

Present Status of the KEKB Control System

Tadahiko KATOH, Atsuyoshi AKIYAMA, Sakae ARAKI, Masahiro KAJI*
 Teruyuki KAWAGUCHI**, Takashi KAWAMOTO, Ichitaka KOMADA, Kikuo KUDO,
 Takashi NAITO, Jun-ichi ODAGIRI, Makoto TAKAGI***, Susumu YOSHIDA***, and Noboru YAMAMOTO,
 Accelerator Facility, KEK, Tsukuba, Ibaraki, 305 JAPAN

(*) Mitsubishi Electric Co. Ltd., Japan

(**) Mitsubishi Electric System & Service Engineering Co. Ltd., Tsukuba, Ibaraki, 305 Japan

(***) Kanto Information Service, Tsuchiura, Ibaraki, 300 JAPAN

Abstract

Construction of the KEKB storage rings is now in the last phase. The control system for these two rings is under way. The main server workstation MELCOM ME-RK460 with two CPUs is connected to an FDDI switching equipment DEC GIGA Switch. From the FDDI switch, 26 optical fibre links are connected to 26 sub-control rooms around the rings and the injector linac. At present, 15 VME-bus based Input/Output Computers(IOCs) are installed in the local control rooms. About 60 IOC's will be added to the system this year. The software system is based on the EPICS Tool Kit distributed through EPICS Collaboration. The relational database system, ORACLE, is used to store all the data about rings. By using this database, EPICS channel database records can be generated automatically.

1 Introduction

KEKB[1] is a project to build an asymmetric electron-positron collider in Japan. The first commissioning of the machine is scheduled in the fall of 1998. The main parameters of the KEKB accelerators are summarized in Table 1.

Ring		LER	HER	
Energy	E	3.5	8.0	GeV
Circumference	C	3,016.26		m
Luminosity	L	10^{34}		$1/cm^2/s$
Beta function at IP	β_x / β_y	0.33/0.01		m
Beam current	I	2.6	1.1	A
Particles / bunch		3.3×10^{10}	1.4×10^{10}	
Emittance	ϵ_x	1.8×10^{-8}		m
	ϵ_y	3.6×10^{-10}		
Synchrotron tune	ν_s	0.01 ~ 0.02		
Betatron tune	ν_x	45.52	46.52	
	ν_y	45.08	46.08	
RF voltage	Vc	5 ~ 10	10 ~ 20	MV
Harmonic number	h	5,120		
Bending radius	ρ	15.3	76.6	m

Table 1
Main parameters of KEKB

Figure 1 shows a schematic view of the KEKB accelerator control system. The system consists of three layers,

- Presentation layer: A Unix server and X-terminals in the central control room,

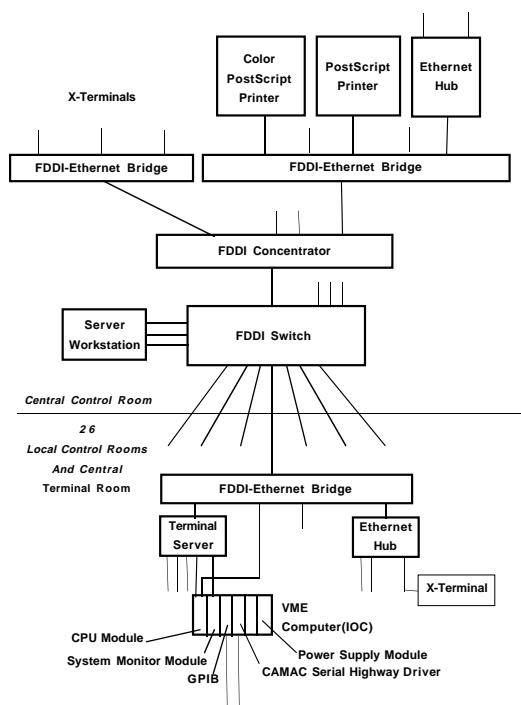


Figure. 1. Schematic view of the KEKB accelerator control system.

- Equipment control layer: VME single board computers distributed in the local control rooms, and
- Device interface layer: VME/CAMAC/VXI modules and GP-IB/ARCNET devices.

The network system which connects the first two layers is another key component of the control system. The network in the KEKB control system is described in Section 2.

As reported before[2], We use the EPICS [3] tool kit as a foundation of the control system. Section 3 briefly summarizes architecture of the EPICS and related topics.

A relational database is another key component of the KEKB accelerator control system and is discussed in Section 4.

We developed a magnet power supply controller board for the KEKB accelerators. It uses ARCnet as a communication interface and to reduce cost and space of cabling for 2,154



Figure 2. The front panel of the FDDI switch in the KEKB central control building. Each optical fibre cables are connected to the FDDI-Ethernet bridges in the local control rooms.

magnet power supplies of the KEKB accelerators. Development of the board is briefly reported in Section 5. Ref. [4] reports the status of this power supply controller board in detail.

