EXPERIENCE WITH THE NEW CONTROL SYSTEM OF IHEP ACCELERATORS COMPLEX

V. Voevodin, L. Kim, E. Klimenkov, V. Komarov, V. Kuznetsov

IHEP, Protvino, Russia

ABSTRACT

Over the years 1995 - 2004 the IHEP 70 GeV proton accelerator complex (startup 1967) has gradually been refurbished to a completely digitally controlled facility. Due to severe restrictions in personnel and budget, this had to be done slowly and in small steps. The paper analyses certain project decisions in view of experience gained in exploitation. Problems arising with continuing support of DEC stations and the migration to Linux PC servers are discussed. The saturated capacity of the data transmission channels and options for improvement are considered. In particular, the new type of equipment controller, based on PC, as well as the transition from VME crate with RTI to PC with the industrial MIL1553 Bus Controller, are being considered. For operator tools, it will be of advantage to replace the X-terminals by Linux PCs. We conclude with an overview of the status of these activities.

ARCHITECTURAL PROJECT DECISIONS

Hardware

We started to build the U-70 control system (CS) in accordance with preliminary plans and project decisions [1], [2]. The three levels architecture, so called “standard architecture , was adopted for the

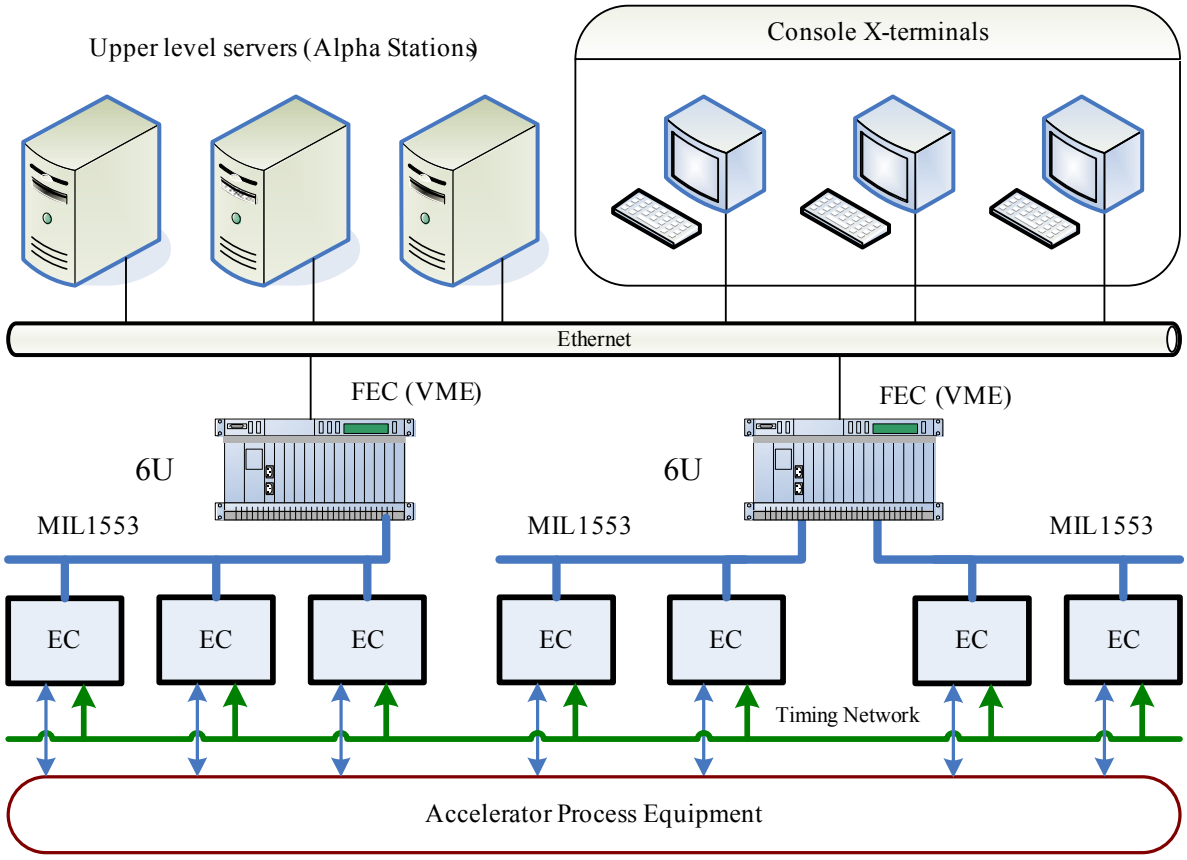

Figure 1: CS initial architecture

U-70 CS (Figure 1). The upper level consisted of two functional parts, namely Alpha Stations as specialized servers and X-terminals as user points to access the control system facilities. The Front End Computers (FEC) level was built on the base of VME crates with AUI Ethernet connector, Single Board Computers (SBC) of Motorola firm and one or two MIL 1553 Bus Controllers (RTI). The

