1. CAMAC system

As mentioned above, there are more than 2000 CAMAC modules still being used mainly for controlling the RF and vacuum equipment. High-power RF sources, themselves, are reused and the control system remains with few modifications. Therefore, CAMAC modules are also reused after overhauling the CAMAC crates and racks. Cooling fans in the crates and racks were all replaced by new ones. The electrolytic capacitors used in the power-supply units of the CAMAC creates were all replaced by new ones to prevent future failures.

A CAMAC serial highway runs within a local building to obtain a simpler configuration and good response time. The data-transmission rate is 2.5 Mbps in the bit serial mode.

4.2. VXIbus

The beam monitor group developed VXI modules for measuring about 800 beam positions, and 6 VXI main-frames are used in each of 20 local control rooms. Each position monitor has 4 electrodes and connected to a 4-input multiplexer VXI module, of which the output is cascaded to the next stage multiplexer module; a DSP module is used to detect the beam position.

4.3. ARCNET

ARCNET is used to control the magnet power supplies. The ARCNET controller chip has a very nice feature, that the chip, itself, manages the network configuration automatically and dynamically; namely, it detects any errors caused by removing a node or incomplete communication and modifies the network configuration by sending a reconfiguration packet to other nodes. After reconfiguration, the network communication is recovered without the node that caused an error. Also, if a new node is added to the network, the newcomer sends a packet, which is detected by other node, which sends a reconfiguration packet to the members, and the network is configured dynamically. It gives us live-insertion/deletion of ARCNET nodes.

4.4. GPIB

GPIB is used to connect measuring instruments to the KEKB control system. There are digital voltmeters used to measure magnet currents. For some remotely located equipments, LAN-GPIB interface controllers are used by connecting to an Ethernet.

4.5. RS-232

Some devices are equipped with RS-232 serial communication line interfaces. For them, we provide RS-232D interfaces from the VME IOCs. An example is the mass-spectrum analyzers in the vacuum control sub-system. RS-232 is also used to communicate with the PLCs. Device/driver support routines for these PLCs have been developed based on Ascii/serial driver/device support routines developed by Jeff Hill and Allan Honey for the KECK Observatory.

4.6. Modbus Plus

Several kinds of PLC devices are used in the KEKB control system. Among these, the PLCs which talk the Modbus Plus protocol are used in the magnet protection systems and the radiation safety system. While these systems are managed by the PLCs locally and autonomously, the interlock status should be monitored on the console in the central control room. A set of software has been developed in order to interface these PLCs to EPICS. It has the standard structure required for the device access layers of EPICS. It consists of a device support and a driver support. The device support deals with the processing specific to the Modbus Plus protocol. The driver support [21] interfaces directly with the hardware and handles physical I/O requests.

5. Software system

The KEKB control system adopted EPICS as a basis of control software. EPICS provides basic functionality, including periodic scanners, event scanners and the network protocol called Channel Access (CA). It also provides CA client applications, such as an application to monitor any alarm-status change, an application to record control data into permanent storage for later analysis and applications to construct operator interface displays.

These EPICS tools, or CA client software, are configurable through specific configuration files. In many cases in the applications of EPICS, standard EPICS tools can satisfy needs of the system. However, once the requirements exceed the ability of these tools, we need to find some way to extend the capability of these tools. These are mostly written in compiler languages, such as C and/or C++. However, it does not prevent us from extending these tools; the extending process is in general time consuming. In an experimental facility like an accelerator for high-energy physics, quick application development is required. To solve this dilemma, we introduced the use of interpretive languages in the KEKB control system. We discuss the details in Section 5.4.

5.1. EPICS environment

The first version of EPICS installed in the computer of the KEKB control group for evaluation was EPICS release 3.12.beta11. Currently, the KEKB control system uses EPICS release 3.13.2 with minor modifications for the KEKB control system; we plan to change 3.13.5 in the near future. Because EPICS 3.12beta11 just supported the Motorola 68K family CPU board, we had to port EPICS 3.12beta11 and later versions to a CPU board based on a PowerPC, which is widely used in the KEKB control system. Most of the code in EPICS can be used for both 68K CPUs and PPC CPUs without any modification; however, we have to resolve any problems caused by the architecture and compiler differences [22]. Concurrent Version System (CVS) [23] is used to keep track of any modifications to the software.

Device drivers for Modbus Plus [21] and ARCNET [24] were written to support these field buses in the KEKB control system. CAMAC modules are controlled by using device drivers and device support routines in the EPICS distribution, except an ADC module in KEKB. This ADC module, designed and used for the TRISTAN accelerator control, uses MSB of 12-bit long data as a sign bit. We therefore needed to develop a device support routine for this CAMAC module.

5.2. EPICS database

CapFast is used as a graphical tool to design the EPICS database. For a simple database, a text editor, such as EMACS, is also used. A large part of the EPICS database down loaded onto IOCs is generated from the dbLoadTemplate command using a template database designed by CapFast, and parameter files generated from an ORACLE database (see Section 5.3) (Fig. 5).

The EPICS runtime database in the KEKB control system has more than 270,000 records in total. It uses 51 record types, including 15 record types developed for the KEKB control system. Analog Input (AI) and Analog Output (AO) records are dominant record types, which occupy about half of the records population.

5.3. Relational database

We are using relational databases to manage the data used in the accelerator control system [25], such as hardware addresses. The KEKB database is used to generate the configuration files for the EPICS runtime database. The logical design of the KEKB database is based on Entity-Relation

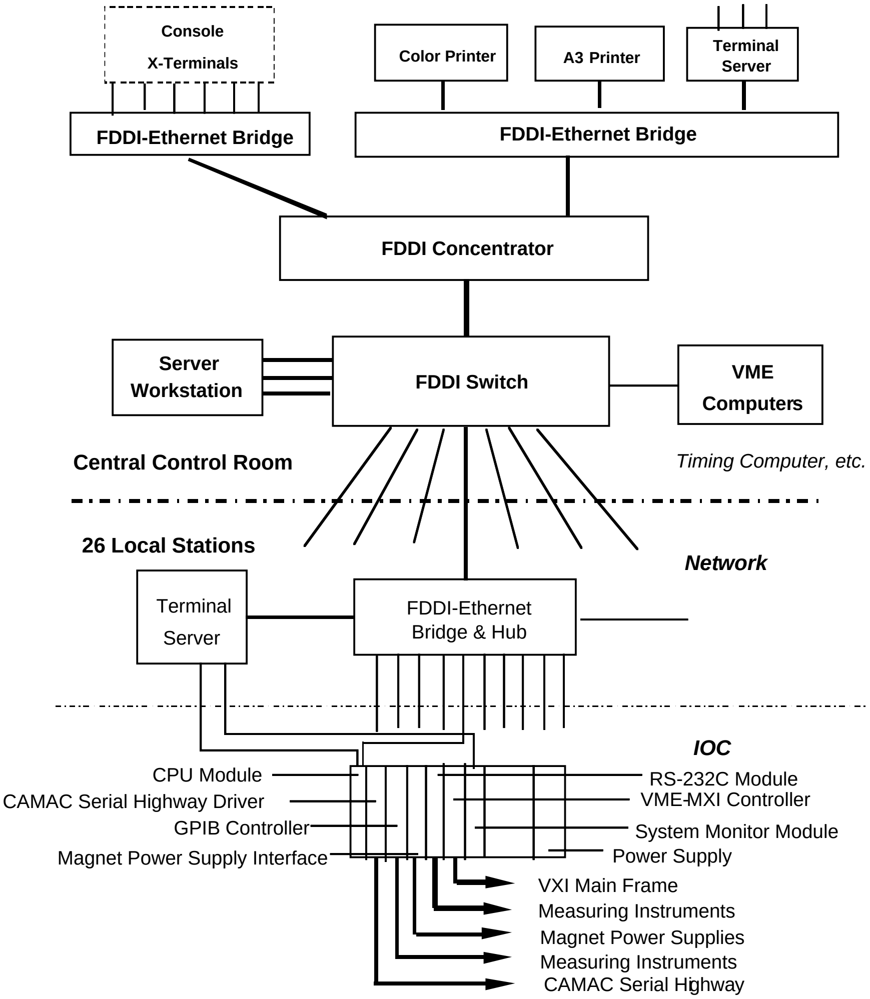

Fig. 5. Schematic configuration of the KEKB accelerator control computer system.

(ER) diagrams. We use ORACLE as a relational database manager. An ER diagram is converted to an ORACLE database using a commercial tool. The device information database for the KEKB accelerator control system is created by this method.

5.3.1. Automatic generation of EPICS files

The KEKB database system automatically generates several types of configuration files used in EPICS system, as listed below:

(1) EPICS Database.
(2) MEDM File (*.adl).

An EPICS database is a file to be downloaded into each IOC (IOC: Input/Output Controller, a VME computer), which contains the configuration of the runtime database on the IOC. Motif Based Display Manager (MEDM) is one of the client tools of the EPICS system, which displays operation panels, menus and other objects on a display window. The configuration file of MEDM is named "*.adl" Each configuration file type has its own generator in the KEKB database. These generators are written in SQL*PLUS language.

