Preliminary Design of the Control System for SSRF and BTCF

Jijiu Zhao , X.S. Geng, F. Zhang, T.J. Shen*
The Institute of High Energy Physics, Chinese Academy of Sciences
P.O.Box 918(10) Beijing 100039, P.R.China
*Institute of Nuclear Research, Shanghai, China

Abstract

The Shanghai Synchrotron Radiation Facility (SSRF) and the Beijing T-charm Factory (BTCF) are new high energy physics projects in China.The control division of IHEP undertook the feasibility research and conceptual design of the control system in 1996. This paper presents the conceptual design of the control system for both of the SSRF and BTCF.

The control system adopts a distributed architecture. High performance workstations with graphic user interface will be used as the console. A large number of VME/VXI based microprocessors and Programmable Logical Controllers (PLC) will be used as the front-ends to take the control jobs for the accelerator equipment. The high speed network and the field buses are considered in the device control layer. The software tool kits EPICS will be chosen to develop the control systems. In addition, some key technologies are also discussed in the paper.

1 Introdution

In recent years, the two new projects in the field of high energy physics were put forward by Chinese scientists. One is the construction of a new generation synchrotron radiation facility in Shanghai (SSRF) which includes the 100 MeV linac, transport line, 2.2 to 2.5 GeV booster and storage ring to serve the synchrotron radiation experiments. The other is the Beijing T-charm Factory (BTCF) , which is a new electron positron collider with the 1.5 to 3 Gev beam energy which will reach its luminosity as high as 1×10^{33} $cm^{-2}sec^{-1}$ at 2.0 Gev.The BTCF will be the best facility in the world for exploring T, Charm and charmonium physics. The R&D of SSRF will start in the near future.

The control system must be capable of (1) operating all of above facilities as machines with separate missions and controlling devices in central or local control rooms (2) controlling and monitoring all equipment of the accelerators including various magnet power supplies, RF cavities, vacuum units and different kinds of instruments for beam diagnostic systems. If something goes wrong in those devices, a reaction must be made by the system and an alarm report should be sent to the operating console at once and (3) providing the methods of accelerator commissioning and a friendly man-machine interface for operators.

Considering the above requirements,the control system should consist of the following parts:

(1) The computer control system
  The host and front-end computers
  The network system
  The man-machine interfaces
  The database management system
  The data logging and alarm report system
  Applications
(2) The timing system
(3) The interlock system
(4) The television and communication system

2 Design philosophy

In the 90's, the distributed processing system has been widely adopted in the world, therefore the control system will be a distributed control system based upon microprocessors which are linked by high speed networks. The subsystem control computers are loosely coupled for the convenience of independent operation at maintenance time. The standard and commercial hardware and software products should be applied in the system,and their cost-performance should be considered. The mature technologies of accelerator control systems in the world will be adopted.

The following points should be considered during the design stage: (1) Selection of computer systems according to the different usage of the console computers, central server and front-end computers. (2) Selection of the computer operating system. A real-time operating system should be chosen for the front-end computers. (3) Selection of high speed network to serve as data communication. (4) Selection of standard I/O modules, (5) Applying commercial software. The selection of hardware and software is important for developing and maintaining the control systems.

3 Distributed architecture

The control system is based on multi-level architecture of distributed processing system.[1] Logically the system is structured into three levels (as Fig1 shows), which are the operator interface layer, the front-end layer and the device controller layer.

The operator interface layer comprises operator consoles and server processors providing, the high-level program support, database support, shared files, network management and general computing resources. Console computers are a set of workstations and PCs which have a friendly graphic man-machine interface.

The front-end layer concentrates the control activities of sets or assemblies of equipment or entire subprocesses. It consists of several local control stations,so technicians can adjust and maintain these subsystems at the local control stations.

