TUESDAY, 27 NOVEMBER 2001

TUA - STATUS REPORTS

TUAI001
physics/0111063

THE NATIONAL IGNITION FACILITY: STATUS AND PLANS FOR LASER FUSION AND HIGH-ENERGY-DENSITY EXPERIMENTAL STUDIES

E.I. Moses
LLNL, Livermore, CA 94550, USA

Abstract

The National Ignition Facility (NIF) currently under construction at the University of California Lawrence Livermore National Laboratory (LLNL) is a 192-beam, 1.8-megajoule, 500-terawatt, 351-nm laser for inertial confinement fusion (ICF) and high-energy-density experimental studies. NIF is being built by the Department of Energy and the National Nuclear Security Agency (NNSA) to provide an experimental test bed for the U.S. Stockpile Stewardship Program to ensure the country's nuclear deterrent without underground nuclear testing. The experimental program will encompass a wide range of physical phenomena from fusion energy production to materials science. Of the roughly 700 shots available per year, about 10% will be dedicated to basic science research. Laser hardware is modularized into line replaceable units (LRUs) such as deformable mirrors, amplifiers, and multi-function sensor packages that are operated by a distributed computer control system of nearly 60,000 control points. The supervisory control room presents facility-wide status and orchestrates experiments using operating parameters predicted by physics models. A network of several hundred front-end processors (FEPs) implements device control. The object-oriented software system is implemented in the Ada and Java languages and emphasizes CORBA distribution of reusable software objects. NIF is currently scheduled to provide first light in 2004 and will be completed in 2008.

1 INTRODUCTION

The NIF currently under construction at LLNL will be a U.S. Department of Energy and NNSA national center to study inertial confinement fusion and the physics of extreme energy densities and pressures. It will be a vital element of the NNSA Stockpile Stewardship Program (SSP), which ensures the reliability and safety of U.S. nuclear weapons without full-scale underground nuclear testing. The SSP will achieve this through a combination of above-ground test facilities and powerful computer simulations using the NNSA's Accelerated Scientific Computing Initiative (ASCI). In NIF, up to 192 extremely powerful laser beams will compress small fusion targets to conditions in which they will ignite and burn, liberating more energy than is required to initiate the fusion reactions. NIF experiments will allow the study of physical processes at temperatures approaching 100 million K and 100 billion times atmospheric pressure. These conditions exist naturally only in the interior of stars and in nuclear weapons explosions.

2 DESCRIPTION OF NIF

The NIF is shown schematically in Figure 1. NIF consists of four main elements: a laser system and optical components; the target chamber and its experimental systems; an environmentally controlled building housing the laser system and target area; and an integrated computer control system.

NIF's laser system features 192 high-power laser beams. Together, the laser beams will produce 1.8 million joules (approximately 500 trillion watts of power for 3 nanoseconds) of laser energy in the near-ultraviolet (351 nanometer wavelength). Currently the largest operating laser is the Omega Laser at the University of Rochester's Laboratory for Laser Energetics. Omega consists of 60 laser beams delivering a total of 40 kilojoules of energy. Figure 2 shows one of the 192 laser beams, detailing the key technologies that make NIF possible. A NIF laser beam begins with a very modest nanojoule energy pulse from the master oscillator, a diode-pumped fiber laser system that can provide a variety of pulse shapes suitable for a wide range of experiments, from ICF implosions to high-energy extended pulses for weapons effects experiments. The master oscillator pulse is shaped in time and smoothed in intensity and then transported to preamplifier modules (PAMs) for amplification and beam shaping. Each PAM first amplifies the pulse by a factor of one million (to a millijoule) and then boosts the pulse once again, this time to a maximum of 22 joules, by passing the beam four times through a flashlamp-pumped amplifier. There are total of 48 PAMs on NIF, each feeding a "quad" of four laser beams.

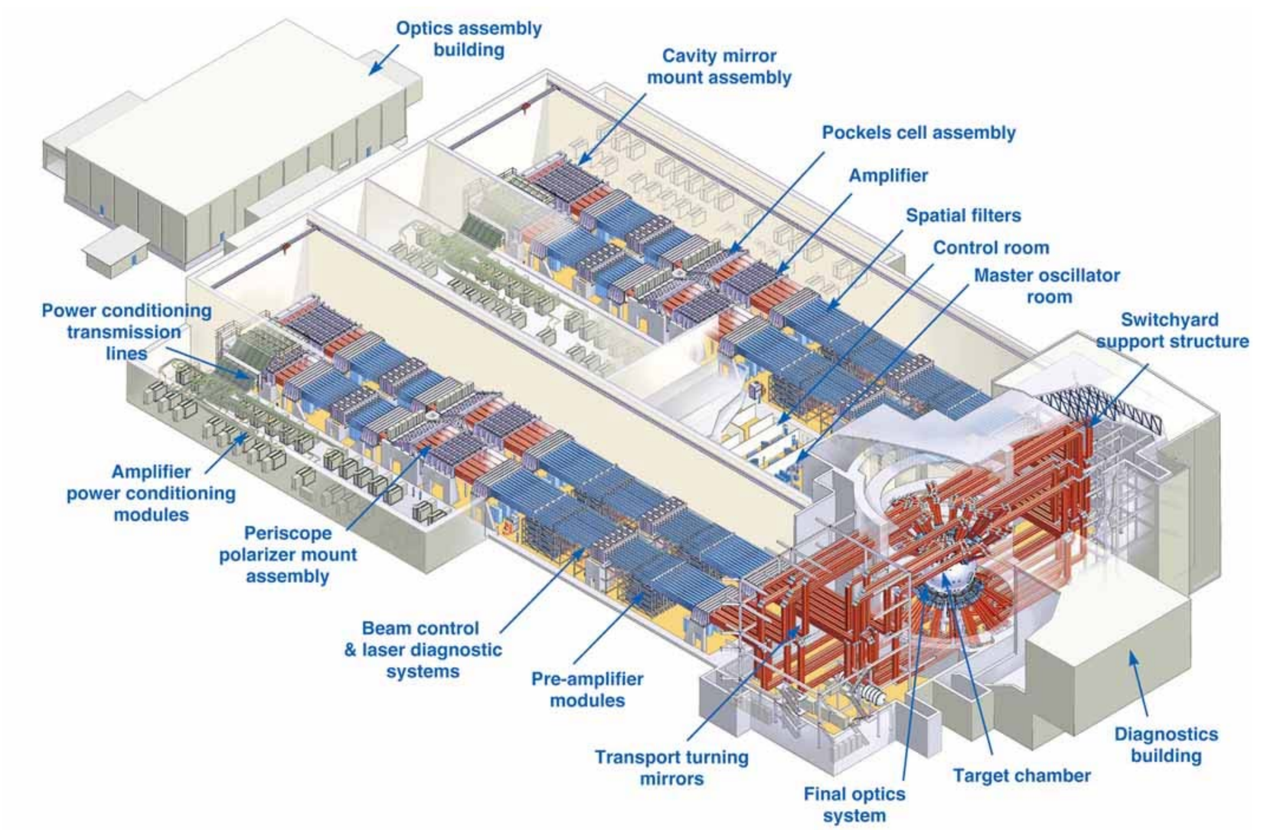


Figure 1: Schematic view of the National Ignition Facility showing the main elements of the laser system. The 10-meter diameter target chamber on the right side of the illustration sets the scale for the facility.

From the PAM, the laser beam next enters the main laser system, which consists of two large amplifier units—the power amplifier and the main amplifier. These amplifier systems are designed to efficiently amplify the nominal 1-joule input pulse from the PAM to the required power and energy, maintaining the input beam's spatial, spectral, and temporal characteristics. The amplifiers, with 16 glass slabs per beam, are arranged with 11 slabs in the main amplifier section and 5 slabs in the power amplifier section. Together these amplifiers provide 99.9% of NIF's energy. The amplifiers use 42-kilogram slabs, 46 cm × 81 cm × 3.4 cm, of neodymium-doped phosphate glass set vertically on edge at Brewster's angle to minimize reflective losses in the laser beam. The slabs are stacked four high and two wide to accommodate a "bundle" of eight laser beams (Figure 3).