5.3.2. EPICS database generation

We expect over 100,000 EPICS records in the KEKB control system. It is not feasible to fill out every field of these records by hand. Some automation is necessary. We describe EPICS template records using CAD software "Capfast" (Fig. 6). Template files may include EPICS macro names. When this template database is downloaded onto IOC, a macro name is substituted by an actual value. The KEKB database provides a pair of the EPICS macro name and the EPICS macro value. "dbLoadTemplate" (EPICS tool) is used for downloading a database.

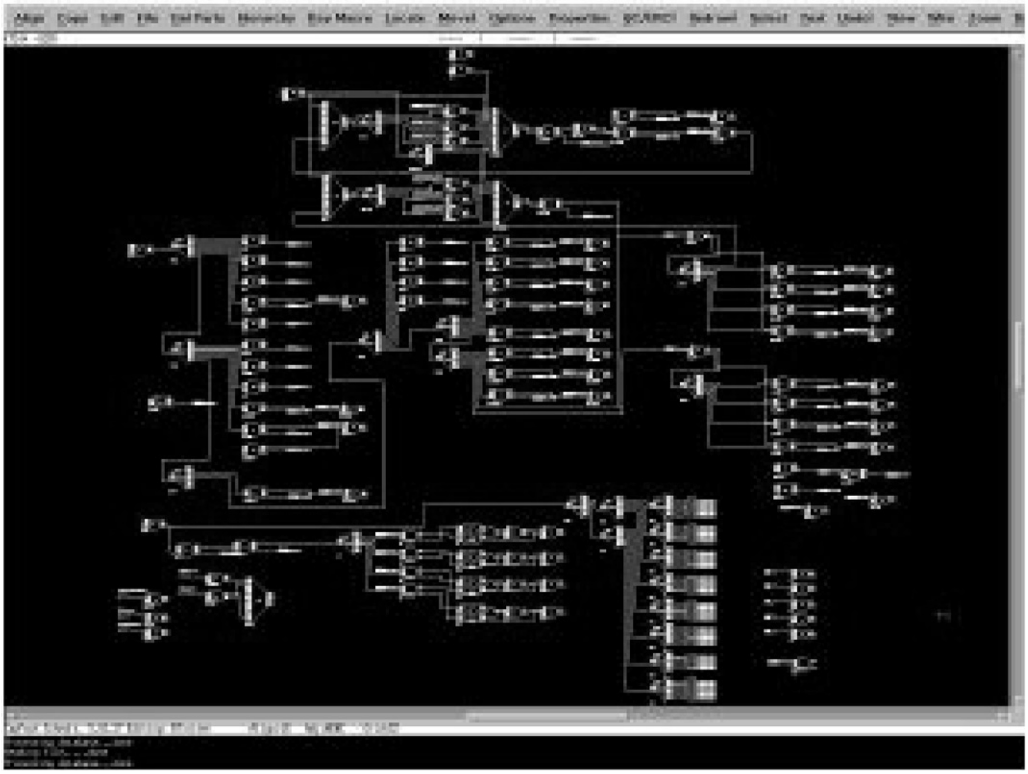

Fig. 6. An example EPICS database as CAPFAST schematic.

5.3.3. MEDM files generation

To generate MEDM files, we use EPICS parameter files as intermediate files. We make MEDM template files, which include EPICS macros, using MEDM as a graphical panel editor. Finally, we generate a configuration file for control windows by some tools described in UNIX shell script or "awk" language.

5.4. Interpretive languages

5.4.1. Two-languages system

In the control system for a high-energy accelerator, like KEKB, quick application development/modification is required. A short turn-around time is especially important during commissioning of an accelerator. MEDM and/or EDD/DM (EPICS Display Editor/Display Manager) are useful development tools for such a purpose. A user of MEDM can build a graphical user interface screen without any conventional programming. However, some high-level applications may not fit in this framework. In the KEKB control system, we have achieved this goal by introducing interpretive programming languages, Python and SAD, into the control system. SAD is a language originally developed at KEK for accelerator lattice design. The other, Python, is a general-purpose language system distributed as public-domain software. These languages are used not only for prototyping applications, but also for developing application software used in daily operation [26,27]. These languages are easier to learn and safer to use compared to compiled languages, such as C or C++. The interface to the appropriate widget library from these interpretive languages, such as Tk/gtk+ widget, greatly reduces the effort needed to develop graphical user interfaces. Modular and object-oriented features in the languages allow an incremental development of application software [28,29]. This method benefits the reliability and maintainability of the software.

The TRISTAN control system uses NODAL [14] and Process Control Language (PCL) [30] for software development. NODAL is an interpretive language with Basic-like syntax and features specific to a distributed control system. PCL is a compiled language with Fortran-like syntax and extensions for a real-time system. PCL is used mainly to develop data modules in Nodal to access hardware. Most of the user interfaces and applications in the TRISTAN control system were written in NODAL mainly by accelerator physicists. This kind of "two-language" system is widely seen in successful computer systems [31]. Unix, for example, has shell scripting languages and C. Elisp and C in the EMACS is another good example. This observation suggests that a successful user extendable system has (at least) two programming languages, an interpretive language and a compiled language.

In a two-language system, a compiled language is used to extend the capability of the interpreted language. This extension is modular so that users can add or delete this extension at anytime. The interpreted languages are used as glue to combine these modules. This approach is also useful to avoid a fat application which includes everything (Fat software). Modular design of software also makes development/test/maintenance easier.

Another advantage of a "two-language" system is the participation of users. An interpreted language usually has simpler syntax and is easy to learn and use for people, including NON-professional programmers. In KEKB, application programs which require knowledge of accelerator physics are written by accelerator physicists. In this case, there is a large overlap in users of an application and its developer. This reduces overhead of communication between the user and the developer. They use mostly SAD [17] language because they use it for their research anyway. Python is used to develop an application where an accelerator model is not required. Hardware engineers and accelerator operators can develop there own applications to simplify their own daily tasks. The clean and simple syntax of Python makes learning it easier. Although there are several textbooks on Python in English, only a couple of these [15] are available in Japanese. These textbooks and a Python tutorial translated into Japanese, which is available on WWW [32], were used at KEKB. Self-training with these texts is sufficient to start using Python. Users can also find a SAD/Python/medm/java control program launcher.

5.4.2. Python

A Python–EPICS interface has been developed to utilize a scripting language, Python, in the KEKB control system as a one of the two languages described in the previous sub-section. Python is an object-oriented scripting language developed by G. van Rossum. The clear yet powerful syntax of Python is suitable for rapid application development. A Tk widget is integrated with Python as a Python object. It allows a user to access Tk widgets from Python in a seamless way, and makes it easier to develop and maintain a Python program with Graphical User Interface (GUI).

Another advantage of Python is the "module". Functions and classes in the module can be imported into a Python program as a library. Modules can be either written in Python or C. This module structure allows the developer of a Python program to convert a Python module into a C module without affecting the client applications of this module. Well-defined API to access Python program data type from a C program helps to develop a C module. It is also possible to create a module from an existing C program using Simple Wrapper and Interface Generator [33] (SWIG).[3] SWIG can generate a wrapper program for Python from prototype declarations in C, automatically.

A wide range of modules are available for Python, including an ORACLE interface, a PosgreSQL interface, Unix libraries, WWW/HTML support libraries, X windows, Tk widgets, and Gtk widgets. Numerical Python provides many linear algebra functions.

The EPICS interface in Python, PythonCA, consists of two modules. One is a C module, _ca.c, and the other is a Python module, ca.py. The C module provides basic access to the CA library, including open/close, get/put, and monitor. It also supports synchronous group operation at the Python level. A monitor callback function is also defined as a Python function. The Python module, ca.py, defines a channel object and other utility functions. By default, PythonCA registers the socket associated with the channel to the file descriptor manager of the Tk library.

5.4.3. SAD

SAD is the name of an accelerator modeling program and its programming language. The syntax of the SAD programming language is designed after the Mathematica programming language [34]. These two programming languages are almost compatible with each other, except for a few points. The largest difference is the symbolic manipulation of expression. Mathematica was originally developed for it, while SAD cannot handle it. For the numerical handling of mathematics, SAD may in general be faster.

In the original implementation of Tk widget interface was used to be routed to an embedded Python/Tk interface. However, the current SAD can call the Tk widget library directly.

The EPICS interface in SAD supports open/get/put/monitor functions. A monitor callback function can be written in SAD, itself.

Most of the beam-handling applications are written by accelerator physicists in the commissioning group. These applications, including a beam-orbit control program and a beam-collision control program, were registered to an application launcher program, "kbl".

