

Status of the KEKB accelerator control system development

Tadahiko KATO, 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.

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

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

Abstract

KEKB, an asymmetric electron-positron collider, is in the last phase of its construction. The status of the development and installation of the control system for the KEKB will be reported in this paper.

Installation of the basic equipment for the KEKB accelerator control system, including FDDI network cables between the central control room and sub-control rooms, was finished in spring 1997. This basic system is now being used for the development of software for the KEKB accelerator control system. Extension of the system for the first commissioning is scheduled later this year.

The KEKB accelerator control system is based on the EPICS (Experimental Physics and Industrial Control System) tool kits for accelerator control system. EPICS tool kits uses the variant of so-called "standard model" of an accelerator control system. Use of EPICS tool kits reduces needs for the development of basic software.

Relational database is another key component in the KEKB accelerator control system. An end user interface using the WWW browser is developed and is under the field test.

Power supply controller board has been developed and tested. The board uses ARCnet as the communication interface to the VME controller module in the KEKB accelerator control system.

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.

As reported before[2], We use the EPICS [3] tool kit. The following section briefly summarizes architecture of the EPICS and related topics. The design of the KEKB accelerator control system network is described in Section 3.

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 reduce cost and space of cabling for 2,154 magnet power supplies of the KEKB accelerators. Development of the board is reported in Section 5.

Table 1
Main parameters of KEKB

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

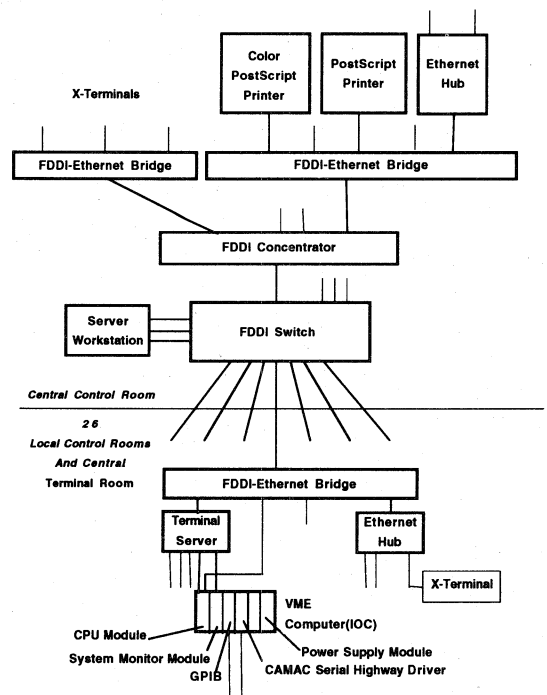


Fig. 1 Schematic view of the KEKB accelerator control system.

2 EPICS

EPICS is a collection of software tools developed by EPICS collaboration. EPICS assumes that the control system is 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 VxWorks real-time operating system is used as an IOC in EPICS. The IOC's are distributed along the accelerator.

EPICS run-time database on the IOC and CA constitute the core of the EPICS software. An 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) Scanner to scan the status of hardware in the way specified by the user, 2) 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.

User can also extend the EPICS core software using the hooks in itself. For example, user can add a support of 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) runs on the host computer. These 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 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 IOC's. X-terminals are used as an operator interface in the KEKB accelerator control system.

We currently use the 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. The gateway program to the LINAC control system is now being ported using the portable CA server program included in R3.13.

2.1 SAD-Python

MEDM and/or EDD/DM (Display Editor/Display Manager) are useful development tools for building basic control panels. 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[5]-EPICS interface is developed to fulfill this shortcoming. Python is an object oriented scripting language developed by G. van Rossum. Clear yet powerful syntax of Python is suitable for rapid

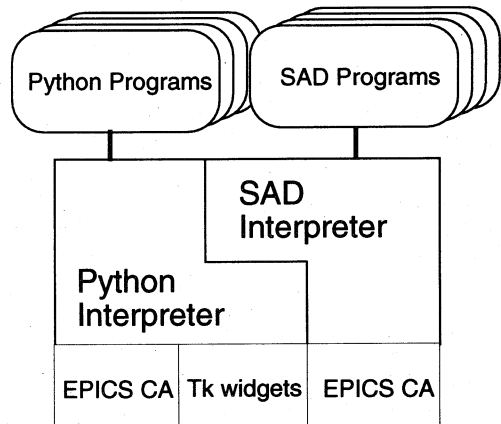


Fig. 2 SAD/Python relation. SAD embeds the Python interpreter in it. SAD calls Python to use a Tk widget. Both SAD and Python has EPICS Channel Access interface for the historical reason.

application development. A Tk widget is integrated with Python as a Python object. It allows 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[4] code allows a SAD programmer to develop a program with GUI as a SAD program. Figure 2 shows the relation of SAD, Python and EPICS-CA in current implementation.

2.2 Power PC

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

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 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.

3 Network

The Figure 1. shows a schematic view of the KEKB accelerator control system. As shown in the figure, the network consists of a switched FDDI network as a backbone network. An FDDI/Ethernet bridge bridges the FDDI backbone network and an Ethernet network in a sub-control room. Ethernet part of the bridge works as a switching hub rather than simple bridge. This configuration reduces 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 the transition to the future network technology, such as 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 Ethernet and a serial port of the terminal server. This is useful to diagnose status of 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 abnormal status, it sends the alarm message through the serial link. The VME system monitor board resets VME backplane when it receives the reset command through this serial connection.

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

4 Relational Database

We will use the relational database 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 EPICS runtime database. Logical design of 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 WWW(World Wide Web) browser. Figure 3 shows an example of the displayed screen 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. Poor development environment and incompatibility between WWW browsers are current major problem in this approach but 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 magnet power supplies and the accelerator control system. We developed the power supply controller card which can be installed inside the magnet power supplies. It uses

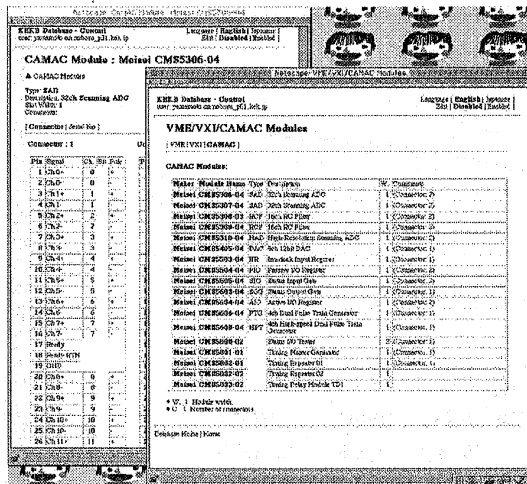


Fig. 3 Examples of the user interface to the ORACLE database using WWW browser.

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 were built and tested. The controller cards successfully controlled power supply units in the EPICS environment.

6 Conclusion

A basic set of the KEK 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] "SAD home page" at "http://www-acc-theory.kek.jp/SAD/sad.html"
- [5] M.Lutz, "Programming Python", O'Reilly & Associates, Inc. USA, 1996
- [6] S. Wolfram, "Mathematica: A system for Doing mathematics by Computer", 2nd ed. Addison-Wesley Publishing Company, Inc. USA, 1991