The slabs are surrounded by vertical arrays of flashlamps measuring 180 cm in length. NIF's 192 laser beams require 7600 flashlamps and 3072 glass slabs. Each flashlamp is driven by 30,000 joules of electrical energy. The intense white light from the flashlamps excites the neodymium in the laser slabs to provide optical gain at the primary infrared wavelength of the laser. Some of the energy stored in the neodymium is released when the laser beam passes through the slab. The flashlamps and amplifier slabs will be cooled between shots using nitrogen gas. NIF will be able to shoot once every 8 hours; however, a shot rate enhancement program funded by collaborators from the United Kingdom is working to increase this rate to once every four hours.

The NIF amplifiers receive their power from the Power Conditioning System (PCS), which consists of the highest energy array of electrical capacitors ever assembled. The system's design is the result of collaboration between Sandia National Laboratories in Albuquerque, LLNL, and industry. The PCS will occupy four capacitor bays (Figure 1) adjacent to the laser bays. Each PCS module has eight 20-capacitor modules, delivering 1.7 megajoules per module, which power the flashlamps for one beam. The system must deliver over 300 megajoules of electrical energy to the flashlamp assemblies in each laser beam. Recent tests on a prototype PCS and flashlamp system have been fired over 7000 times at a rate of 1200 shots per month.

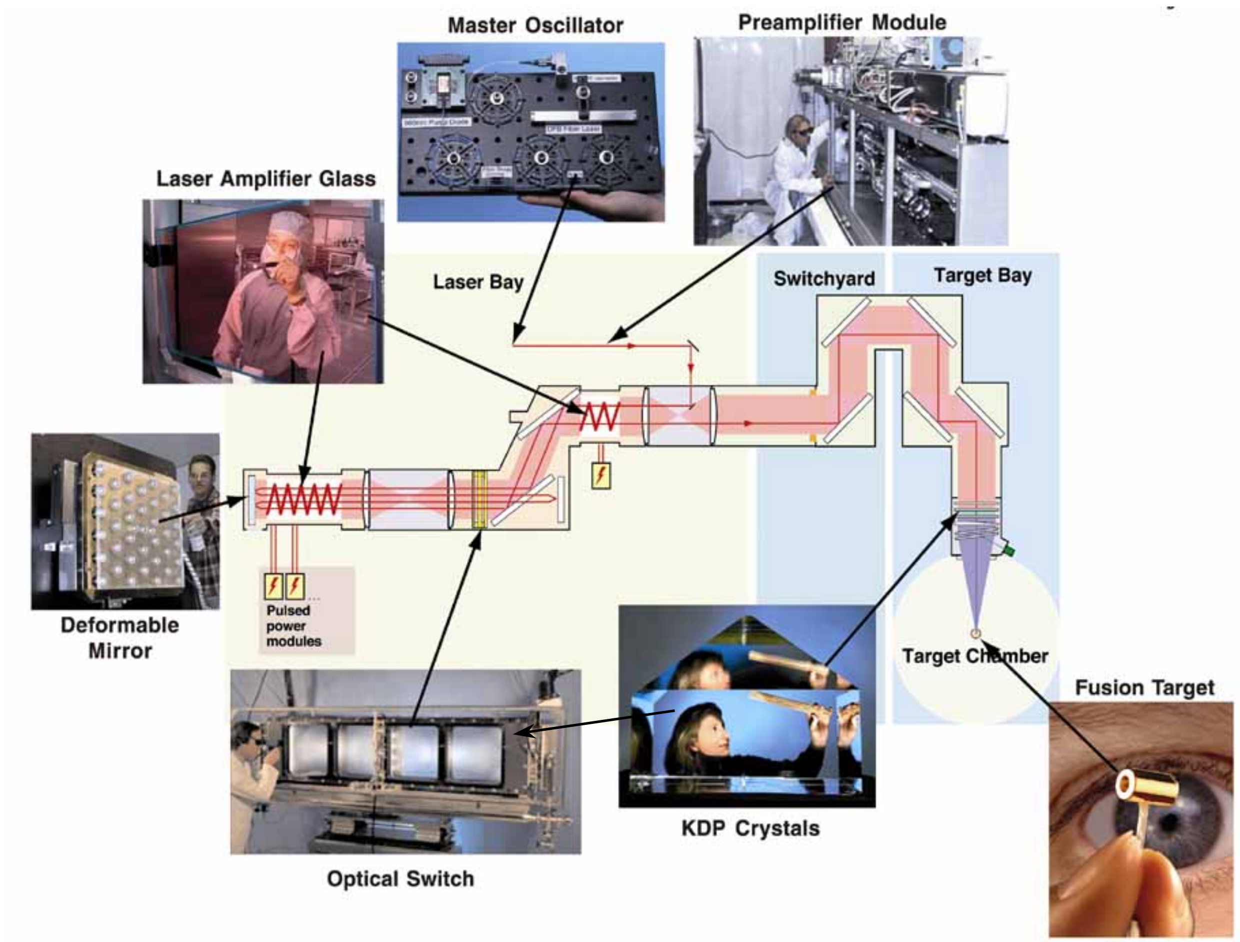


Figure 2: Schematic representation of a NIF laser beam line highlighting some of the key technology developments.

A key component in the laser chain is a kind of optical switch called a plasma electrode Pockels cell (PEPC), which allows the beam to pass four times through the main amplifier cavity. This device uses electrically induced changes in the refractive index of an electro-optic crystal, made of potassium dihydrogen phosphate (KDP). When combined with a polarizer, the PEPC allows light to pass through or reflect off the polarizer. The PEPC will essentially trap the laser light between two mirrors as it makes two round-trip passes through the main amplifier system before being switched out to continue its way to the target chamber. The PEPC consists of thin KDP plates sandwiched between two gas-discharge plasmas that are so tenuous that they have no effect on the laser beam passing through the cell. Nonetheless, the plasmas serve as conducting electrodes, allowing the entire surface of the thin crystal plate to charge electrically in about 100 nanoseconds so the beam can be switched efficiently. Figure 2 shows a prototype 4-cell PEPC (optical switch) that will be stacked vertically in a single unit called a line-replaceable unit (LRU).

All major laser components are assembled in clean modules called LRUs. These LRUs contain laser optics, mirrors, lenses, and hardware such as pinhole filter assemblies. All LRUs are designed to be assembled and installed into NIF's beampath infrastructure system, the exoskeleton of NIF, while retaining the high level of cleanliness required for proper laser operation. LLNL's industrial partner, Jacobs Facilities, Inc. is responsible for the installation, integration, and commissioning of the NIF laser beampath infrastructure in a way that ensures that the required cleanliness levels are maintained throughout the installation and commissioning phase of the Project.

The NIF target area consists of the 10-meter-diameter high-vacuum target chamber shown in Figure 4. The target chamber features a large number of laser entry ports as well as over 100 ports for diagnostic instrumentation and target insertion. Each laser entry port allows a quad of four laser beams to be focused to the center of the target chamber through a final optics assembly (FOA).