5.5. Portable channel access server

To operate KEKB accelerators, tuning of the linac as the injector for the KEKB rings is thought to be very essential. Ideally, the KEKB control system can control both KEKB rings and the linac. Also both operators in linac control room and in KEKB control room should be able to monitor and adjust the equipment of the other accelerators. For that purpose, we had to develop suitable method in between the two systems to communicate with each other. In EPICS collaborations, there is a portable CA server for EPICS developed at Los Alamos National Laboratory for SUN workstations. We decided to modify it for our purposes, and have been implementing it to the KEKB control system step by step [35]. We can now monitor and set the magnetic field of

---

[3] URL: http://www.swig.org

the Q-magnets in the linac, control the beam-transport magnets in the linac beam line, control klystrons, and measure the beam positions by strip-line monitors through EPICS.

5.5.1. *Connection of two control systems*

To connect the KEKB control system and the existing linac control system, a gateway must be provided between the two systems. In the first case, there is an IOC dedicated to the linac, which keeps the database of the parameters and values of the linac equipment. In the KEKB case, the total number of channels is estimated to be very large, and is too much for an IOC. Therefore, this method may not be realistic. On the other hand, a gateway workstation is provided in the configuration in the second case shown in Fig. 7. The functionality of the EPICS CA is realized by a portable CA server implemented in the gateway workstation. The EPICS portable CA server was designed and developed by members of EPICS collaboration society. Its aim is to integrate a control system which does not use EPICS into a control system based on EPICS. A UNIX gateway computer will be placed between the two systems and will translate messages between the two languages; one is EPICS and the other is a protocol specific to the non-EPICS system. The CA protocol was implemented as a set of C++ classes. A user can easily extend CA server classes using an inheritance technique to build a custom-made CA server.

5.5.2. *Implementation of a CA server*

We have tested the two approaches described above. In both cases, an EPICS record database is automatically generated from the equipment database. It is also useful to have utility programs to generate screen-configuration data for a man–machine interface program, MEDM. For these purposes, tools were designed and developed. A tool named CreateLinacDatabase was developed. It generates a linac parameter configuration file Linac.conf, which in turn is converted to an EPICS database file using dbLoadTemplate, a database template file. Another tool, CreateLinac AdlFile, generates a screen description file for MEDM.

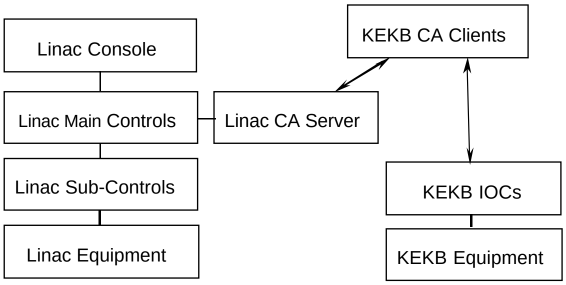

Fig. 7. Schematic configuration of portable channel access between KEKB and injector linac control systems.

5.6. *Future operating system on IOCs*

The operating system on the IOCs is VxWorks. This operating system is not an open system, but a proprietary one. Recent trends concerning software operating systems indicate that open software, like Linux, will be the main stream. EPICS is now ported to a Linux operating system by M. Kraimer at Argonne National Laboratory. But, Linux, itself, is not suitable for real-time purposes, and limited application programs can run on Linux. For real-time applications, we surveyed on the so-called real-time Linux, such as RT Linux, L4-Linux, and other implementation of Linuxen. We have chosen L4-Linux as a base of software to test the idea of EPICS on Linux with a real-time scheduling kernel [36].

L4-Linux is a port of a Linux kernel on top of a real-time micro-kernel, named L4, or its successor, named fiasco [37,38]. In this system, Linux as well as its processes run as an independent task scheduled by a real-time scheduler in L4. L4-Linux was developed while aiming to run a real-time system and a time-sharing system on a single computer, and hence L4-Linux in itself is not a real-time system [39]. However, its basic architecture allows Linux processes, with some modification, to be turned into a real-time process that serves time-critical events. The point here is that the real-time process can share its virtual address space with other normal Linux processes to communicate. This is an advantage in porting IOC core programs, which are based on a multi-threading scheme [40]. In addition, the real-time processes can be connected to a hardware interrupt through only L4, thus avoiding Linux involved in the decision of process dispatching on interrupts. This feature enables IOC application developers to implement device drivers as a user-level application, even if they utilize hardware interrupts. In order to evaluate the feasibility of this scheme, we have ported a CAMAC driver of EPICS to an L4-Linux-based system running on a PC-compliant VME CPU board. We have also measured the interrupt response of the system with heavy disk I/O activities as a background, using the same CPU board. The measured interrupt latency was about 800 μs, which is considerably lower than the value of several tens of milliseconds expected in normal Linux [36]. We expect that the latency can be further reduced, down to around 100 μs or less.

6. Power supply interface controller module

The number of power supplies used for two accelerator rings is more than 2200. Most of them are those for small magnets, like steering magnets. To connect such a large number of power supplies to the IOCs, we adopted ARCNET as a field bus, and developed the Power Supply Interface Controller Module (PSICM) [41], which is an ARCNET interface board for the power supply. A PSICM has the shape of 3U Euro-card format (100 mm × 160 mm) with a DIN 64-pin connector, and can be plugged into the power supply (Fig. 8). The ARCNET allows the use several kinds of media. We adopted a Shielded Twisted-Pair (STP) cable as the media and HYC2485 as the media driver. This configuration allows up to 20 ARCNET nodes to be connected on single segment in a daisy-chain manner. The STP cable includes an auxiliary twisted-pair for the external trigger signal other than for ARCNET use. The maximum number of ARCNET nodes is 20 in one segment, and is limited by the error rate of the communication. Sometimes ARCNET reconfiguration occurs in some conditions affected by

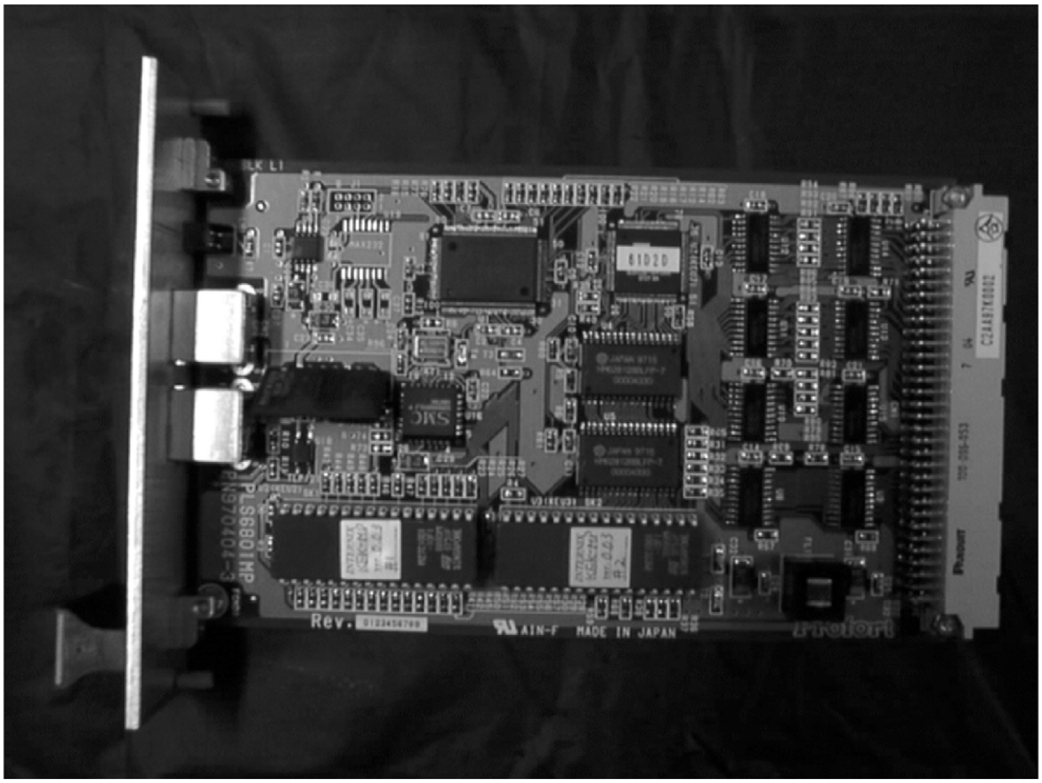

Fig. 8. Photograph of the PSICM.

the temperature and other environmental parameters. We finally introduced an ARCNET hub to reduce number of nodes in a segment to obtain much more stable operation. As described above, there are several types of power supplies. There are unipolar and bipolar power supplies, new and modified old ones, 12 and 16 bits in accuracy. The required specifications for such power supplies are widely distributed. We decided to design the software installed on the PSICM which can recognizes what kind of power supply is connected. In other words, each power supply must have its unique type and serial number data in itself. While ARCNET and PSICM reduced a great deal of wiring cost, there are other control paths between the IOCs and subsystems of the magnet power supply system [24,41]. The power supply output current is monitored by a digital voltmeter with scanner. They are connected to the IOC using GPIB. There are 274 magnet power supplies for the KEKB injection beam-transport lines. They are also controlled in a similar manner.

