PEFP CONTROL SYSTEM USING EPICS

HyunMi Choi , In.Seok Hong, YoungGi Song, YongSub Cho

Proton Engineering Frontier Project, Korea Atomic Energy Research Institute
{ hmchoi, ish, ygsong, choys }@kaeri.re.kr

ABSTRACT:

KAERI (Korea Atomic Energy Research Institute) has been performing the project named PEFP (proton engineering frontier project). PEFP has been performing the project of a high power proton accelerator. Control system for 20 MeV proton accelerating structure has developed. We use the EPICS(Experimental Physics and Industrial Control System) tool kit as a foundation of the control system. EPICS is adopted for control systems which have OPI(OPerator Interface) and IOC(Input Output Controller). We have performed the PEPF control system on SUN workstation host computer. In this paper, we present the vacuum monitor, RFQ, and DTL Turbo pump control system.

KEY WORDS: EPICS, control system

1. INTRODUCTION

The construction of PEFP project whose final objective is to build 100 Mev proton accelerators started in 2002 and expected to finish in 2012. In 2005, we have performed 20mA proton beam of 20Mev. For developing the control systems of the 20Mev accelerator as well as 100 Mev accelerator, we chose EPICS(Experimental Physics and Industrial Control System)[1] as the most suitable tool. We have studied EPICS applications for various situation and as the application we developed vacuum control system using EPICS base3.14.4 as the core software and EPICS extensions (e.g., EDM(Extensible Display Manager), MEDM(Motif Editor and Display Manager) etc.)[2] As the user interface.

The EPICS is in use at a number of physics projects in North America, Europe and Asia that vary in size, specification, platform and requirements. For both the SNS(Spallation Neutron Source) and the SLS(Swiss Light Source), many developments have been make to the EPICS core software, the configuration tools, and the operator tools.

There are a number of projects using EPICS for a broad spectrum of applications. EPICS began as a collaboration between Argonne National Laboratory and Los Alamos National Laboratory in 1991, building on work that was initially done at the ground test Accelerator[3]. It is now running on accelerators that have as many as 180 distributed front-end controllers and control rooms with 20 consoles and a gateway to make system parameters available to offices, web site, and other remote control stations. It is also used at single controller and one workstation systems.

We use the EPICS tool kit as a foundation of the control system. We developed a vacuum control system for using Ethernet Multi Serial Deice Severs on PEFP control system. The control system now shows stable and reliable characteristics enough to meet our control requirement. However, the control system is continuously being upgraded to accommodate additional control requirements such as vacuum device control.

*This work supported by the 21C Frontier R&D program of the MOST
hmchoi@kaeri.re.kr

2. RELATED WORK

2.1 Epics System Architecture

The EPICS architecture is standard 3-layer architecture[4]. It consists of front-end computers usually running a real-time operating system. The front-end computers communicate to field I/O through many different field buses. The operator consoles communicate to field buses.

The support for operator consoles and EPICS developers exists on most flavour of UNIX and Windows. Most of the client applications run in both environments. The front-end computers usually run vxWorks and are supported on PowerPC, X86 architectures. The gateway is currently operating in Solaris and Linux.

Field buses supported from the vxWorks environment are extensive and include: Controlnet, PCI , CAN-bus(Controller Area Network), Industry Pack, VME(Versa Module Eurocard), GPIB, Bitbus, Modbus+ and Ether IP.

In EPICS release 3.14.x, the front-end software was ported into LINUX, Solaris, windows, RTEMS(Real-time Executive for Multiprocessor Systems), etc.

2.2 Epics Core Software

In nearly every project, extensions are made to EPICS. Some laboratories develop new hardware interfaces or new client programs[3]. Some develop new configuration tools. To facilitate these activities, the approach has been to unbundled all of the EPICS software and give very close management to the communication protocol, Channel Access(CA) the Process Database, and the build procedures. These make up the Core of EPICS.

2.3 Epics Extensions

Since physics applications have a variety of challenging requirements and physicists and programmers have an array of favorite approaches, EPICS provides interfaces for extensions at all levels. .

MEDM[2] stands for Motif Editor and Display Manager. It is a graphical user interface (GUI) for designing and implementing control screens, called displays, that consist of a collection of graphical objects that display and/or change the values of EPICS process variables. The supported objects include buttons, meters, sliders, text displays/entries, and graphs. It has two modes of operation, EDIT and EXECUTE. Displays are created and edited in EDIT mode, and they are run in EXECUTE mode. Besides these, the EPICS extensions have ADT(Array Display Tool), AR(Data Archiver), BURT(Backup and Restore Tool), HistTool, and so on.

The Extensible Display Manager[2] provides screen management, file I/O, dynamic colors, and a dynamic data interface. It allows extensions in two important areas: new data sources and new display widgets. Although there are several other display managers in use, EDM will likely become the engineer's display platform of choice.

3. PEFP CONTROL SYSTEM USING EPICS

EPICS run-time database on the IOC and CA constitutes the core of the EPICS software. EPICS client software accesses the record in this run-time database using the logical name of the record. Read and write accesses to the EPICS run-time database form 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 supports routines. Epics core soft 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 requests from the client program. Configuration files prepared by the user define the actual behaviours of these programs.