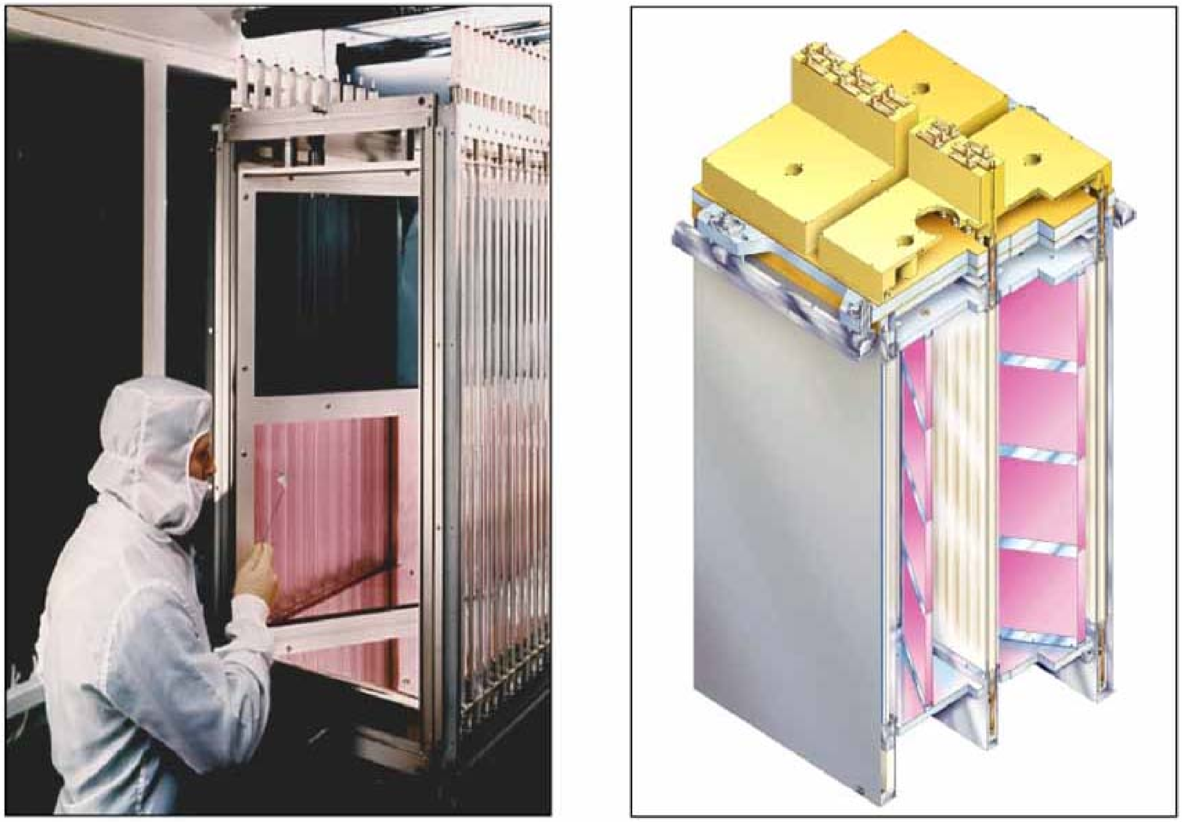


Figure 3. The photograph on the left shows an amplifier used on Beamlet, the scientific prototype of NIF. The illustration on the right shows the NIF 2 × 4 amplifier in cutaway view.

The FOA is a precision optical assembly containing beam smoothing gratings, additional KDP and deuterated KDP plates for second- and third-harmonic generation to convert the infrared laser light into the ultraviolet, the final focus lens, debris shields, and a vacuum gate valve for each beam. The NIF target chamber and final focusing system has been designed with maximum flexibility for experimental users. During initial operation, NIF is configured to operate in the "indirect-drive" configuration, which directs the laser beams into two cones in each of the upper and lower hemispheres of the target chamber. This configuration is optimized for illuminating a fusion capsule mounted inside a cylindrical hohlraum and using x-rays generated from the hot walls of the hohlraum to implode the capsule indirectly. NIF can also be configured in a "direct-drive" arrangement of beams by moving some quads of beams from the upper and lower beam cones into a more symmetric arrangement of beams. Direct-drive ignition requires better energy and power balance between laser beams and better beam smoothing and focusing, but the simpler geometry makes direct-drive inertial confinement fusion more attractive for ultimately producing a viable power production plant.

3 NIF CONTROL SYSTEMS

The Integrated Computer Control System (ICCS) for the NIF is a layered architecture of 300 FEP coordinated by supervisor subsystems. FEP computers incorporate either VxWorks on PowerPC or Solaris on UltraSPARC processors that interface to over 45,000 control points attached to VME-bus or PCI-bus crates respectively. Supervisory computers use Solaris workstations to implement coordination, database services, and user interfaces. Typical devices are stepping motors, transient digitizers, calorimeters, and photodiodes. The front-end implements an additional segment comprised of 14,000 control points for industrial controls including vacuum, argon, synthetic air, and safety interlocks using Allen-Bradley programmable logic controllers. The computer network uses Ethernet for control and status signals and is augmented with asynchronous transfer mode to deliver video streams from 500 sensor cameras within the laser to operator workstations. Software uses CORBA distribution to define a framework that incorporates services for archiving, machine configuration, graphical user interface, monitoring, event logging, scripting, alert management, and access control.

Software coding uses a mixed language environment of object-oriented Ada95 and Java. The code is one-third complete at over 300 thousand source lines.

4 NIF PROJECT STATUS

NIF is currently over four years into its construction. The conventional building construction is nearly complete. The attached 8000-square-foot Class-100 clean room Optics Assembly Building is undergoing commissioning of LRU assembly, handling, and transport equipment. Both large laser bays are operating under Class-100,000 clean room protocols. Over 1500 tons of beampath infrastructure have been installed in the laser bays. The NIF Project is entering the installation and commissioning phase. First light, which is defined as the first quad of four laser beams focused to target chamber center, is scheduled for June 2004. Full completion of all 192 laser beams is scheduled for September 2008. In the time between first light and project completion, approximately 1500 experiments in support of the SSP, inertial confinement fusion, high-energy-density physics, weapons effects, inertial fusion energy, and basic science will have been performed.

After project completion, NIF is expected to provide approximately 750 shots per year for a wide variety of experimental users. Recently, NIF was designated as a National User Facility with the support of the NNSA Office of Defense Programs. A National User Support Organization is being put in place to provide the necessary interface between the user communities and the national NIF Program. The first Director of NIF is Dr. George H. Miller, from LLNL, who also serves as the Associate Director for NIF Programs at LLNL.

5 CONCLUSIONS

The National Ignition Facility has come a long way since the first DOE critical decision in January 1993 affirmed the need for NIF and authorized the conceptual design process. In that time, NIF has met every scientific and technical challenge and is now in the final stages of design and construction prior to commencing installation of the 192 laser beams. By 2004 this unique facility will be providing the first glimpses under repeatable and well-characterized laboratory conditions of phenomena heretofore only found in the most extreme environments imaginable.

Figure 4: NIF’s 10-meter-diameter target chamber mounted in the target bay and viewed from below.

6 ACKNOWLEDGEMENTS

The author would like to express his appreciation for the many people, institutions, and industrial partners that are diligently working to provide the National Ignition Facility for our nation. This work was performed under the auspices of the U.S. Department of Energy by the University of California, Lawrence Livermore National Laboratory under contract W-7405-Eng-48.

REFERENCES

For more information on the NIF Project please visit our web site at http://www.llnl.gov/nif

TUAT002
physics/0111162

Status Report for the RHIC Control System*

J. Morris, T. D'Ottavio
BNL, Upton, NY 11973

Abstract

The Relativistic Heavy Ion Collider (RHIC) at Brookhaven has completed nearly two years of successful commissioning and operation. The control system is briefly reviewed and its contribution to the RHIC effort is analyzed, with emphasis on special challenges and innovative design: continuing efforts are also discussed.

1 INTRODUCTION

The Relativistic Heavy Ion Collider (RHIC) consists of two concentric rings with a circumference of 2.4 miles. The rings are made up of 1,740 superconducting magnets. The RHIC control system must address the special challenges posed by the size and character of RHIC. Precise beam control is needed to avoid quenches of cryogenic magnets and to provide adequate beam lifetime for stores of 4 hours or more. Control of settings for the widely distributed magnets must be well synchronized. Data acquisition from beam instrumentation must also be synchronized. Data capture and diagnostic tools are needed to support analysis of acceleration ramps and post mortem analysis of beam aborts and magnet quenches. During the past two years, the RHIC control system has continued to evolve to meet the demands of accelerator commissioning and operations.