6.1. Functional requirements and design

Although the KEKB storage rings do not require any synchronous ramping of magnetic fields for acceleration, synchronous operation of the magnet power supplies is still important. For efficient beam handling, the beam optics should be changed without loosing the stored beam. To achieve this, the magnet power supply control system has two features. One is the synchronous operation of PSICMs. A PSICM can receive an external trigger signal to start current tracking. The external trigger signal generated by the IOC in the central control room is distributed to the IOCs in the local control rooms through the software trigger system [8]. From an IOC to PSICMs, the trigger signal is sent through an auxiliary twisted-pair in STP cables for ARCNET. Using an external trigger signal, PSICMs can start tracking synchronously within 0.1 ms or less. The other feature is a flexible tracking curve. The PSICM is designed to receive arbitrary tracking data. This feature is useful for fine synchronous operation. By using a proper tracking curve, it is possible to compensate for the delay of magnetic field against the current setting, which is caused by a delay in the power supply and a delay in the vacuum chamber. This feature gives us another benefit, even for the asynchronous operation. For power supplies with a slow response, a proper tracking curve makes current setting faster without any overshoot than linear tracking. To make application programming simple, the magnet power supply control system requires that a control parameter can be given not only by current in ampere, but also by some abstracted parameter of beam optics, such as K1 for a quadrupole magnet, K0 for a steering magnet, etc. We generically call such parameters “K-values” for simplicity.

6.2. Functions of PSICM

Since the PSICM has a microprocessor, it is not only an ARCNET interface board but also an intelligent controller. The control program is stored in EPROMs on PSICM. It provides advanced functions of the power supply. The PSICM provides the following three functions: (1) generating control signals to the power supply for single actions such as power on/off, interlock reset and a polarity change; (2) setting the output current; and (3) sending the status of the power supply and the PSICM, itself. The status can be sent periodically on demand or when specific status is changed. The PSICM communicates with an IOC through ARCNET. The elemental unit of communication is a “message”. The message from an IOC to a PSICM is called a “command”. More than one command can be packed in single ARCNET packet. The message from a PSICM to an IOC is only the status with a fixed format. To set the output current, the PSICM writes a digital value to a Digital to Analog Converter (DAC) in the power supply. There are three tracking modes and two trigger sources. The tracking modes are as follows: (1) the direct output mode: the output current is set directly to the DAC without tracking. This mode is only for diagnostics of the DAC and PSICM, itself. (2) The constant slewing rate mode: the output current is changed with linear tracking. The PSICM receives the target current and the time of the tracking duration from the IOC and writes a linearly interpolated value to the DAC every 1 ms. This mode is mainly used for stand-alone tests of the power supply. (3) The wave-generator mode: the output current is changed with an arbitrary tracking curve. The PSICM receives tracking data as an array of currents from the IOC, then sequentially writes them to the DAC every internal clock interval, which can be a multiple of 1 ms. This mode is used for usual operation. The constant slewing rate mode and the wave-generator mode require a trigger to start tracking. The trigger source is either a "start" command sent from the IOC through ARCNET or an external trigger signal sent through the auxiliary twisted-pair.

6.3. *Power supply control software*

6.3.1. *Driver level software*

Low-level software is EPICS driver support and EPICS device support in the IOCs. The driver support is designed to perform communication through ARCNET. The elemental services of the driver support are transmitting and receiving single packet. Inside the driver support, packets are queued. There are a transmission queue and a receiving queue for each power supply independently. The device support is designed to provide two kinds of services. One category includes the services dedicated to the PSICM. Each service corresponds to the single command of the PSICM such as power on/off, interlock reset, etc. Using these services EPICS records which perform the single action can be easily created. Other category includes general purpose services which perform single packet I/O. They are nearly same as ones in the driver support. Using these services, although EPICS record support should directly encode or decode messages on an ARCNET packet, it is rather convenient when a record issues various commands sequentially.

6.3.2. *Middle-level software*

Middle-level software provides various logic to the power supply system. Most of them are resident in IOCs using a EPICS runtime database, which is a collection of the EPICS records. Each power supply has a large special record, called a PS-record, which has been developed only for magnet power supply at KEKB. Most of the control logic is concentrated in the PS-record and programmed in C.

6.3.3. *Parameter conversion*

The most complicated logic is current setting. Since an application program may request current setting in terms of a K-value, middle-level software must have parameter conversion logic using some information, such as a magnetic field excitation curve. The parameter conversion is typically carried out in the following manner: (1) a K-value is multiplied by the beam-line momentum to yield an integrated magnetic field strength. The momentum is kept in an EPICS software record. Each storage ring or injection beam transport line has such a momentum record. (2) The integrated magnetic field strength may be modified by a "fudge" factor and a fudge offset. They are introduced to correct the magnetic field excitation curve in an empirical manner. (3) The integrated magnetic field strength is converted to current using a magnetic field excitation curve. Each PS-record has the characteristic parameters of the excitation curve of its own magnet.

6.3.4. *Asynchronous operation*

Asynchronous operation of current setting is an operation on a single power supply independently of other power supplies. For this operation, an external trigger is not used. A PS-record provides four methods of current setting. One is called "Direct Setting", in which current setting is carried out regardless of magnetic hysteresis. The other three methods are "Standardize Setting", "Simple Standardize Setting" and "Sequence Setting". In these methods, the regular hysteresis loop is more or less considered. To set a lower current by Sequence Setting, for example, the current first goes up to the maximum current, and then keeps the maximum current for a moment. Next, it goes down to zero current, and then keeps zero current for a moment. Finally, it goes up to the target current. Thus, unless Direct Setting is used, the magnetic fields is kept on a regular hysteresis loop. These methods are particularly useful for magnets in the injection-beam transport lines.

6.3.5. Synchronous operation

Synchronous operation of current setting is an operation on more than two power supplies simultaneously without loosing the beam in the storage ring. In this operation, only Direct Setting is possible. To perform a synchronous operation, frequent negotiations among power supplies are necessary. As an arbiter of them, we introduced a server process for each storage ring. It manages the sequence of the synchronous operation. The server process runs in the host computer and is programmed in Python. Synchronous operation is carried out according to the following steps: (1) the server receives a request with parameters from an application program. The parameters are a set of power supply ID numbers, a set of K-values to be set and the time of the tracking duration. These parameters are passed through EPICS software records. (2) The server sends the K-values to the PS-records. (3) Each PS-record converts the given K-value to the current, and then estimates the minimum time of the tracking duration and sends it back to the server. (4) The server checks the estimated minimum times. If the application program does not specify a time, the maximum estimated times is adopted. (5) The server sends the adopted time to the PS-records. (6) Each PS-record calculates tracking data and sends them to the PSICM. (7) Each PS-record checks the status of the PSICM. If the PSICM is ready to start tracking, the PS-record sends “ready” to the server. (8) The server waits until all PSICMs become ready, and then the server generates an external trigger signal. (9) The server checks whether all power supplies have started tracking.

7. Communication system

The KEKB accelerator communication system consists of a bi-directional CATV network, a mobile-phone system, a public-address system and a directly connected intercom system.

7.1. CATV system

The CATV system has been working for about 18 years, since TRISTAN was commissioned. It has 33 channels downward and a few channels upward. The frequency range is not popular now in Japan and the equipment is now obsolete. Therefore, we are going to use the KEK in-site network for distributing video information.

7.2. Information exchange system

There are mutual information exchange systems with other section in the KEK site, such as the facility support group, the physics experiment group and the photon factory group. A serial communication line is used to receive information from the facility support group, i.e. air conditioning, cooling water and electric power supplies data. Between the physics experiment group and the KEKB accelerator control group, a gateway IOC is located near the BELLE detector, and information is exchanged through the memory of the IOC. Information exchange between the Photon Factory and KEKB is done mainly using video information and an intercom.

7.3. Video signal distribution system

Almost all of the video signals generated around the KEKB accelerators are sent once to the central control building for distributing them to various parts of the accelerators. The signals are sent through optical-fiber cables using E/O and O/E converter pairs. The signals which come into the central control building are buffered by amplifiers, and some of them are sent to the CATV system. Some of the signals are sent to local control rooms through a patch-panel. Using this system, we can record, distribute or display utilizing commercially available equipment.

8. Radiation safety system

8.1. Overview

The purpose of the radiation safety system of the KEK B-factory (KEKB) is to protect personnel from radiation hazards. It consists of an access control system and a beam interlock system. The access control system protects people from strong radiation inside the accelerator tunnel due to an intense beam by controlling access to the beamline area. On the other hand, the beam interlock system prevents people from radiation exposure by interlocking the beam operation. The present system was originally constructed [42] for the TRISTAN e^+e^- collider, and modified to meet the requirements of KEKB. It secures not only KEKB, but also the PF-AR, which used to be the booster ring of TRISTAN, and is now dedicated to synchrotron photon experiments. For the convenience of accelerator operation and access control, the region covered by the safety system is divided into three major access control areas: the KEKB area, the PF-AR area, and the beam-transport (BT) area. These areas are schematically shown in Fig. 9. The KEKB and BT areas are separated by interarea gates, and so are the PF-AR and BT areas, while a concrete wall of 3 m-thick separates the BT area from the linac area, which is secured by another safety system.