2 Network

As shown in Figure 1, the network uses a switched FDDI network[Figure 2] as a backbone network. An FDDI/Ethernet bridge bridges the FDDI backbone network and an Ethernet network in a local control room. The Ethernet part of the bridge works as a switching hub rather than a simple bridge. This configuration reduces the possibility of packet collision.

We adopted a star configuration and a fast FDDI switch for the backbone network. Each local control room is connected directly to the central part of the accelerator control network. It gives us the possibility of transition to the future network technology, such as an ATM, without re-cabling of fiber optical cables.

A console(serial) port of an IOC is connected to a terminal server. User can remotely access IOCs through both the Ethernet and a serial port of the terminal server. This is useful to diagnose the status of the IOC when it does not respond to the network. A VME system monitor board is also connected to a serial port on the terminal server. The VME system monitor board monitors voltages of power supply lines in VME-bus and several external signals. When the board detects an abnormal status, it sends the alarm message through the serial link. The VME system monitor board resets the VME backplane when it receives the reset command through this serial connection.

The FDDI backbone network as well as branch Ethernet networks were installed in spring of 1997. The installed system is used regularly for the development of software and

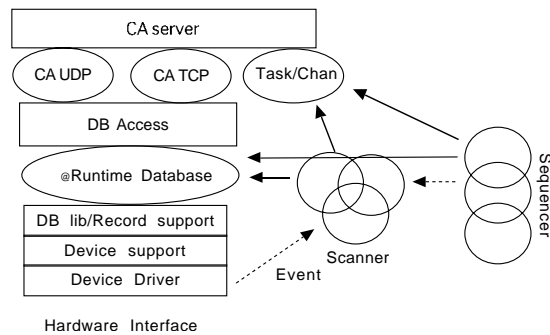


Figure 3. EPICS core softwares running on an IOC. Scanner processes scan the status of hardware in the way specified by the user.

for hardware test. So far no problem has been found in the system except initial hardware trouble.

3 EPICS

EPICS is a collection of software tools developed by the EPICS collaboration. EPICS assumes that the control system consists of 1) Distributed intelligent Input/Output Controllers(IOCs), 2) High-speed network based on TCP/IP and 3) Unix and/or Windows NT host computers for various high level control applications including human interfaces and data logging.

A VME single board computer running the VxWorks real-time operating system is used as an IOC in EPICS. The IOCs are distributed along the accelerator.

The EPICS run-time database on the IOC and CA constitute the core of the EPICS software. EPICS client software accesses the record in this run-time database using the logical name of the record. The protocol to access data in the distributed run-time data base is called "Channel Access(CA)". Read and write accesses to the EPICS run-time database from the EPICS client software triggers the access to the hardware from the IOC. User of EPICS only needs to supply hardware specific routines, hardware driver and device support routines.

EPICS core software also includes programs such as 1) A scanner to scan the status of hardware in the way specified by the user, 2) A CA server to handle the database access request from the client program. Configuration files prepared by the user define the actual behaviors of these programs [Figure 3].

A user can also extend the EPICS core software using the hooks in itself. For example, a user can add support for a new device to the EPICS by supplying a device driver and/or device support routines.

EPICS CA applications such as MEDM(Motif based Editor & Display Manager), ALH(Alarm Handler) and AR_cmd(EPICS Archiver) run on the host computer. These



Figure. 4. Mitsubishi ME RK-460 computer running HP-UX 10.20.

applications are driven by configuration files. So the EPICS user can build a basic control application without any programming.

We use Mitsubishi ME RK-460 computer(HP-UX 10.20) equipped with two CPUs [Figure 4] as the host computer in the KEKB accelerator control system. The host computer also serves a development environment for VxWorks which is the operating system of IOCs. X-terminals are used as an operator interface in the KEKB accelerator control system.

We currently use version R3.12 of EPICS on this host machine. A test of the latest release of EPICS software, R.3.13 beta 11, is under way.

3.1 CA server

The latest release of EPICS base software, R3.13, includes a portable CA server. The portable CA server is a framework to develop a gateway program which bridges the EPICS based control system and the other control system.

In KEKB project, it is required to control the LINAC control system, which is developed in house, and the KEKB ring control system, which is based on EPICS, from the same operator console. The gateway program to the LINAC control system will make this possible. The prototype of the gateway program to the LINAC control system, called LINAC-CA server, was developed based on the pre-released version of the portable CA server code[5]. The code is ported to EPICS R3.13 on HP-UX10.20. This prototype can control only magnet power supplies in the LINAC control system. The ability to control other devices in the LINAC control system is being implemented.