2 SYSTEM OVERVIEW

2.1 Controls Hardware

The RHIC control system consists of two physical levels: console level computers and front-end computers (FECs). FECs provide access to accelerator equipment. FECs consist of a VME chassis with a single board computer, network connection, and I/O modules. Most FECs use the Motorola PowerPC™ processor. Commercially available VME I/O modules are used when possible. VME modules have been custom-designed for functions including power supply control, signal acquisition, and timing.

Console level computers are Sun™ workstations. One prototype Linux console has been used in the main control room during the 2001 RHIC run. Xterminals are also used as consoles in RHIC service buildings and other remote locations. The console level includes server machines that host processes for services such as data logging and alarming. Dedicated servers provide database and file services.

Console level computers and FECs are networked via Ethernet. A number of data links[1] provide synchronization of operations. The links are the Event Link, the Real Time Data Link[2], the Permit and Quench Interlock Links, and two Beam Synchronous Links.

2.2 Controls Software

Front-end computers (FECs) use the VxWorks™ real-time operating system. FEC software modules fall into two broad categories: device drivers and accelerator device objects (ADOs[3]). Device drivers, typically written in C, simply provide a software interface to the VME I/O modules. ADOs are C++ objects that are responsible for coordinating operation of the accelerator equipment and providing an interface for client programs. A code generation mechanism facilitates the development of ADO code and associated database information.

Console level software runs under the Solaris™ operating system. Console software has been primarily developed in the C++ language. Programmers rely on an extensive set of C++ class libraries. Members of both Controls and Physics groups develop RHIC applications. Many key physics applications have been written in Tool Command Language (TCL)[4].

The UI Toolkit is a large C++ class library that provides an object-oriented interface to X/Motif widgets and third party graphics and table tools. A C++ class interface is provided for access to Sybase database services. Assorted data storage classes are used to read/write Self Describing Data Set (SDDS)[5] files, often with associated database information to simplify searches for files of interest. C++ classes have been written to provide an application interface to accelerator device objects. A Common DEVice (CDEV)[6] interface is also offered. The CDEV interface is available to C++ programs but it also provides control system access for TCL programs.

Server processes can be built from the same ADO tools used on FECs and present the same ADO interface to application programs. Other servers are built in the CDEV generic server style.

3 SYSTEM HIGHLIGHTS

This section describes some of the aspects of RHIC controls that have been most significant for the current run.

3.1 Model Driven Ramp Control

Model driven control of magnet ramps[7] is a well established part of RHIC operations. The path up the ramp follows *stepstones*, which are points at which the optics of the accelerator is specified. Physical properties of the machine, like tune and chromaticity, can be specified at these *stepstones*. An on-line model server converts these properties to magnet strengths.

{*} Work supported by US Department of Energy

3.2 Sequencing

Over the last year, a significant effort was put into developing a software infrastructure that supports the sequencing of routine operational procedures[8]. This system is now used for a wide variety of purposes both within RHIC systems and the injector accelerator systems that feed RHIC. General areas of use include setting up and checking instrumentation and power supply systems, ramping RHIC through energy states, and system recovery from fault states.

The sequences themselves are written in a simple sequencer language developed at BNL. Each step in a sequence is either a primitive task (set a value, trigger an event, etc.) or a call to another sequence. The nesting capability enables reusability of sequences and permits more complex sequences to be constructed. No logic or parallel execution is currently allowed in the sequences, although these enhancements are being considered for a future version.

The sequencing software infrastructure includes two different GUI applications and two server programs. The two GUI programs service different sets of users, while the server programs allow sequences to be run by other applications and servers. The execution of all sequences is logged, allowing for easier diagnostics when problems arise.

The sequencing system has been heavily used over the last year and has lived up to its initial expectations: reproducible playback of procedures, improved execution times, and minimization of errors. To date, over 100 sequences have been created to handle routine operations of the RHIC complex.

3.3 Power Supply Diagnostics

Successful operation of RHIC depends on reliable and reproducible performance of more than 800 power supplies for ring magnets during acceleration ramps. A system was developed to capture detailed power supply measurements from RHIC ring power supplies during ramps. Diagnostic tools were developed to support analysis of this ramp data [9].

The ramp data sets provide a quantitative measure of the reproducibility of power supply performance. They are used to analyze power supply response when new modes of ramping are introduced and to identify power supply problems that may go undetected by ordinary alarm mechanisms. The ramp diagnostic system has now become a routine part of RHIC operations. Data is captured during every ramp and routinely analyzed by control room personnel. A watching mechanism has also been introduced to bring problems to the immediate attention of operators.

3.4 Coordinated Data Acquisition & Storage

In the fall of 2000, a project was initiated to improve the Control System data logging and retrieval capabilities. The goal of this project was to ensure that data needed for the analysis of RHIC could be captured and stored in a way that allowed easy retrieval and correlation of data from all systems. This project built on earlier work to establish data correlation[10] and logging infrastructure. New effort was focused in the following areas.

1) *Coordinated Acquisition* - acquire data from all necessary systems at appropriate rates and ensure that acquisition of data from different systems is synchronized to the level required for correlation. Standard events were defined to trigger data acquisition and systems were modified as necessary to conform to standard triggering methods. Since appropriate acquisition rates for some systems change during the course of a machine cycle (e.g. much more data is typically needed during ramps), the Sequencer was used to put these data acquisition triggers in the proper mode for each machine state.

2) *Data Storage* – ensure that adequate data and time stamp information is stored for all systems. Upgrades to the logging system were necessary to support some types of data and to maintain sufficient timing information for correlation. The common Logger format (based on SDDS) is used for almost all systems. Additional changes have been made to simplify access to data. Database headers, stored for each data file, are tagged with a *fill number*. A *fill* is defined as the period of time encompassing injection, acceleration, and storage of beam in RHIC. Files are also stored in a directory structure by *fill*, a procedure that has long been in use at CERN's LEP facility[11].

3) *Data Retrieval* - provide simple mechanisms for display of data from multiple systems with data selected by time period or *fill number*. Graphic display programs have been upgraded to add selection of data by *fill number* and to accommodate the display of array data in assorted formats. Additional enhancements include the ability to add event markers to graphs. The ability to plot data relative to an event allows the overlay of data from different fills.

The improvements have been extremely important to RHIC running in 2001. Data is being routinely logged for all systems. The viewing tools are routinely used to analyze machine performance.

3.5 Post Mortem

The RHIC Post Mortem (PM) system[12] was designed and developed to provide information about the state of the collider at the time of a beam abort, a quench of one of the RHIC superconducting magnets, or some other failure event that may cause beam to be lost from the machine. The data collected by this system helps to determine 1) the cause of the failure, 2) whether the machine is ready for another injection, and 3) how future stores might be improved.

Events on the RHIC event link (e.g. beam abort, quench) are the triggers that cause data, buffered for this purpose, to be read and stored by software systems that exist on both FECs and console-level computers. A GUI application then allows a user to view and filter the available data by system (e.g. power supply, beam loss monitor) and to select the data of interest, which is displayed graphically in a system-specific way.

PM data is currently gathered from the power supply, beam loss monitor, current transformer, quench detection and real-time data link systems. The PM system was used extensively as a diagnostic tool by the power supply/magnet groups throughout RHIC's first two years and continues to serve that purpose.

3.6 Tune Feedback

A tune feedback system [13] has been undergoing commissioning. Variations in measured tune are captured by a phase locked loop tune measurement system. Compensating changes in magnet strength are calculated in the tune measurement FEC and delivered to the Real Time Data Link (RTDL) via a reflective memory connection. The RTDL values are used to make appropriate adjustments in magnet power supply references.

4 PERFORMANCE/RELIABILITY

A significant and steady effort has been devoted to maintaining reliable operation of the control system during RHIC commissioning. Effort has also been directed at ensuring that failures are reported promptly by the alarm system and that recovery can be accomplished with minimal disruption to accelerator operation. During the first physics run in the summer of 2000, control system failures contributed less than 2% of RHIC downtime. The statistics are expected to be similar for 2001. A significant source of controls failures in 2001 has been radiation induced memory upsets in equipment alcoves located near the RHIC tunnel. Logging and post mortem servers have had a high degree of reliability but occasional failures have resulted in lost data. Graphic display performance is sometimes a bottleneck, particularly on Xterminal displays.