8.2. System architecture

Fig. 10 schematically shows the present safety system. The core of the system consists of Programmable Logic Controllers (PLCs), which satisfy the requirements for the safety system, such as the reliability, maintainability and flexibility. A supervisory PLC and two PLCs, one for access control and the other for communication with the accelerator control system, are installed in the KEKB control building, while local PLCs (PLC#1–PLC#15) are distributed around the KEKB and PF-AR rings. The maximum distance between the supervisory PLC and the local PLC is 2000 m.

Typical local PLCs have 80 input channels and 96 output channels for the status read and control of local devices, such as entrance doors and beam stoppers. Each local PLC has a 16ch output module and a 16ch input module for exchanging safety signals with the supervisory PLC. Because the number of channels for the communication is relatively small, signals from local devices are processed in each local PLC, and the summarized signals are sent to the supervisory PLC. These signals are transmitted by optical-fiber links. In the case of communication errors of the fiber links, the safety system stops accelerator operation.

The main operator interface of the system is a classical panel with key switches and push buttons

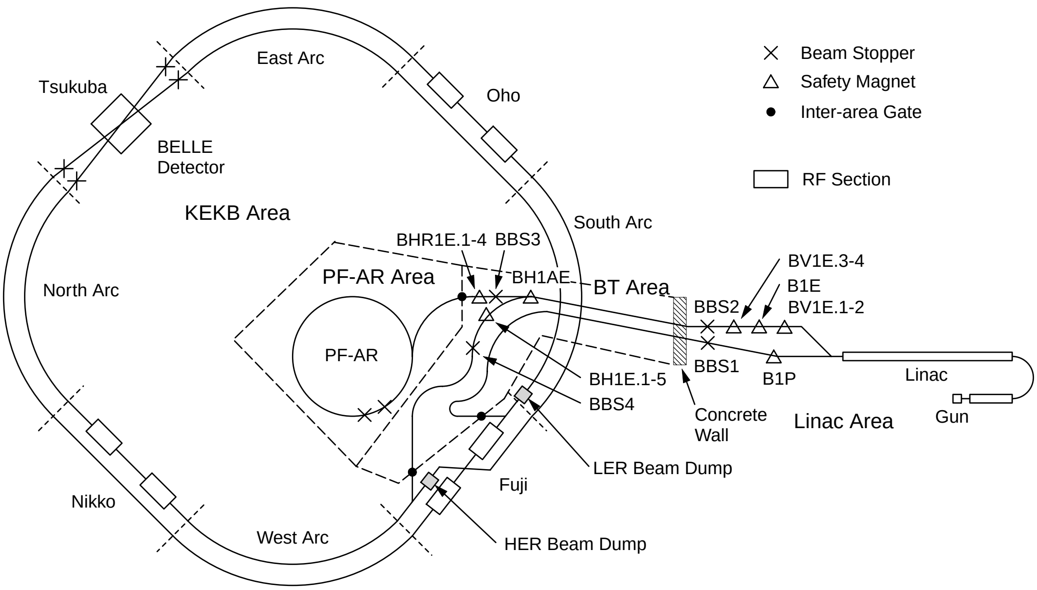

Fig. 9. Access control areas and devices for access safety.

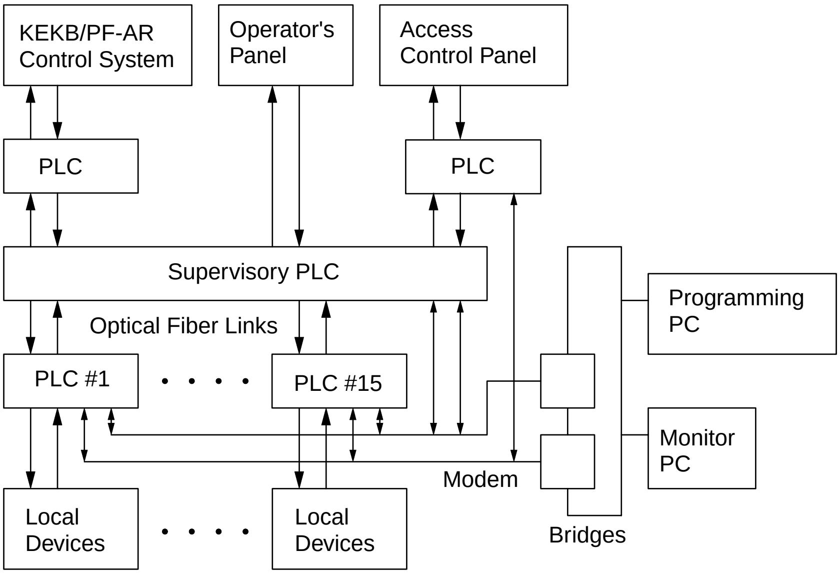

Fig. 10. Architecture of the radiation safety system.

for setting the system state, LED lamps for status indication, and an alarm buzzer. It has the advantage of reliability, but the disadvantage of inflexibility. It provides minimum information about the system status. LED lamps of the operator panel indicate whether each area is safe or not, but cannot display the status of each local device. Detailed monitoring is carried out with color graphic displays. For example, a building outline with doors is indicated on the display, showing the door positions; the status of each door is indicated by color. All of the status information on local devices and PLCs is obtained through Modem links, which are utilized to upload each PLC program developed on a workstation as well. The access control panel is also an important operator interface, which is used for the access control described below.

8.3. Access control system

8.3.1. Entrance

As mentioned before, there are three access control areas: the KEKB area, the PF-AR area, and the BT area. Every area has several entrances and emergency exits. The KEKB area, for example, has eight entrances, seven auxiliary entrances, and 18 emergency exits. Each entrance is equipped with an ID card reader and a key box with safety keys. A person intending to access this area must put his/her ID card into the card reader. The data on the card are checked with a database, and if he/she is registered, a safety key is released. The entrance door can be unlocked using the safety key. He/she is required to keep the safety key during the stay in the area, and return it to the key box when leaving. The safety system can enable/disable the key release and door release for the access control. The access procedure described above is the one used when they are enabled. An intercom and a TV camera are also installed at each entrance to facilitate access control.

8.3.2. Access states

For the present system, there are three access states: Free access, limited access, and no access, which are described below.

Free access: In general, all of the hazardous components are shut down, and the area is safe in this state. Even in this state, however, the entrance doors are locked to prevent access of non-radiation workers. Radiation workers can access the area without an operator's supervision.

Limited access: When an area is transferred to this state from the free access state, a search of the area must be carried out to sweep it. After a search, only access under the operator's supervision is allowed. To prevent access without supervision, the key release and door release are disabled. When access is necessary, the operator enables them by turning on switches at the access control panel. The operator can communicate with workers entering by the intercom and watch them on TV screens.

No access: This is a state for beam operation. No access is allowed. Transition to this state is possible only when no emergency-stop button is depressed, all of the doors are closed and locked, and all of the safety keys have been returned to the key boxes.

8.3.3. Access safety

Safety of each access control area is accomplished by beam stoppers and dipole magnets (safety magnets), which are shown in Fig. 9. To make the KEKB area safe, for example, stored beams in the High-Energy Ring (HER) and the Low-Energy Ring (LER) must be dumped, and injection to both rings must be inhibited. To inhibit HER injection, dipole magnets, BH1AE and BH1E.1-5, are turned off and a BT Beam Stopper BBS4 is inserted. LER injection is prevented by turning off the dipole magnet, B1P, and by inserting a BT beam stopper, BBS1. KEKB has four ring-stoppers: two for LER, and two for HER. However, they are not designed to withstand the high beam power of KEKB, because their purpose is not to dump the high-current beam, but to assure access safety. To dump a beam, a beam-abort system is installed in each ring, which consists of kicker magnets, a Lambertson magnet, and a beam dump. When people access the KEKB area, the abort systems of both rings are triggered first, and then the ring stoppers are inserted into both rings. Even if a small potion of a beam is accidentally left after a beam abort, it is completely dumped by beam stoppers.

The KEKB area is subdivided into eight sections by intersection gates: four straight and four arc section shown in Fig. 9. Normal-conducting RF cavities are installed in the Oho and Fuji straight section and superconducting RF cavities in Nikko straight section. The RF operation of the cavities is interlocked to be off when the corresponding subsection is in the free access state. Door switches are also included into the RF interlock to prevent people from X-ray exposure during access in the limited access state.

8.4. Beam interlock system

The beam interlock system consists of emergency-stop buttons, door switches, safety keys, radiation monitors, beam-loss monitors, and meter relays. The main purpose of the former three kinds of devices (emergency-stop buttons, door switches, and safety keys) is to protect people against radiation exposure inside the accelerator tunnel. On the other hand, the function of radiation monitors, beam-loss monitors, and meter relays is to protect people outside the accelerator housing. The response of the safety system to the activation of each interlock device is summarized in Fig. 11 and described below.