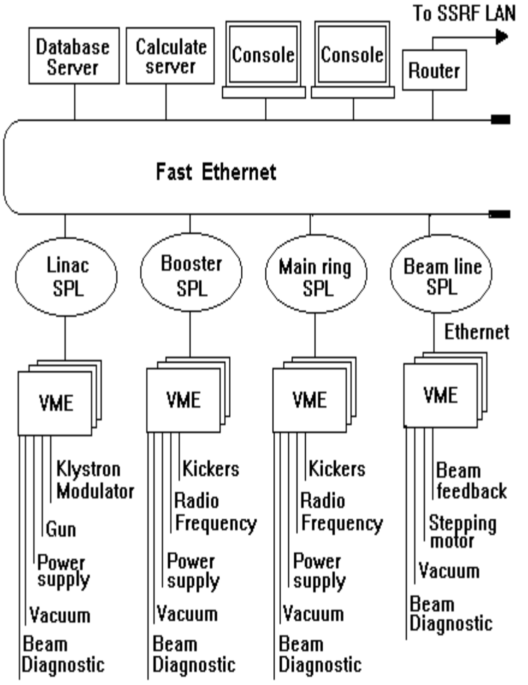

Figure1. Structure of the Control System

The front-end layer provides distributed proce-ssing power. A large number of microprocessors will be used as front-end computers to take the control jobs for the accelerator equipment and exchange the information with the central server computer through the network.

The device controller layer provides interface to the hardware, either as separate modulers or as microprocessors incorporated in the equipment to be controlled.[2] The Field buses will be used to connect the intelligent devices or some instruments exchanging data between front-ends and the device controllers.

The network links connect the control computers and the microprocessors serving data communication. This architecture has the advantage of localizing the hard real-time activities as near as possible to the equipment. It also provides the possibility to reduce the amounts of data that are transferred to the upper layer.

The control system software architecture is divided into the several parts as shown in Fig2.They are console manager, datebase server , network communication and front-end controls . In such a distributed structure, database plays an essential role as the repository of all information to which all parts of the control system refer. A real-time multi-task scheduler in the network environment will be developed to manage application processes running on different nodes.A numberof I/O driver routines and application programs for accelerator control should also be developed. EPICS will be chosen as the control system development toolkits.

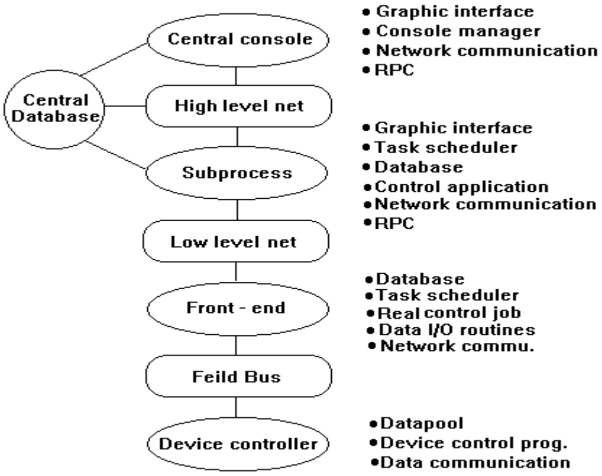

Figure 2. Software Structure of the Control System

4. Key technology

4.1 Computers and operating system

The selection of the computers and its operating system is important for system standardization.

The SUN, HP, or DEC, workstation with X-window, OSF/motif can be chosen to be used as the console, and some kinds of personal computer with windows, windows NT can be chosen for some sub_system controls. The central server computer and its operating system should be some mini-computers such as HP or, DEC Alpha computers. The UNIX operating system is recommended for the first choice.

For the front-end computer, the VME/VXI single board computers should be used for real-time control jobs with the VxWorks (or with Tornado interface) real-time operating system.The single-chip microcomputers could be used as device controller within the equipment.

4.2 Network

The console and server computer will be connected to the front-ends by a local area network link (LAN). As the backbone of the network, Fast-Ethernet can provide appropriately scaled access to the distributed equipment locations. Individual connections to FECs will be made via Ethernet. The TCP/IP network protocol has been selected. It provides the network file system NFS, network information service NIS, remote file sharing RFS and remote procedure call RPC.