5 FUTURE

Final commissioning of the tune feedback system is anticipated before the end of the current RHIC run. To minimize recovery time after quench protection interlocks, software is being developed for automated analysis of post mortem data. Alcove radiation problems are being analyzed and solutions considered, including the relocation of some sensitive equipment.

Work to increase the reliability of logging and post mortem servers will continue. Work will be undertaken to improve graphic display performance. Disk storage space will be expanded to accommodate the demands of the post mortem, power supply diagnostic and beam instrumentation systems. The use of Linux consoles will be expanded in the Main Control Room. The Linux consoles are more economical than Sun workstations and may be outfitted with four monitors. A pilot project is underway to evaluate the use of the Java language for future software development.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the contributions of Don Barton, Jonathan Laster, Robert Michnoff, and Johannes van Zeijts.

REFERENCES

[1] B.R. Oerter, "Accelerator Timing at the Relativistic Heavy Ion Collider," Proc. ICALEPCS'99, Trieste, p.191

[2] H. Hartmann, "The RHIC Real Time Data Link System," Proc. 1997 IEEE Particle Accelerator Conf., Vancouver, p.2499

[3] L.T. Hoff, J.F. Skelly, "Accelerator Devices as Persistent Software Objects," Proc. ICALEPCS'93, Berlin, Nucl. Instr. and Meth. A 352(1994)185

[4] Ousterhout, J.K. ,"Tcl and the Tk Toolkit", Addison-Wesley Professional Computing, 1994

[5] M. Borland, "A Self Describing File Protocol for Simulation Integration and Shared Postprocessors," Proc. 1995 IEEE Particle Accelerator Conf., Dallas, p.2184

[6] J. Chen et al., "CDEV: An Object-Oriented Class Library for Developing Device Control Applications," Proc. ICALEPCS'95, Chicago, p.97

[7] J. van Zeijts, "Model Driven Ramp Control at RHIC", this conference

[8] T. D'Ottavio "Description of the RHIC Sequencer", this conference

[9] J. T. Morris, "Power Supply Ramp Diagnostics", this conference

[10] R. Michnoff et al., "RHIC Data Correlation Methodology," Proc. 1999 IEEE Particle Accelerator Conf., New York, p.693

[11] M. Lamont, CERN, private correspondence

[12] J.S.Laster, "Post Mortem System – Playback of the RHIC Collider", this conference

[13] J. van Zeitjs, "Tune Feedback at RHIC", this conference

STATUS OF THE SLS CONTROL SYSTEM

S. Hunt, M. Dach, M. Grunder, M. Heiniger, C. Higgs, M. Janousch, R. Kappeler, T. Korhonen, R. Krempaska, J. Krempasky, A. Luedeke, D. Maden, T. Pal, W.Portmann, H. Pruchova, D. Vermeulen, PSI, Villigen, Switzerland

Abstract

The Swiss Light Source is a high brightness synchrotron light source at the Paul Scherrer Institute in Switzerland. It consists of a 100 MeV electron Linac, a booster synchrotron, a 2.4 GeV storage ring, and experimental beam lines. The machine and beamline control system consist of 150 VME crates running Epics on Motorola power PC processors. The network is based on switched 100mbit/sec and Gigabit Ethernet technology. Consoles and servers are PCs running Linux. To achieve high availability of the control system, emphasis has been put on software engineering and the use of a relational database for all system configuration. Most hardware channels are directly connected to VME input/output cards rather than using a field-bus, and this has resulted in higher performance, better reliability, and reduced costs. Any of the 100,000 data channels can be archived at high speed, and the resulting data accessed through the Web. The VME input/output cards can be 'hot-swapped' in case of failure, and have circular buffers for post-mortem analysis in case of beam loss. Having all machine parameters available through the control system in a consistent and easy to use manner has contributed to the fast and successful commissioning of the machine.

1 SLS COMPONENTS

The Swiss Light Source is a third generation synchrotron light source at the Paul Scherrer Institute, Switzerland. It consists of a 100Mev electron Linac, a full energy booster synchrotron, a 2.4GeV 400mA storage ring, and initially four beam lines: A materials science beamline with a minigap wiggler; A protein crystallography beamline with a minigap in-vacuum undulator; a microscopy beamline with a permanent magnet undulator; and a spectroscopy beamline with an electromagnetic undulator.

2 STATUS

2.1 Schedule and Costs

The SLS came into operation this year on schedule and under budget. During the three year construction phase the building construction, technical system installation, and machine and beamline commissioning were completed.

2.2 Performance

The SLS has met or exceeded all of its design parameters. It is now running routinely in top-up mode where electrons are injected every few seconds to ensure a constant beam current. This provides very stable beam conditions for the experiments. Beam stability is very high with an RMS orbit error of less than one micron measured over a 17 hour time period.

2.3 Reliability

The SLS is running very reliably with most beam time now dedicated to user operations. Users consist of internal PSI research, external university groups and industry. The control system has contributed to less than four hours of lost beam time in the year since storage ring commissioning started. This was mainly due to configuration errors not hardware failures.

3 CONTROL SYSTEM ARCHITECTURE

3.1 Equipment interface level

Monitor and control of equipment is via the direct connection to 150 VME crates running Epics [1]. Standard I/O modules provide interfaces for analogue and digital input and output as well as motor control, temperature measurement, serial line connection, scaler modules and position encoders. Most interfaces are industry pack (IP) modules mounted on hot-swap VME 64X carrier boards. Connection to signals is on the rear of the crate using 80mm deep transition modules.

3.2 Network

The SLS controls network is a 100M bit switched Ethernet network, with some 1 Gbit connections. The network is isolated, with no routing to the rest of the institute or to the Internet. Some devices such as the file and database servers are on both the private and general network, allowing the exchange of data to office and central systems.

3.3 Operator interface level

The operator interface level consists of Linux PCs used as consoles and servers. Consoles in the control room have four screens each, and a number of single screen consoles, which also act as boot servers, are located in the technical gallery and on beam lines.

4 HIGHLIGHTS

4.1 Timing

The controls timing system [2] is a high performance design based on the APS timing system. Its high resolution and low jitter performance allows the accurate synchronization of hardware signals and software across the SLS control system. Its use simplifies the operation of the machine allowing complex sequences of events to be carried out by changing very few parameters. Its integration into Epics means timing parameters can be treated just like any other control system variable.

4.2 On line model server

The on-line model server and physics applications are provided by the beam dynamics group [3]. The model server can read the beam positions and actual magnet strengths form the control system and can very accurately predict the effects of proposed new settings before they are implemented. The very close agreement of the model and machine make it possible to achieve very good machine performance including beam stability.

4.3 Integration of Beamlines

Beamline controls [4] are handled by the same hardware and software used for machine controls. As well as reducing costs and development time this has enabled controls engineers to work on either system as priorities dictate.

4.4 Integration of sub-systems with turn-key controls

Both the Linac and the RF modulator systems were delivered by industry as turn key contracts [5]. These systems were delivered with an Epics control system making it simple to integrate into the global control system.

4.5 Moving physics applications and parameters into the control system

Calculation of some machine parameters has been moved from high-level applications into the low-level control system [6]. This provides more stability and higher performance. It also allows the use of our standard tools such as the archiver, alarm handler and save-restore tools.

4.6 Digital power supply control

Control of the 500 booster and storage ring magnets is carried out using individual fully digital power supplies. This has contributed to the very high stability of the electron beam. Interface to the VME crates from the power supply controller is via a custom designed optical serial link. All internal control parameters and readings can be read and set via this link and appear as standard Epics process variables.