Emergency-stop button: When an emergency-stop button is pushed, the safety system triggers the beam-abort system, and turns off all of the ring magnets and the RF system. A beam-request signal to the linac is removed to stop beam injection. Furthermore, safety magnets are turned off, and BT stoppers are inserted. When an area is transferred to the no access state and beam operation is going to be started, an audible alarm is sounded in the area. People accidentally left without a safety key can prevent beam operation by pushing an emergency-stop button.

Door interlock: Every door separating an access control area from other areas has an electrical lock and at least one switch to indicate the door status. If a door fault is detected during beam operation, the safety system sends a trigger signal to the beam-abort system, and turns off one ring magnet of each ring. Beam injection is also stopped by removing the beam-request signal, turning off the safety magnets, and inserting the BT stoppers. In the no access state, the door release signal to the electrical lock is disabled.

Safety key: A key box with safety keys is equipped near to each entrance together with an

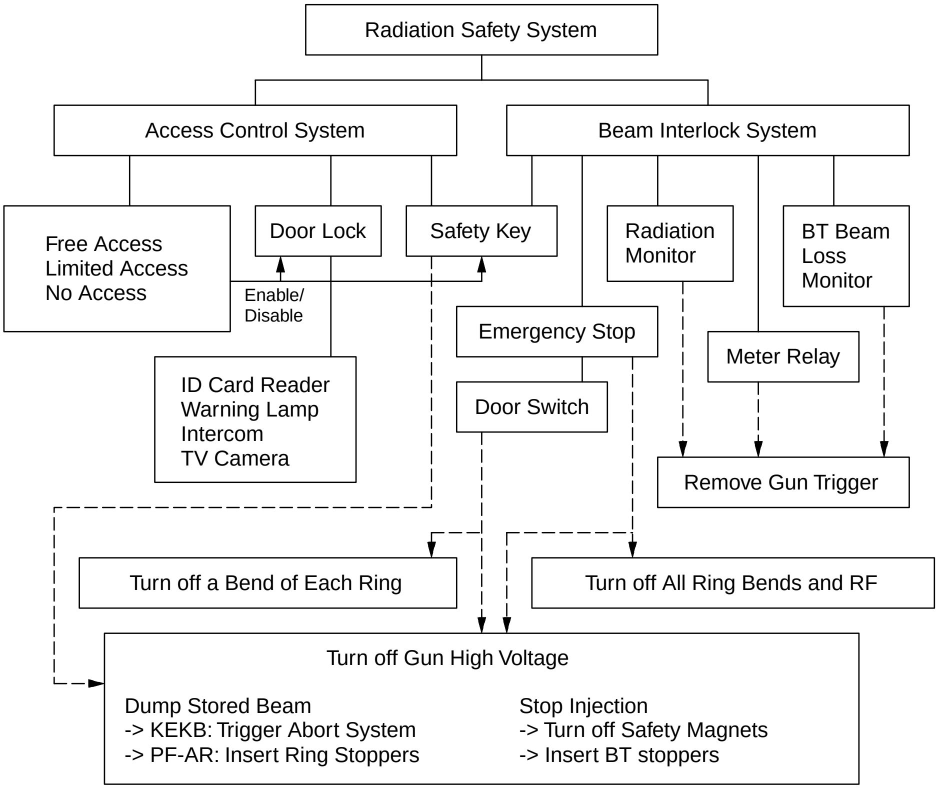

Fig. 11. Devices for access control and beam interlock.

ID card reader for key release. It is required for everyone accessing the area to take a key and to carry it during a stay in the area. Once a key is removed, the beam operation is interlocked until all keys are returned. The key release is disabled in the no access state.

Radiation monitor: Although concrete shielding has been added along the beam line in the experimental halls, the experimental halls are still the weakest parts concerning the radiation containment, because the other parts of the KEKB ring are covered with earth soil with a thickness of 6.7 m. The shielding walls are designed so that the radiation level due to beam loss under normal operation is less than the acceptable limit. However, accidental beam loss may cause a higher radiation level outside the shielding walls. Therefore, two radiation monitors are installed in every experimental hall. Each monitor has a BF3 proportional chamber for neutrons and an air-filled ionization chamber for photons. If the radiation level becomes higher than the preset limit, the safety system stops beam injection to the KEKB rings.

Beam-loss monitor: Beam-loss monitors were installed in the BT area, which are included in the interlock system. If there is an accidental beam loss in BT, the radiation level on the ground above the BT tunnel may become larger than the limit. Therefore, the beam injection is interlocked to be off. Beam-loss monitors are also installed in the KEKB area, and are used to stop beam operation for machine protection. They are not included in the safety system because the KEKB tunnel has sufficient shielding.

Meter relay: The current of the dipole magnet, BH1AE, is monitored by a meter relay. If the current of BH1AE is accidentally turned off during KEKB injection, the electron beam is transported to the AR injection line and lost somewhere near the BT-AR boundary. This may increase the radiation level of the AR area near the boundary. When a decrease of current is detected during KEKB injection, therefore, the beam is turned off by removing the gun trigger.

8.5. *Other hazards*

In the accelerator tunnel, there are various hazards other than the radiation. They are electrical (high voltage, and high current), high magnetic field, and cryogenic hazards. The safety system is also useful to control the access into an area with these hazards.

In the interaction region of the KEKB ring, there are superconducting quadrupole magnets and a solenoid magnet of the BELLE detector, all of which yield strong leakage magnetic fields. Therefore, the excitation of these magnets is usually carried out after the Tsukuba straight section is transferred to the limited access state.

Superconducting accelerating cavities are installed in the Nikko straight section. The accidental release of liquid helium or nitrogen in the accelerator tunnel may cause a shortage of oxygen. Therefore, access is limited during operation of the cryogenic system. The oxygen level is monitored at several points along the beam line. If the oxygen level decreases below 18%, the safety system sounds an alarm to evacuate the Nikko straight section.

9. KEKB timing system

9.1. *Introduction*

The KEKB control system required a new timing system to match a low longitudinal acceptance due to a low-alpha machine. This timing system is based on a frequency divider/multiply technique and a digital delay technique. The KEKB timing system is slightly complicated, because the KEKB ring RF frequency (508.887 MHz) is not a divisor of the Linac RF frequency (2856 MHz). The KEKB ring frequency and the linac frequency are locked with a common divisor frequency (10.385 MHz). The common divisor frequency determines the injection timing. This section gives an overview of the KEKB timing system and the RF bucket selection system [43].

9.1.1. *Difference in the injection schema between TRISTAN and KEKB*

At TRISTAN a 5-bunch injection system was adopted, which means that 5 bunches of the linac beam were merged into a bunch of the TRISTAN beam. We cannot inject multi-bunch beams in order to match the low longitudinal acceptance, because the KEKB ring is low-alpha machine. Timing jitter between injected beam and RF bucket is allowed less than 30 ps. That of low-level control system is limited within several picoseconds. We thus intended to inject a single, high-current bunch beam. Furthermore, we synchronize the KEKB RF frequency (508.887 MHz) and linac frequency (2856 MHz) so that we can always catch Linac beams at the same phase of the KEKB RF frequency.

9.1.2. 2856 *and* 508.887 *MHz*

Although the KEKB RF frequency is not a divisor of the linac RF frequency, those frequencies have a common divisor frequency (10.385 MHz). Because of the existence of a common divisor frequency, we can synchronize the KEKB RF bucket timing and Linac beams at the common divisor frequency intervals. The linac RF frequency and the KEKB RF frequency are locked by a common divisor frequency with a newly developed multi-synthesizer.

9.2. *Multi-synthesizer*

There are two types of multi-synthesizers newly developed for KEKB. We now introduce one type. Fig. 12 shows a block diagram of the synthesizer. The source of the synthesizer is 571.2 MHz, which is used with a subharmonic buncher in the linac. The frequency, 571.2 MHz is multiplied by 5 and generates 2856 MHz, which is used as the linac RF frequency. Simultaneously, the frequency 571.2 MHz is divided by 5 and generates a 114.2 MHz frequency, which is also used with

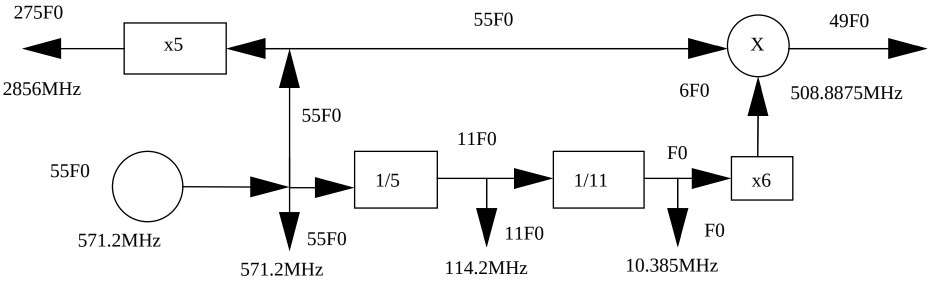

Fig. 12. KEKB multi-synthesizer.