Equipment Controllers (EC) level included Multibus-I crates housing necessary set of modules developed and produced at IHEP: SBC with Intel 80186 CPU, Timing Message Receiver (TMR), MIL1553 remote terminal and a set of various I/O modules. The TMR linked to the U-70 General Timing System [2], [3] ensured the execution of the EC real time tasks. The I/O modules were intended for organization of the interface with the accelerator process equipment.

The U-70 CS Project foresaw three categories of the interfacing means. The first one is the I/O module assembly tailored for the EC tasks execution. The second category is based on the CAMAC standard. In this case the process equipment communicates directly with modules plugged in the CAMAC crates. The EC controls the dedicated equipment through the Branch Driver placed in the Multibus-1 crate. The third category of the interface is organized in the following way. The specialized micro controllers (mC) are embedded in accelerator's equipment and linked with EC through RS-485 bus (Figure 2). For this purpose the RS-485 bus controller is plugged into Multibus-I crate. Now all that hardware is widely used to control numerous power supplies.

At the very beginning of the U-70 CS Project development the MIL 1553 standard link was foreseen as the only field bus. Now RS-485 multidrop bus is a second communication means. In both of them the ordinary twisted pare cables are applied. The same type of cable ensures the delivery of the coded timing signals to the equipment controllers. As for the Ethernet communications we use Fiber optic lines between remote buildings and, in general, coaxial cables inside the buildings.

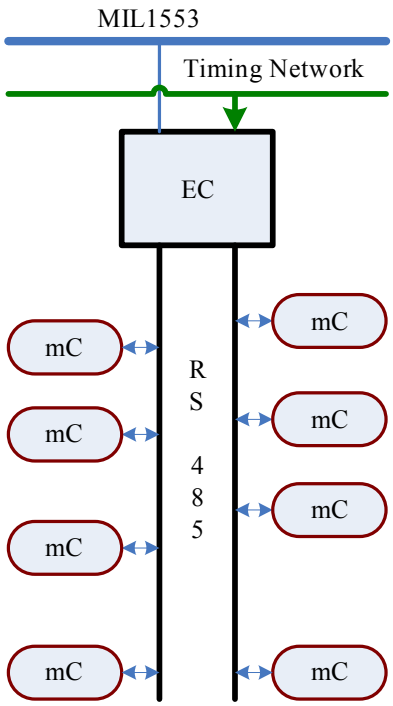

Figure 2: CS mC level

Software

The Alpha stations were equipped with DEC Unix and Red Hat Linux version 4.2 operating systems. The general software layout was described in [3], [4]. The upper level software consists of three main relatively independent parts: real time distributed databases (DDB), Graphical User Interface (GUI) and data processing applications (DPA). The GUI and DPA are pure data-driven software, their functionalities were completely based on the data and descriptions stored in DDB.

Since the start of U-70 CS exploitation the DPA software ran on three Alpha stations with Linux, GUI – on two Alpha stations with DEC Unix, DDB – on any computer with any system. The X Window, Motif and XRT packages lay in the basis of GUI, which realizes the wide set of functions with the DDB information.

The FEC level software runs under control of LynxOS operating system and performs transformation and transmission of packets between Ethernet and MIL 1553 communication fields. We use the maximum size of packet on the UDP protocol level up to 8K and only 256 bytes on the MIL 1553 level. There are the following specialized application protocols on top of UDP and TCP transport protocols to access EC services: EC application tasks loading, configuration files loading, DDB tables loading to EC memory, EC virtual terminal, access to DDB tables data in memory of EC.

CS EXPLOITATION PROBLEMS

The first problem arose as a result of the operational information growth. The intensified interaction between X terminals and the corresponding servers increased drastically the network load that often exceeded throughput threshold. In consequence the screen data renovation was slowed down and the queues of FEC/EC communication packets were temporarily blocked due to the traffic jam. In order to improve the situation (at least partially) the VME and X terminals network segments were separated. It allowed ensuring more stable execution of the VME control functions. The next step in that direction was the installation of fiber optic 100Mb Ethernet link between remote buildings. Finally we expect that X terminals replacement with PCs as well as sharing certain tasks between servers and operator's console PC will be efficient measure to solve the network-overloading problem.