4.7 Hot swap and post mortem analysis

Most of our I/O modules support the features of hot swap and have hardware buffers for post-mortem analysis following loss of beam or other events. The analogue input modules also support over-sampling of data to give 18-bit resolution and noise reduction by averaging.

4.8 Relational database

An oracle relational database is used for system configuration, and operational management. Features provided include generating configuration files (Archiving, Cdev, etc.), reporting of bugs and system failures, tracking the location of all hardware modules, and generating Epics substitution files. Users interrogate and modify database tables using a web interface.

5 REASONS FOR SUCCESS

5.1 Standardization

We have succeeded in standardizing on a small number of different hardware modules. The same hardware is used to monitor and control a large variety of devices. This has reduced development time and makes maintenance easier. The same version of software is loaded into all systems and regularly automatically updated. This includes low-level software (Epics system code and drivers) , as well as system code, application code, and configuration files on Linux servers and workstations. When a developer changes an application on request the changes are distributed to all systems.

5.2 Not building hardware in house

Where possible, commercial off the shelf modules have been used. Where such modules were not available to meet our requirements, contracts were placed with industry for design and production of the necessary modules. Contracts for design and production of VME and industry pack modules now include the provision of Epics drivers. This has further reduced the time for testing and integration into the control system.

5.3 No Fieldbus

By having direct connection using VME I/O rather than using a fieldbus level, the system is simpler, more robust and has higher performance. Cost per channel is low due to high signal densities, and maintenance and debugging are easier.

5.4 Not making major software developments

By largely using existing software components, we have been able to concentrate on solving the controls problems needed for each sub system. This has meant we were able to deliver working controls systems early in the project, and did not experience any major problems bringing the system into operation. Even when some tools did not provide all of the features we would have ideally liked, we have lived with a slightly reduced functionality, rather that embark on major developments.

REFERENCES

[1] L. Dalesio et al., “The Experimental Physics and Industrial Control System Architecture: Past, Present and Future”, ICALEPCS 1993, Berlin.

[2] T. Korhonen, M. Heiniger, “Timing System of the Swiss Light Source”, ICALEPCS 2001, San Jose.

[3] M. Boege, Jan Chrin, “On the use of Corba in High Level Software Applications at the SLS”, ICALEPCS 2001, San Jose.

[4] J. Krempasky et al, “ The SLS Beamlines Data Acquisition and Control System “,ICALEPCS 2001, San Jose.

[5] S. Hunt, “Purchasing Accelerator Subsystems as Turnkey Components”, ICALEPCS 2001, San Jose.

[6] A. Luedeke, “system integration of high level applications during commissioning of the Swiss Light Source”, ICALEPCS 2001, San Jose

TUAT004
cs.OH/0111020

Gemini MCAO Control System

C. Boyer, J. Sebag, B. Ellerbroek, Gemini Observatory, Hilo, HI 96720, USA

Abstract

The Gemini Observatory is planning to implement a Multi Conjugate Adaptive Optics (MCAO) System as a facility instrument for the Gemini-South telescope. The system will include 5 Laser Guide Stars, 3 Natural Guide Stars, and 3 Deformable mirrors optically conjugated at different altitudes to achieve near-uniform atmospheric compensation over a 1 arc minute square field of view [1,2,3,4]. The control of such a system will be split into 3 main functions: the control of the opto-mechanical assemblies of the whole system (including the Laser, the Beam Transfer Optics and the Adaptive Optics bench), the control of the Adaptive Optics System itself at a rate of 800FPS and the control of the safety system. The control of the Adaptive Optics System is the most critical in terms of real time performances. The control system will be an EPICS based system. In this paper, we will describe the requirements for the whole MCAO control system, preliminary designs for the control of the opto-mechanical devices and architecture options for the control of the Adaptive Optics system and the safety system.

1 OVERVIEW

As for conventional Laser Guide Star (LGS) Adaptive Optics (AO) system, the MCAO system will contain the six primary subsystems as follows: (1) the Laser System (LS), which includes all the elements necessary to produce the 5 laser guide stars; (2) the Beam Transfer Optics (BTO) used to transmit the laser light to the Laser Launch telescope; (3) the Laser Launch Telescope (LLT) located behind the secondary mirror at the top end of the Gemini Telescope; (4) the Adaptive Optics Module (AOM), which includes the deformable mirrors, the wave front sensors and the associated optical, mechanical, electrical components; (5) the Safe Aircraft Localization and Satellite Avoidance System (SALSA); (6) the Control System (CS), which will implement the Real Time Controller (RTC), that drives the deformable mirror based upon wavefront sensor measurements. This system also provides the supporting control functions such as opening/closing control loops, and will implement the control of all the opto-mechanical devices and loops of the BTO, LLT, AOM as well as the control of the SALSA system.

The MCAO CS controls the alignment, operation, and diagnostics of the whole MCAO system. It must manage a large number of opto-mechanical devices and still meet stringent real-time performance requirements. In order to do so the MCAO CS is split in 3 main functions: (i) the control of the Adaptive Optics System, this control is achieved by the RTC, which performs the real time wave front reconstruction and directly controls the deformable mirrors (DM) tip-tilt mirror (TTM), and the readout of the WFS components; (ii) the control of the SALSA system, (iii), the control of the all the opto-mechanical assemblies, which is implemented in 3 controllers:

- The Adaptive Optics Module Controller, which manages all of the opto-mechanical assemblies of the AOM except the deformable mirrors, the tip-tilt mirror and the readout of the WFS.
- The Beam Transfer / Laser Launch Telescope Controller which manages all of the opto-mechanical assemblies of the BTO and LLT.
- The Laser Controller, which manages the opto-mechanical assemblies of the laser itself (not described in this paper).

On top of this a sequencer component will be implemented to manage all the independent subsystems and act as the main public interface for the entire MCAO system. The sequencer will coordinate all of the internal tasks and provide external systems with the commands and status information they need to control the MCAO System.

The MCAO CS will be implemented using the standard Gemini Control System model. It will be a sub-system of the Observatory Control System and fully implemented as an EPICS system.

2 CONTROL OF THE AO SYSTEM

2.1 Requirements

This controller is dedicated to the Adaptive Optics control loop itself. It is the heart of the system and the most critical part in terms of real time performance. It will handle 3 basic real time functions: (i) the Natural Guide Star (NGS) real time control process; which controls the 3 tip/tilt sensors, the TTM and 3 DM anisoplanatism modes at a rate of 800Hz (Number of operations required: 3.16Mflops) (ii) the Laser Guide

Star (LGS) real time control process; which controls the 5 16x16 Shack Hartmann Wavefront Sensors (total of 2040 illuminated sub-apertures) and the 3 DMs (a total of 636 active actuators and 422 unactive actuators) at a rate of 800Hz (Number of operations required 2.26Gflops) (iii) the optimization and background processes, the goal of such processes is to continuously optimize or update the different parameters of the closed loop processes according to the atmospheric conditions for example, and also to provide data to outside components such as the MCAO BTO system.

2.2 Hardware solution

As discussed in the previous paragraph, the LGS control is the major user of CPU power, with a matrix multiplication being the most critical part. Performance at a level of ~3Gflops is required. Results of studies on current processors and architectures as well as benchmarks have both converged to a G4 PPC solution.

Such a solution is presented here. It is based on 4 VME Synergy quad PPC VSS4 boards (1 Synergy board per DM). The VSS4 board has one PMC (PCI Mezzanine Card) site, and using Synergy Micro Systems PEX3 PMC expander takes that site to yield three more, for a total of 3 PMC sites per VSS4 motherboard. To output the signals to the DM, a high speed parallel interface (PIO) board plugged directly into one of the PMC site of each quad G4 board will be used. To input the pixels, we will use a second high speed parallel interface board plugged into another PMC site. Each Synergy board will receive the pixels values, compute the centroids, perform the matrix multiplication that corresponds to its DM and send the actuator voltage through its daughter board directly. The last VSS4 board is dedicated to the optimization process, the control of the NGS process and the EPICS interface.