4.3 Front-ends

All of the real-time tasks will run on the front-ends which have the capacity of parallel processing. All control and data acquisition transactions with the machine's equipment take place here. It translates a standard data representation in the control system to a suitable format for devices and device control modules. The raw data from devices are stored in the local database of the FECs. Only necessary data for the high layer will be sent out so that the overhead of the network transmission can be reduced.

The execution code running on front-end will be developed by a development system with UNIX and the VxWorks operating system. The VME products with MC680X0 or power PC microprocessor,VME/VXI bus and variety of VME I/O modules might be chosen as front-end units.The Programmable Logical Controller (PLC) will be used in the front-end layer to serve the digital and slow controls. (See Figure3)

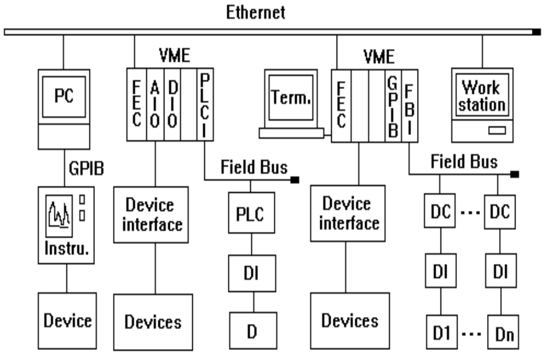

Figure 3. The Front-end and device comtroller

The field bus will be used for data communication between devices in the field and high layer FECs[3] . The following products can be chosen as a field bus: CANbus, Profibus, or lonworks.

4.4 Graphic man-machine interface

The console should have a friendly graphic man-machine interface which is easy to use. The console manager software will be based on X-window, OSF/motif, Medm of EPICS, TCL/TK, Labview, and ILOGview graphic packages. A graphic subroutine library will be provided for application programmers to draw pictures on the console.

4.5 Database

A distributed relational database, such as Sybase, is an important part of the control systems. With an Open client and SQL server of Sybase, much useful data with time stamps can be stored and analyzed. In recent years, the object-oriented database has been used in the field of accelerator controls, so we will pay attention to the development of the OODB.

4.6 Software tool kits

After system analysis, designing, modeling and simulating stages, the system can be constructed. There is a set of software tools and applications, Experimental Physics and Industrial Control System (EPICS), jointly developed by ANL and LANLof United States. The EPICS consists of OPI, IOC and CA.[4] With EPICS, the developers of the control system can generate a distribution database automatically, access I/O channels on the UNIX workstations conveniently, and create graphic man-machine interface easily. So it is better to develop the control systems by EPICS. Recently, a new application program interface CDEV of EPICS has been developed by Jefferson Lab, We will consider to use it in the two new projects.

5 Timing system

The timing system provides the synchronization of the accelerators. For the SSRF, on the one hand, a slow timing system generates a trigger sequence to synchronize electron gun, klystron, modulator in Linac, and septums, kickers in the Booster and Storage Ring, on the other hand, the fast timing signals generated by ECL circuits synchronize beam bunches sent by the electron gun with the bucket in the Booster or Storage Ring, so that the bunch can be injected into the corresponding buckets.

The trigger pulse generated from the RF frequency generator which is 500MHZ will be transmitted to the electron gun as the base of the synchronous clock. Trigger jitter should be less than 500ps.

6 Conclusion

The SSRF and BTCF are new generation of synchrotron radiation facility and collider so there will be some special requests for the control system. Up to now, some key technologies are not well known for us, so it is necessary to install a prototype system in the R&D stage to study the following issues:

(1) Developing the backbone of the control system with EPICS.
(2) Developing FECs,field buses and intelligent device controllers
(3) Designing various device interfaces boards

After that, a well-considered design can be proposed.

References

[1] A. Daneels Euro. Lab Part. Phy. 26 Sep 1994
[2] M.Milling, W. Busse Nuc. Ins. & Met. in Phys. Res.
[3] KEK B-Factory Design Report
[4] EPICS Release R3.12, APS Argonne