Another difficulties were created by the field bus restrictions. As the control system embraced new technological subsystems of U-70 complex, the total number of equipment controllers was increasing. However the modest throughput of MIL 1553 arrangement limits the maximum number of equipment controllers connected to one multi-drop bus. It is obvious that addition of new MIL 1553 buses solves this problem. But insertion in the same VME crate additional RTI cannot significantly increase the field bus total capacitance since the FEC level software provides sequential service for all equipment controllers connected to the VME via MIL 1553 buses. Therefore for the efficient solution of the problem the additional Front End Computers are required. Realization of such a solution on the base of VME crates is very expensive whereas the application of PC as a FEC has to be much cheaper.

With the passing of the years both the hardware and software control means are getting obsolete that causes the other problems. In particular the maintenance of Alpha stations and X terminals becomes more difficult and expensive. Moreover at the present we already have equipment, which cannot be repaired at all. The same situation has been also created for the obsolete software losing the support of the producing firms. All that compels us to look for the new decisions. One of them is the gradual transition to the suitable for control system PCs and PC-compatible servers.

CS HARDWARE EVOLUTION

Today the market offers wide range of the personal computers with high quality screen monitors, which may be used as additional control system access points and substitution for X terminals in future. As a result we hope to diminish essentially the network loading and avoid the data traffic jams let alone the simplification of the maintenance problem.

As was already mentioned the attractive and not expensive way to increase the total number of MIL 1553 buses is to install sufficient number of PCs with industrial MIL 1553 bus controller. The separate PC can be used for each field bus thus ensuring the optimal throughput and best reactions of FEC/EC communication.

The great variety of PCI modules existing on a market allows solving some problems by means of using PC-compatible computers and fully industrial hardware [5], [6]. In order to support this new direction the PCI Timing Message Receiver (TMR-PCI) was developed. This module synchronizes the PC application software with the accelerators events being delivered by the U-70 General Timing System. In this case PC directly controls accelerator equipment and unites the functionalities of upper and EC levels of control system with direct access to Ethernet. Beam diagnostic subsystems with large volume of data are the first candidates to be modernized in this way. Besides that the PCs with RS-485 bus controller and Timing Message Receiver may help us to avoid the MIL 1553 bottleneck for some subsystems.

The migration to the fully PC-based Control System has been already very urgent when one of GUI servers failed during accelerator run and we could not manage to repair it. In order to keep afloat our controls the Web Alpha server was reinstalled as GUI server. Later on we used PC with Linux to restore Web server. It was done for the first time and then, during this year, three PC-compatible servers have replaced three Alpha stations running DPA under Linux control.

CS SOFTWARE EVOLUTION

Operating systems

We started with DEC Unix for software development platform and for two Alpha stations (GUI servers). The Linux version 4.2 was installed on the rest of computers used as DPA, NFS and some other servers. Later each new computer was provided with the current Red Hat Linux version. Now the contemporary releases of Red Hat free software operating systems are installed on the following computers:

- Linux 6.2 on DS-10 Alpha Server operating as NFS;
- Linux 7.2 on PC Web server showing the current beam parameters;
- Linux 7.3 on PC with MIL 1553 bus controller;
- Linux 9.0 on PC controlling TV cameras [6];
- Fedora 3 on PC for new Linux GUI development and on three PC -compatible servers that had already replaced the data processing Alpha stations.

For some PCI modules (GPIB [5], RS-485 bus controllers) there are drivers based on Linux kernel version 2.4. But for other store-bought modules (such as TV cameras controller [6]) and homemade TMR-PCI the Linux drivers did not exist. Thus, we were compelled to start the writing of Linux device drivers. At the present the drivers for the modules controlling TV cameras are working with kernel 2.4. The TMR-PCI has two modifications of driver, namely, for kernels 2.4 and 2.6. In the nearest time we'll try to transfer usable RS-485 bus controller driver from kernel 2.4 to kernel 2.6.