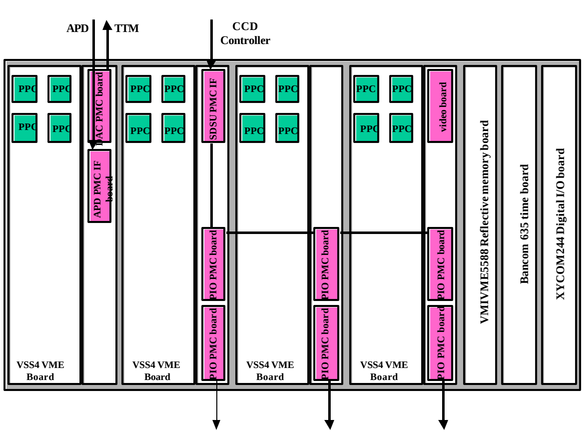


Figure1: RTC possible hardware architecture

The RTC will be developed outside Gemini. The RTC Vendor will define the final design of the RTC architecture.

3. CONTROL OF THE SALSA SYSTEM

The SALSA system consists of 3 main functions: (i) the Laser Traffic Control System (LTCS), for the beam collision avoidance; (ii) the aircraft detection; (iii) the satellite avoidance.

The LTCS will collect pointing information from all the telescopes located on the site, will determine in real time if collisions will occur and will close the safety shutter of the Laser Telescope in case of collision. The LTCS is developed jointly by the Keck Observatory and Gemini Observatory.

There will be 3 levels of aircraft detection:

- One infrared camera mounted on the telescope with a small field of view. This camera is boresited with the laser launch telescope and it only detects airplanes close to the laser beam. Upon detection, it automatically stops the laser by closing a fast shutter located in the beam (safety shutter).
- Two all-sky cameras working in the visible to provide complete sky coverage. Its purpose is to detect a moving airplane and to send a warning if it is coming toward the laser beam. The cameras will be located outside the telescope in weatherproof units and their signals will be sent to the telescope for analysis.
- Radar feed from local air traffic agency. This system would provide a display with airplane location using the radar data available from different local airports.

The aircraft detection controller will be developed by Gemini Observatory and is currently under design.

Lastly, for satellite avoidance, the plan is to use the US Space Command Laser Clearinghouse Program.

4. CONTROL OF THE OPTO-MECHANICAL ASSEMBLIES

3.1 BTO/LLT Controller

The Beam Transfer Optics (BTO) is the MCAO sub-system, which brings the 5 laser beams from the Laser System to the Laser Launch Telescope (LLT) mounted behind the telescope secondary mirror. The BTO/LLT Controller is responsible for managing all of the opto-mechanical devices associated with the BTO and LLT under the direct control of the MCAO sequencer. This controller is developed by the Gemini Observatory. Preliminary design is available on the web[6].

It will control a total of (a) 27 servo motors; (b) 3 AC motors; (c) 2 stepper motors; (d) 10 piezzo actuators; (e)

BTO diagnostic wavefront sensors (2 cameras); (f) 4 closed loops (one at 800Hz rate, the other ones at slower rates as 1Hz). The Controller will be a full EPICS/VxWorks system[7].

The following architecture has been defined to satisfy the BTO/LLT control and software requirements. The Controller will be implemented using two separate CPU boards. The first CPU board will provide the EPICS interface and all device control. This board will also implement the slow closed loop algorithms. The second CPU board (non EPICS) will be dedicated to the fast closed loop at 800Hz. The CPU boards will be single PPC boards from Motorola.

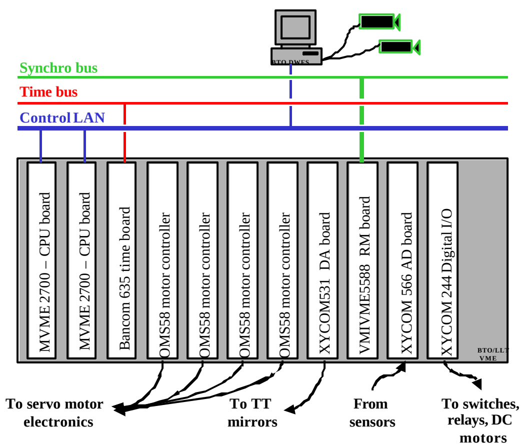


Figure 2: BTO/LLT Controller hardware architecture

3.2 AOM Controller

The Adaptive Optics Module (AOM) includes all of the optics, sensors and electronics needed to compensate the input f/16 science beam and relay it to a science instrument at f/33. These components include the principal elements of the real-time MCAO control loop: the 3 DMs, the TTM, 5 high-order LGS wavefront sensors, 3 tip-tilt NGS wavefront sensors. The AOM Controller is responsible for managing all of the opto-mechanical devices associated with the AOM under the direct control of the MCAO sequencer except the DMs, TTM and the readout of the NGS and LGS WFS. This controller is developed by the Gemini Observatory. Preliminary design is available on the web[6].

It will control a total of (a) 32 servo motors; (b) 9 DC motors; (c) 10 piezzo actuators; (d) 5 closed loops at slow rate (1Hz or 0.1Hz). The Controller will be a full EPICS/VxWorks system.

The following architecture has been defined to satisfy the AOM control and software requirements. The CPU board will be single PPC board from Motorola. This board will handle the control of all the devices and all the slow closed loops.

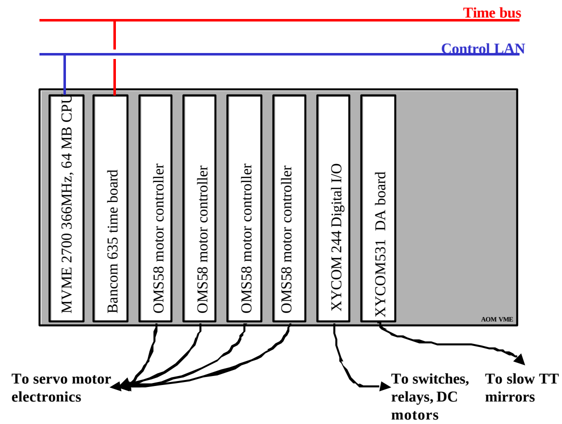


Figure3: AOM Controller hardware architecture

REFERENCES

[1] B.L. Ellerbroek and G. M. Cochran, "A Wave Optics Propagation Code for Multi-Conjugate Adaptive Optics". SPIE San Diego July 2001.

[2] Celine d'Orgeville, Brian Bauman, Jim Catone, Brent Ellerbroek, Don Gavel, Richard Buchroeder, "Gemini North and South Laser Guide Star Systems Requirements and Preliminary Designs". SPIE San Diego July 2001.

[3] B.L. Ellerbroek and F. Rigaut, "Astronomy: Optics adapt to the whole sky", *Nature*, Vol. 403, No 6765, p. 25 (Jan 2000)

[4] Francois Rigaut, Brent Ellerbroek and Ralf Flicker, "Principles, Limitations and Performance of Multi-Conjugate Adaptive Optics", in *Adaptive Optical Systems Technology*, SPIE Proc., Vol 4007, pp. 1022-1031 (2000)

[5] Jean Pierre Veran and Glenn Herriot "Centroid gain compensation in a Shack Hartmann adaptive optics system: implementation issue"

[6] "MCAO for Gemini South, Preliminary Design Review Documents", available on the Gemini Observatory web site: http://www.gemini.edu/sciops/instruments/adaptiveOptics/AOIndex.html

[7] M. Kraimer, EPICS Input/Output Controller (IOC) Application Developer's Guide, Argonne National Laboratory, July 1997, APS Release 3.13.0 beta11

[8] Veran J.P., Rigaut F., Maitre H. & Rouan D. 1997 J. Opt. Soc. America, A, 14, 11