The figure1 has showed such process.

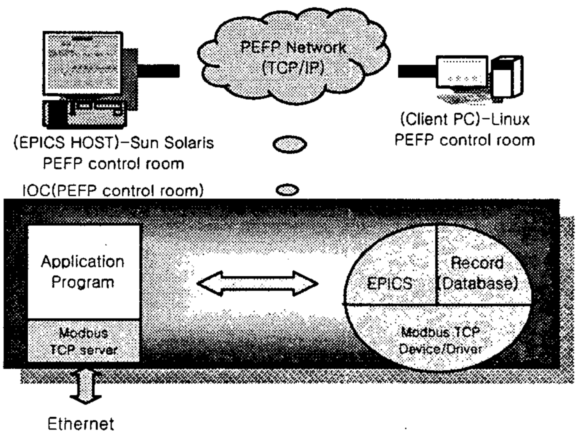

Figure1. Structure of PEFP control software

The EPICS BASE R3.13.x runs the IOC(Input Output Controller) on the vxWorks realtime operation system, but the EPICS BASE R3.14.x supports the multiple operation system. The PEFP control system is considered the stability of control system. Therefore we developed the control system using EPICS on Sun Solaris workstation.

The EPICS is begun the UNIX environment system. We are begun also the UNIX environment system. If any system installs the EPICS on Solaris system, it must make the EPICS path. The following commands create an example application.

```
mkdir <top>
cd <top>
<base>/bin/makeBaseApp.pl -t example example
<base>/bin/makeBaseApp.pl -i -t example example
```

If EPICS Extensions installs on Solaris system, it must compile the extensions config files (for base R3.13.x) and the extensions configure files(for base R3.14.x). Also it must modify the RELEASE file under config directory. For example, BASE path is <top>/usr/local/epics/base.

EPICS_BASE=/usr/local/epics/base

We have installed MEDM, EDM of Extensions tool. It is installed under Extensions directory. The following figure is the EDM and MEDM screen.

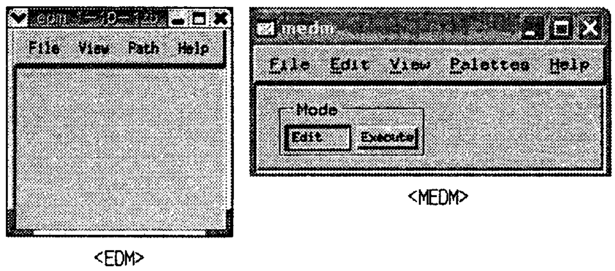

Figure2. Screen of EDM and MEDM

The two display tools have similar functions. We select the EDM tool. Because the EDM tool has more graphical capacity than the MEDM tool. It supports easily to understand and see on the control screen to operator.

The main objective of this control system is to allow accelerator operators to view and control all accelerator subsystems (e.g. vacuum, RF control, timing system, cooling device, etc)

In designing a control system for a large facility like Proton Accelerator, what is the most important and difficult to be controlled the each system. In addition to operating the machine, the control system should supply extra functions such as monitor and control the many control system. A large facility has sometimes even inconsistent kinds of hardware or software. In this situation, we should find a way to design a consistent control system which applies to all components and it is also safe, accurate and easily expandable.

The PEFP control system now shows stable and reliable characteristics enough to meet physicist requirements. The following figure is the vacuum monitor, RFQ, and DTL Turbo pump control. It is running on Solaris system for PEFP control.

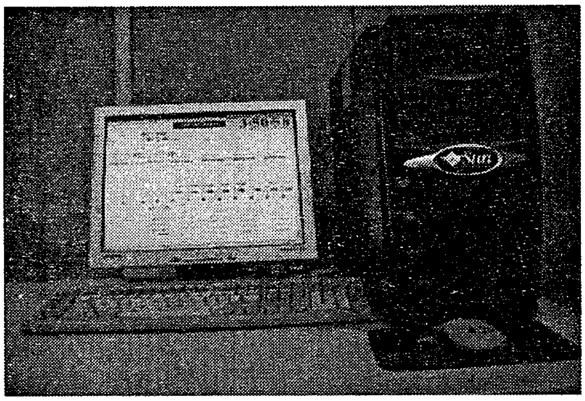

Figure3. PEFP control screen using EDM

4. CONCLSIONS

The vacuum monitor, RFQ, and DTL Turbo pump control system has installed in control room. The Software development/improvement continued until, long-term operational experiences even after it. The Designing EPICS database template for each hardware device will be the major effort in the software development. We are upgrading and modifying the control system to accommodate new control requirement and to apply long-term operational experiences.

REFERENCES

[1] Experimental Physics and Industrial Control System, http://www.aps.anl.gov/epics

[2] http://www.aps.anl.gov/epics/extensions

[3] L. R. Dalesio, 2001. EPICS: Recent Application and Future Directions. PAC, Chicago, pp. 276-278

[4] L. R. Dalesio, M. R. Kraimer, A. J. Kozubal, 1991. EPICS Architecture. ICALEPCS, Tsukuba Japan, pp. 278-282