Distributed Data Bases

DDB management system (DDBMS) was written in C and adapted for all computers and operating systems being in use. Now fifteen DDB binary files are distributed over ten computers. Before to send data the DDBMS of source computer with certain processor transforms binary data to internal format of destination processor. The data stored in table form is distributed among DDB files in such a manner to optimize both access time and network load. The DDB contains all the necessary information to ensure the accelerators run and CS operation. Therefore the CS functionality growth causes the permanent evolution of the DDB. At the present the general functional parts of the data are:

- The last settings of all accelerators subsystems.
- The last results of the measurements in all accelerators subsystems.
- A library of previously saved accelerators regimes.
- Descriptions of parameters the CS is dealing with.
- Descriptions of ECs and topology of MIL 1553 buses.
- Descriptions of mCs and topology of RS-485 buses.
- Descriptions of data processing programs and indication of computers they should run.
- The information, which is necessary to run GUI.

If any new application program, parameter for measurement or control, EC, mC, computer have to be added or topology of MIL 1553, RS-485 field buses has to be changed the corresponding descriptions must be modified. Only after that the control system can access the new or changed element. For each new processor or new version of operating system used to store database and/or to have access to DDB, the corresponding parts of DDBMS should be adapted for the new CS components. Now DDBMS supports Alpha RISC, Intel and Motorola processors. It can run under control of DEC Unix, LynxOS and all versions of Red Hat Linux.

GUI

GUI consists of five programs written in C with the use of X Window, Motif and XRT libraries. The GUI is running under DEC Unix at two Alpha stations. Each X terminal of control system logs in one of these stations and starts the first GUI program. The CS users requirements growth causes the permanent evolution of GUI. At the present the GUI programs supports five main functionalities:

1. The password input and CS user identification. Users of CS are described in DDB with priorities they have for each accelerator to change settings.
2. Movements over the tree structure of menus to select accelerator object and operator task to perform. At the end points of tree the corresponding data processing programs are initiated in the corresponding computers.
3. List of predefined and described in DDB parameters is linked to each operator task corresponding to the end point of tree. Current values of these parameters are shown to operator in form of table. Measurements are renewed every super-cycle; settings may be edited in the table, processed by applications and sent to hardware.
4. Functional dependences of parameters presented in the table may be shown in shape of graphs or diagrams in real time. An operator defines the desirable parameters, form of graphical presentation, regime of output.
5. A data processing program can prepare specific data presentation for current task in form of picture file. The GUI program outputs this picture on operator demand in separate window on the screen and redraws it every super-cycle.

To migrate to fully Linux based control system the new GUI was developed and written in C with usage of gtk+ package. It consists of the same five parts with the same functionalities as the old GUI. However there are some extensions. For example, an operator at any time may interactively select and

combine in one table any parameters of any accelerators described in the control system. After this he can see their functional dependences in graphical form in real time.

Both GUI are in operation now. A PC, running Linux and GUI programs, may be installed in any place as additional point to access CS or replacement of X terminal with much less network load.

Data processing

Data processing applications are written in C and C++. Each program is described to be loaded to computer where the majority of its data are stored in databases. This software realizes the algorithms of accelerators control and processing of measurements. It prepares the picture files for GUI and Web server representing most important current beam parameters.

All DPA were transported from Alpha stations running under Linux 4.2 control, to PC-compatible servers running under Fedora 3 operating system. For carefully written programs this procedure does not require enormous manpower.

CONCLUSIONS

The control system is in operation from the end of 1997. The accelerators run time losses caused by CS faults were negligibly small [3]. An introduction of the new techniques and software versions was done so as to be transparent for users and to improve the CS exploitation characteristics. Taking into account all the circumstances described above we came to the conclusion that the U-70 CS must be transformed into the fully PC-based control system (Figure 3). Our goal is to get maximally