Table 2
The reference frequencies used in KEKB ring and linac

KEKB ring reference (MHz)	Linac reference (MHz)
	2856
	571.2
508.887	508.887
	114.2
10.385	10.385

another subharmonic buncher. The 114.2 MHz frequency is divided by 11 and generates a 10.385 MHz frequency, which is a common divisor frequency. The common divisor frequency is multiplied by 5 and mixed with 571.2 MHz frequency, and finally generates a 508.558 MHz frequency, which is the KEKB RF frequency. All of the frequencies are connected and locked with a common divisor frequency 10.585 MHz. All reference frequencies are listed in Table 2.

9.3. Overview of the KEKB timing system

9.3.1. Single reference frequency in the KEKB machine

Although the KEKB machine consists of two rings (HER and LER), it has a single reference frequency 508.887 MHz. We cannot change the reference frequency of each ring independently. Of course the phase of the reference frequency of each ring can be changed independently. We avoided the complication of a double reference system. Since we never measure the dispersion parameters or the chromaticity parameters of both rings simultaneously, no problems have occurred with the single reference system.

9.3.2. Distributing the reference frequency

The reference frequency is distributed with coaxial cables and optical fiber cables as shown in Fig. 13 Main line which is circulated around the KEKB ring consists of coaxial cables, which are locked with PLL in phase. The main line has stabilized within 1° around the KEKB ring. Satellite lines consist of optical fiber cables, of which phase stabilities match less than 0.2 ppm per degree in electrical length. The satellite line has stabilized less than 0.5° in total (Fig. 14).

9.4. Bucket selection method

9.4.1. 10.385 *MHz and* 2.028 *kHz*

Since the KEKB RF frequency (508.887 MHz) and the linac frequency (2856 MHz) are locked in 10.385 MHz intervals, as shown in the synthesizer paragraph, we can inject linac beams in this intervals. The interval equals a 49-bucket spacing in the KEKB ring. Since the frequency (10.385 MHz) divide by 5120 (KEKB harmonic number) equals 2.028 kHz, we can inject linac beams in 2.028 kHz intervals at the same bucket in the KEKB ring. We thus choose a frequency of 2.028 kHz as the basic injection frequency. We can choose any buckets with delay timing from the basic frequency in 10.385 MHz interval units, that is 49-bucket spacing in the KEKB ring. Since the number 49 and the number 5120 have no common divisor, the 10.385 MHz intervals

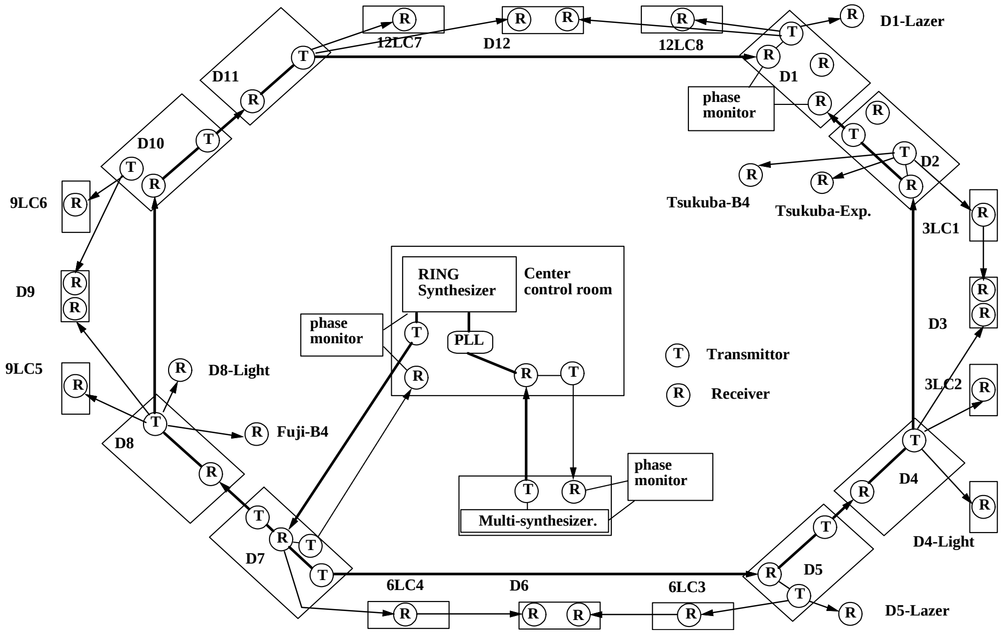

Fig. 13. Reference frequency distribution diagram.

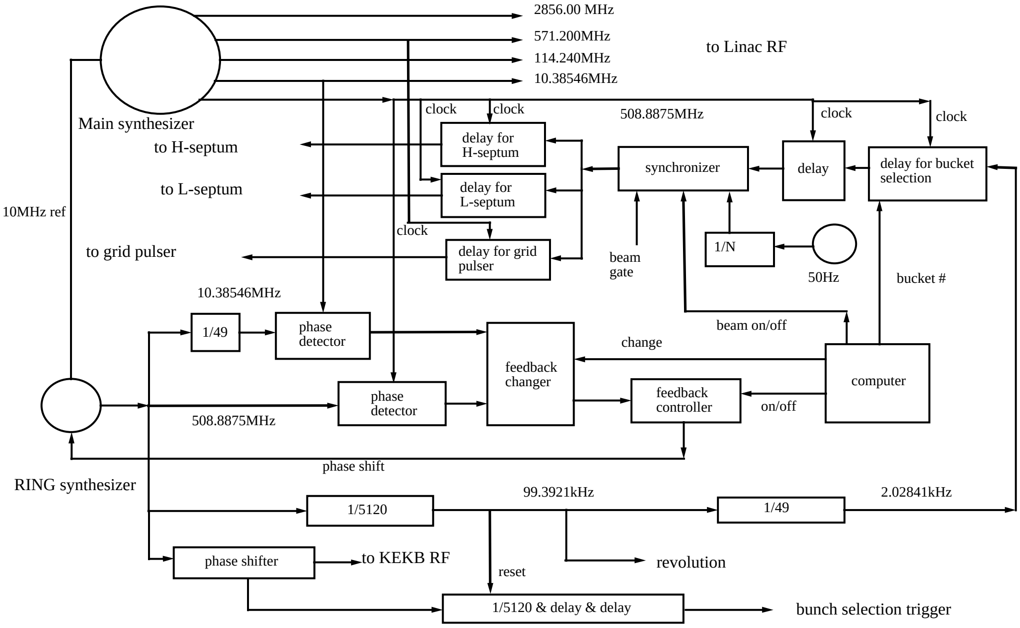

Fig. 14. KEKB timing diagram.

(49-bucket spacing) times 5120 equals 2.028 kHz interval. Thus, within a 2.028 kHz interval we can select any bucket addresses, as shown by

$$\text{Address\#} = \text{mod}(\text{delay\#} \times 49, 5120).$$

In reverse

$$\text{Delay\#} = 209 \times \text{mod}(\text{address\#}, 49) + \text{int}(\text{address\#}/49).$$

9.4.2. Injection phase and collision phase

We can match the phase between the KEKB RF bucket and the linac beam by changing the KEKB reference frequency phase. The phase between the electron and positron collision timing at the intersection region can be adjusted by changing the LER RF phase.

9.5. Frequency shift and phase lock to the linac synthesizer

At injection timing, the KEKB RF frequency and the linac frequency are locked at a frequency 10.385 MHz. After injection, The KEKB RF frequency can be changed in order to adjust the ring circumference and in order to measure the dispersion and chromaticity parameters. When the KEKB RF frequency is changed, the frequency-lock system with the linac frequency is killed. At the next injection timing, we first lock the KEKB 10.385 MHz frequency with the linac 10.385 MHz frequency and then lock the KEKB 508.887 MHz frequency with the linac 508.887 MHz frequency. We can thus continuously add KEKB ring beams even after frequency changing. The linac 508 MHz frequency and the linac 2856 MHz frequency are always locked with the linac 10.385 MHz frequency.

9.6. Beam abort timing

In order to protect insertion detectors and accelerator instruments, we must sometimes abort beam in the case that the beam irradiate radiation to the detector, or in the case accelerator instruments suffer heavy beam loading. Since we introduced kicker magnets to abort beam, of which rise time is around 1 μs, we cannot fill the beam during the rise time interval, which is called beam gap. Besides, we have to adjust the abort timing to the beam gap when the beam is aborted. Thus, the abort kickers are synchronized with the revolution timing that is connected the beam gap.

9.7. Summary

Since we introduce single-bunch injection and a frequency lock system between the KEKB ring frequency and the linac frequency, we can inject linac beams within several psec jitters. We can select any bucket address in the KEKB ring, change the ring frequency freely without injection timing, and inject linac beams continuously during the next injection timing.