3.2 SAD-Python

MEDM and/or EDD/DM (EPICS Display Editor/Display Manager) are useful development tools for building basic control panels. 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. Python[8]-EPICS interface is being developed to fulfill this shortcoming. Python is an object oriented

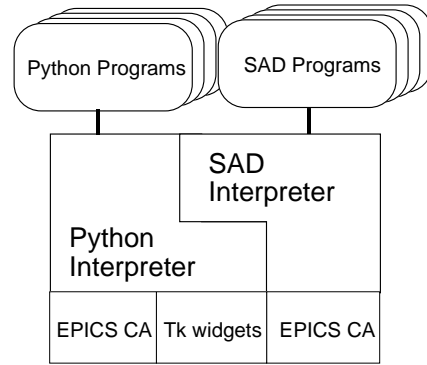


Figure. 5. SAD/Python relation. SAD embeds the Python interpreter in it. SAD calls Python to use a Tk widget. Both SAD and Python has EPICS Channel Access interface for the historical reason.

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 GUI(Graphical User Interface). The embedded Python interpreter in the SAD[6] code allows a SAD programmer to develop a program with the GUI as a SAD program. Figure 5 shows the relation of SAD, Python and EPICS-CA in the current implementation.

3.3 Power PC

We have been using Force CPU40 VME board computers for the evaluation of the EPICS software. We introduced Force CPU64 CPU boards for better CPU performance in FY 1996. We have tested a VME single-board computer based on a Motorola PowerPC604e running at 200MHz and found a ten times CPU performance improvement over the CPU64 in EPICS environment. A comparison of the performance of these VME single board computers in the EPICS environment is given in [[7]].

Experience of porting the EPICS core software to the PowerPC architecture encourages us to choose new CPU board for much better performance. We will introduce a new CPU board based on the MPC700 CPU running at 266MHz later this year. It should also be pointed that the ease of porting greatly depends on the compatibility of VxWorks libraries and the compiler between two architectures.

4 Relational database

We will use relational databases for 1) management of the data used in the accelerator control system, such as hardware addresses, and 2) archiving of the data taken by the accelerator control system.

The first database is used to generate the configuration files for the EPICS runtime database. The logical design of



Figure 6. Examples of the user interface to the ORACLE database using WWW browser.

the database based on Entity-Relation(ER) diagrams, is in progress.

The second database keeps various kinds of data from the accelerators, such as orbit position data and the magnet configuration. The correlation between these data should be retrieved from the database.

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. The device database can be accessed by a WWW(World Wide Web) browser. Figure 6 shows an example of the screen displayed by the WWW browser accessing the device database. Oracle Web Server on the host machine dispatches a request from the browser to the application program written in the SQL Plus language. The application program dynamically generates HTML documents and sends them back to the Oracle Web server. It makes us free from the maintenance of the client program. A poor development environment and incompatibility between WWW browsers are current major problem in this approach but they will be resolved soon.

5 Power supply controller

Over 2,400 magnet power supplies will be connected to the accelerator control system. It became very important to reduce the number of cables between the magnet power supplies and the accelerator control system. We developed power supply controller card which can be installed inside the magnet power supplies. They use ARCnet as a field-bus for the communication with the accelerator control system. Around twenty power supply controller cards are daisy-chained in a single ARCnet link. The board has a CPU unit that manages ARCnet communication and the ramping of the output current. Prototype cards of the power supply controller[Figure 7] were built and tested[4]. The controller cards successfully controlled power supply units in the EPICS environment.



Figure 7. A photo of the power supply control board based on ARCnet.

6 Conclusion

A basic set of the KEKB accelerator control system was installed and has been working. The power supply control card design was fixed and the cards for the magnet power supplies will be delivered by the end of this fiscal year. Power PC based IOCs will be also installed in the system in this fiscal year.

Software development/improvement will be continued until the commissioning of the KEKB accelerators, even after it. Designing the EPICS database template for each hardware device will be the major effort in the software development.

References

- [1] "KEKB B-Factory Design Report", KEK Report 95-7, August 1995
- [2] T.Katoh et al., "Status of the KEKB accelerator control system development's", ICALEPCS95
- [3] B. Dalesio et al. "Distributed Software Development in the EPICS Collaboration," in Proceedings of International Conference on Accelerator and Large Experimental Physics Control Systems, M. Crowley-Milling, P.Lucas and P.Schoessow, Eds. (ICALEPCS,Chicago, USA, 1995),pp 360 - 366.
- [4] "KEKB Power Supply Interface Controller Module", A. Akiyama et al., in these proceedings.
- [5] "Operation of KEKB Linac and Ring with EPICS",M. Kaji and K. Furukawa, Proc. of ??th Linac Acceleration meeting Japan, 199?.
- [6] "SAD home page" at "http://www-acc-theory.kek.jp/SAD/sad.html"
- [7] "Performance Evaluation of EPICS on PowerPC", J. Odagiri et al., in these proceedings.
- [8] M.Lutz, "Programing Python", O'Reilly & Associates, Inc. USA, 1996
- [9] S. Wolfram, "Mathematica: A system for Doing mathematics by Computer", 2nd ed. Addison-Wesley Publishing Company, Inc. USA, 1991