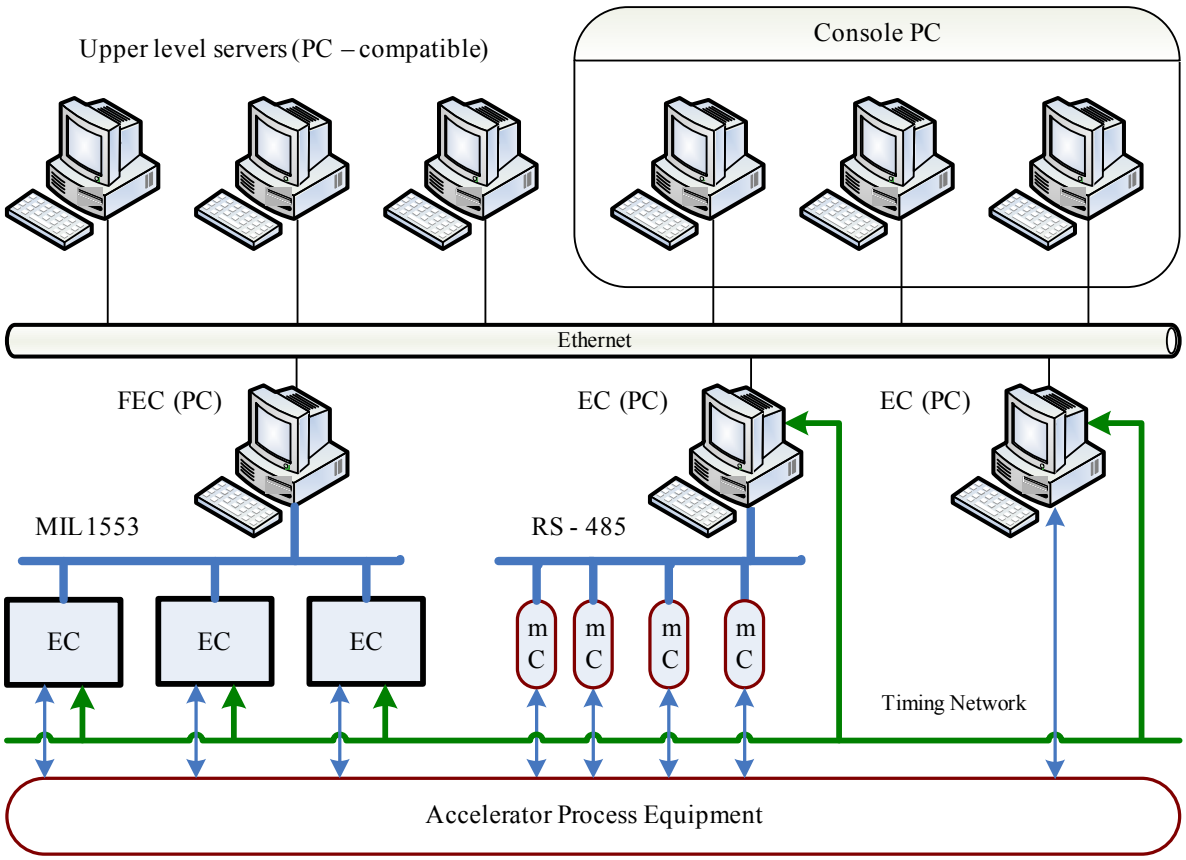

Figure 3: CS future architecture

homogeneous structure operating with Red Hat free software and home made software we are using now. The work has been already begun and we passed the first important steps in that direction.

In order to keep the major part of the expensive software developed for equipment controllers and microcontrollers all these components will be left on the lower level of the system. However the ECs dealing with the large volume of information will be based on the PC with the industrial modules and home made TMR-PCI. They will be directly linked to accelerator process equipment.

REFERENCES

[1] V.Komarov, A.Sytin, E.Trojanov, V.Voevodin, V.Yurpalov, "Upgrade of the U-70 complex controls , ICALEPCS'95, Chicago, Illinois, USA, 1995, V.2, p.930.

[2] V.Komarov, Yu.Milichenko, V.Voevodin, V.Yurpalov, "Draft Design Study for the Control System of the U-70 Complex , IHEP&CERN, PS/CO/Note 96-26, Switzerland, Geneve, 1996.

[3] V.Komarov, Yu.Milichenko, F.Perriollat, V.Voevodin, "New Integrated Control System of IHEP Accelerators Complex , Proceedings of the ICALEPCS'99, Trieste, Italy,1999,p.639.

[4] V.Voevodin, "Software Architecture of the U-70 Accelerator Complex New Control System , Proceedings of the ICALEPCS'99, Trieste, Italy,1999,p.457.

[5] E.Klimenkov, N.Mamuchashvili, Yu.Milichenko, V.Voevodin, "Distributed analog signal observation system based on PC's with Linux , The Third International Workshop on Personal Computer and Particle Accelerator Controls (PCaPAC-2000), Hamburg, Germany, 2000.

[6] E.Klimenkov, G.Losev, Yu.Milichenko, V.Voevodin, "Beam parameters measurement based on TV methods , XIX Russian Particle Accelerator Conference RUPAC–2004, Dubna, 2004 (http://accelconf.web.cern.ch/accelconf).