Acknowledgements

The authors thank Prof. Shin-ichi Kurokawa for his valuable support and advice during designing and constructing the KEKB control system. They also want to express their thanks to link-persons for their cooperation.

The authors also wish to thank members of the Radiation Science Center of KEK for their help in constructing the radiation safety system.

References

[1] T. Katoh, S. Kurokawa, Design of KEKB control system, Proceedings of the CTDCA, VECC, Calcutta, December 6–8, 1994.

[2] KEKB Design Group, KEKB B-Factory Design Report, KEK Report 95-7, August 1995.

[3] T. Katoh, S.I. Kurokawa, T.T. Nakamura, T. Mimashi, N. Yamamoto, Control system design for KEKB accelerators, IEEE PAC, 1995, pp. 2205–2207.

[4] T. Katoh, et al., Present status of the KEKB control system, Proceedings of the ICALEPCS'97, Beijing, China, November 3–7, 1997, pp. 15–18.

[5] T. Katoh, et al., Status of the KEKB accelerator control system development, in: Harima Science Garden City 1997, Accelerator Science and Technology, 1997, pp. 452–454.

[6] A. Akiyama, S. Araki, J.-I. Odagiri, T. Katoh, T. Kawamoto, I. Komada, K. Kudo, T.T. Nakamura, T. Naito, N. Yamamoto, M. Kaji, T. Kitabayashi, K. Yoshii, N. Koizumi, M. Takagi, S. Yoshida, KEKB control

system: the present and the future, Proceedings of the 1999 Particle Accelerator Conference, New York, 1999, pp. 343–345.
[7] T. Naito, J. Odagiri, M. Suetake, N. Yamamoto, T. Katoh, Synchronization scheme for the KEKB accelerator, Proceedings of the ICALEPCS'97, Beijing, November 3–7, 1997, pp. 247–249.
[8] T. Katoh, A. Akiyama, K. Kudo, T.T. Nakamura, J.-I. Odagiri, N. Yamamoto, Performance of the timing system for KEKB, Proceedings of the Seventh International Conference on Accelerator and Large Experimental Physics Control System (ICALEPCS'99), 1999, Trieste, Italy, pp. 214–216.
[9] T. Katoh, S. Takeda, K. Kudo, A. Akiyama, Communication System for the TRISTAN Accelerators, Washington PAC, 1987:067, 1987.
[10] L. Dalesio, et al., Distributed software development in the EPICS collaboration, Proceedings of the ICALEPCS'95, Chicago, USA, 1995.
[11] L. Dalesio, et al., The experimental physics and industrial control system architecture: past, present, and future, Proceedings of the ICALEPCS93, Berlin, Germany, 1993, pp. 179–184.
[12] W. McDowell, et al., EPICS Home Page, http://epics.aps.anl.gov/asd/controls/epics/EpicsDocumentation/EpicsGeneral/
[13] Vsystem, a software product of Vista Control Systems Inc., http://www.vista-control.com/
[14] M.C. Crowley-Milling, et al., The NODAL System for the SPS, CERN LAB II-CO/74-2, 1974.
[15] M. Lutz, Programming Python, O'Reilly & Associates Inc., USA, 1996.
[16] Python Home Page, URL: http://www.python.org/
[17] SAD home page at http://www-acc-rheory.kek.jp/SAD/sad.html
[18] T. Katoh, A. Akiyama, K. Kudo, T.T. Nakamura, J.-I. Odagiri, N. Yamamoto, The operators consoles for KEKB accelerators, Proceedings of the Seventh International Conference on Accelerator and Large Experimental Physics Control System (ICALEPCS'99), 1999, Trieste, Italy, pp. 101–103.
[19] T. Katoh, A. Akiyama, K. Kudo, T.T. Nakamura, J.-I. Odagiri, N. Yamamoto, Ergonomics in KEKB control system, Proceedings of the Workshop on Accelerator Operation 2001 (WAO 2001), Villars-sur-Ollon, Switzerland, January 28–February 2, 2001, pp. 5–9.
[20] IBM now calls it a NetVista^{TM} thin client. URL: http://www.pc.ibm.com/us/netvista/thinclient.html
[21] J. Odagiri, T. Katoh, K. Kudo, S. Kurokawa, T.T. Nakamura, N. Yamamoto, Interfacing modbus plus to EPICS for KEKB accelerator control system, Proceedings of the Seventh International Conference on Accelerator and Large Experimental Physics Control System (ICALEPCS'99), 1999, Trieste, Italy, pp. 367–369.
[22] J. Odagiri, et al., Performance evaluation of EPICS on PowerPC, Proceedings of the ICALEPCS'97, Beijing, China, November 3–7, 1997, pp. 240–242.
[23] Per Cederqvist et al., Version management with CVS, URL: http://www.cvshome.org/docs/manual
[24] T.T. Nakamura, A. Akiyama, T. Katoh, Ta. Kubo, N. Yamamoto, M. Yoshida, Magnet power supply control system in KEKB accelerators, Proceedings of the Seventh International Conference on Accelerator and Large Experimental Physics Control System (ICALEPCS'99), 1999, Trieste, Italy, pp. 406–408.
[25] M. Kaji, et al., Relational database system in the KEKB accelerator control system, Proceedings of the First Asian Particle Accelerator Conference (APAC98), Tsukuba, Japan, 1998, pp. 501–503.
[26] T. Katoh, A. Akiyama, S. Araki, J.-I. Odagiri, T. Kawamoto, I. Komada, K. Kudo, T. Naito, T.T. Nakamura, N. Yamamoto, M. Kaji, T. Kitabayashi, N. Koizumi, M. Takagi, S. Yoshida, Some utilities for EPICS toolkit, Proceedings of the First Asian Particle Accelerator Conference, Tsukuba, Japan, 1998, pp. 504–506.
[27] T.T. Nakamura, T. Katoh, N. Yamamoto, Development of the Python/TK widgets for the control system based on EPICS, Proceedings of the Seventh European Particle Accelerator Conference (EPAC2000), Vienna, Austria, 2000, pp. 1865–1867.
[28] N. Yamamoto, A. Akiyama, S. Araki, T. Katoh, T. Kawamoto, I. Komada, K. Kudo, T. Naito, T.T. Nakamura, J.-I. Odagiri, M. Kaji, N. Koizumi, M. Takagi, S. Yoshida, Use of object oriented interpretive languages in an accelerator control system, Proceedings of the Seventh International Conference on Accelerator and Large Experimental Physics Control System (ICALEPCS'99), 1999, Trieste, Italy, pp. 600–602.
[29] N. Yamamoto, M. Takagi, Dynamic execution of scripts on EPICS IOC, Proceedings of EPAC 2000, Vienna, Austria, pp. 1883–1885.
[30] PCL (Process Control Language) Manual, Hitachi Limited, Japan.
[31] J.K. Ousterhout, Scripting: Higher Level Programming for the 21st Century, http://www.scriptics.com/people/john.ousterhout/scripting.html
[32] Guido van Rossum, L. Fred, Python Tutorial, Drake http://www.python.org/doc/current/tut/tut.html; Japanese Translation http://www-acc.kek.jp/WWW-ACC-exp/KEKB/control/Activity/Python/doc-j/tut-ja2.1.pdf
[33] SWIG: Simplified Wrapper and Interface Generator, URL: http://www.swig.org/
[34] S. Wolfram, Mathematica: A System for Doing Mathematics by Computer, 2nd Edition, Addison-Wesley, Reading, MA, USA, 1991.
[35] N. Yamamoto, K. Furukawa, N. Kamikubota, T.T. Nakamura, T. Katoh, M. Kaji, T. Kitahayashi, S. Yoshida, Implementing portable channel access server software in the KEKB accelerator control system, Proceedings of the First Asian Particle Accelerator Conference, Tsukuba, Japan, 1998, pp. 498–500.
[36] J.-I. Odagiri, T. Katoh, N. Yamamoto, Porting EPICS to L4-Linux based system, Proceedings of the PAC2001, Chicago, Ill. USA, June 18–22, 2001.

[37] http://os.inf.tu-dresden.de/L4/LinuxOnL4/

[38] http://os.inf.tu-dresden.de/fiasco/

[39] http://os.inf.tu-dresden.de/drops/

[40] M. Kraimer, et al., EPICS: porting iocCore to multiple operating systems, ICALEPCS'99, Trieste, Italy, October 1999.

[41] A. Akiyama, et al., KEKB power supply interface controller module, Proceedings of the ICALEPCS'97, Beijing, China, November 3–7, 1997, pp. 243–245.

[42] S. Takeda, et al., Contribution to the 1987 Particle Accelerator Conference, March 16–19, 1987, Washington DC, USA; KEK Preprint 86–126, March 1987.

[43] E. Kikutani, M. Suetake, M. Tobiyama, Bucket Selector System for KEKB, Proceedings of the seventh International Conference on Accelerators and Large Experimental Physics Control Systems (ICALEPCS'99) 1999, Trieste Italy, pp. 278–280.