TUESDAY, 27 NOVEMBER 2001

TUB - STATUS REPORTS

TUBI001
physics/0111077

An Overview of the LIGO Control and Data Acquisition System*

R. Bork, R. Abbott, D. Barker, J. Heefner, LIGO Laboratory,
California Institute of Technology, Pasadena, CA 91125, USA

Abstract

Interferometric gravitational wave antennas are based on Michelson interferometers whose sensitivity to small differential length changes (for LIGO it is 1 part in 10^{21}) has been enhanced by adding multiple coupled optical resonators. These antennas are used to verify the theory of relativity by detecting gravitational waves from astronomical sources such as spiraling binary stars and black holes. Estimates for detection rates of such events by LIGO range from several events per month to several per year. The LIGO Control and Data system (CDS) features a tightly coupled and highly integrated control and data acquisition system. Control of the interferometers requires many Multiple Input Multiple Output (MIMO) control loops closed both locally and across the 4-kilometer interferometer arm lengths. In addition to providing the closed loop control, the control systems front end processors act as Data Collection Units (DCU) for the data acquisition system. Data collected by these front ends and the data acquisition system must be collected and time stamped to an accuracy of 1 microsecond and made available to on-line analysis tools such as the Global Diagnostics System (GDS)[1]. Data is also sent to the LIGO Data Analysis System (LDAS)[2] for long-term storage and off-line analysis. Data rates exceed 5 Mbytes per second per interferometer continuous. Connection between the various front end processors and the data acquisition system is achieved using fiber optic reflective memory networks. Both controls and data acquisition systems use VME hardware and VxWorks® operating systems. This paper will present an overview of the LIGO CDS and discuss key aspects of its design.

1 INTRODUCTION

The LIGO interferometers located in Hanford, Washington and Livingston, Louisiana are Michelson laser interferometers enhanced by multiple coupled optical resonators. These coupled optical resonators are 4-kilometer long Fabry-Perot cavities placed in each arm of the interferometer. The mirrors that form the cavities are suspended from a single loop of wire mounted inside suspension cages that are mounted on seismically isolated optical platforms within the LIGO vacuum system.

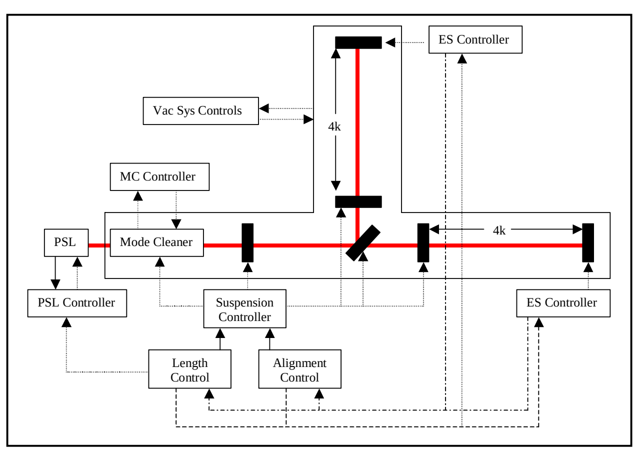
The Control and Data System (CDS) for these interferometers must provide many features. Among the requirements are:

- Provide for monitoring and control of the vacuum systems in which the interferometers reside.
- Provide for continuous data acquisition at rates to 5Mbyte/sec per interferometer.
- Provide for interferometer diagnostics to monitor interferometer systems and measure interferometer performance.
- Provide for local damping of all suspended optics.
- Provide for pitch and yaw (alignment) control of all suspended optics.
- Provide arm length control for four independent degrees of freedom. Each of these lengths must be controlled to an integral number of half-wavelengths of the laser light (λ = 1.06μm) with high accuracy, ranging from nm to less than 0.1pm.
- Provide the integration mechanisms to bring the system from a group of individually damped mirrors to a locked interferometer [4] with a strain sensitivity of 10^{-21}/•Hz.

2 CONTROL SYSTEM DESIGN

2.1 Overview

The following figure shows an overview of the interferometer and its control system.



* Work supported by the National Science Foundation under Grant PHY-920038.

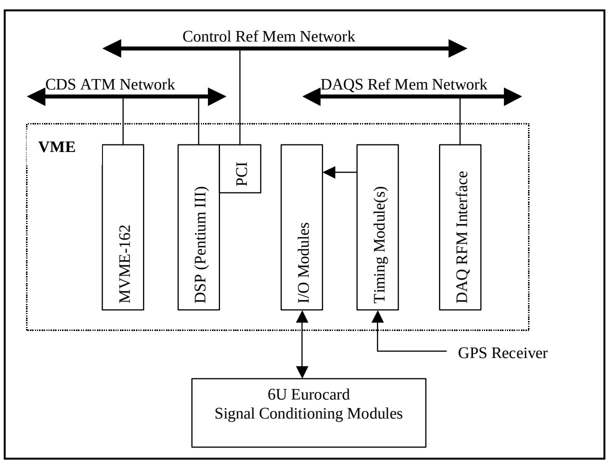
The interferometer itself consists of a pre-stabilized laser (PSL), a mode cleaner with seven suspended optics, and five large core optics. The core optics consist of a recycling mirror, beam splitter, two input test masses and two end test masses.

The LIGO laser is designed as a Non-Planar Ring Oscillator (NPRO) in a Master Oscillator Power Amplifier (MOPA) configuration. The laser was designed and built commercially to LIGO specifications. The unit operates at 10W, with an output wavelength of 1064 nm. To the laser system, LIGO CDS adds in-house designed frequency and power stabilization.

At the output of the PSL, a mode cleaner, mode matching telescopes and steering mirrors are provided to direct the beam into the interferometer. For these optics, as well as the core optics, CDS provides for local damping, alignment and length control.

2.2 Building Blocks

In general, the CDS front end controls are based on VME systems. To the extent possible, Commercial, Off-The-Shelf (COTS) VME modules are employed. Custom VME module designs are primarily limited to timing system interfaces. A basic block diagram of a typical VME subsystem is shown in the following figure.



Processors in the VME systems are of two types: 1) Motorola MVME162, for slow (<10Hz) control and operator communication applications and, 2) PentiumIII processors, for real-time Digital Signal Processing (DSP) applications (to 16KHz). In the case of the high bandwidth servo applications, both an MVME-162 and DSP will reside in the same VME crate. The MVME-162 performs the task of interfacing information between operator stations, via the CDS ATM network, and the DSP, via the VME backplane. The DSP only connects to the CDS ATM network for purposes of downloading software.

VME I/O modules are comprised primarily of Analog to Digital Converters (ADC), Digital to Analog Converters (DAC) and binary input/output modules. All ADC and DAC are simultaneous sampling 16 bit devices, with eight individual ADC and DAC per VME module.

Timing clocks for controls and data acquisition are derived from the Global Positioning System (GPS). A commercial GPS receiver was developed for LIGO that provides a 2^{22}Hz clock output, phase locked to the GPS 1Hz clock. LIGO custom timing boards to clock and synchronize all of the LIGO control systems and data acquisition system distribute the 1Hz and 2^{22}Hz clocks.

For interfacing to the various interferometer sensors and actuators, a large number of custom modules were developed in 6U Eurocard format for installation in 19” rack mount Eurocard chassis. Types of modules designed in-house include whitening/dewhitening filters, anti-aliasing/imaging filters, and I&Q demodulators. Several closed loop controls are also performed in analog circuitry or analog circuitry combined with digital controls. Examples include the pre-stabilized laser and mode cleaner controls.

For communications between the various processors involved in control and data acquisition, three networks are provided:

- CDS Asynchronous Transfer Mode (ATM) network. This network consists of an ATM switch with direct connection to operator stations and servers and uplinks to Ethernet switches at various locations for connection to the VME processors. This network is used for downloading code to VME processors and communicating information between operator stations and the VME processors.
- Data Acquisition Network. This network is based on 1Gbit/sec reflected memory, with up to 4Mbyte of memory per node.
- Control Reflected Memory (RFM) network. This network is based on 240Mbit/sec, 64Mbyte per node, reflected memory. This network is used for real-time communication between DSPs. The RFM interface modules directly connect to the DSP as PCI Mezzanine Cards (PMC).

Backend computers and servers are typically Sun Microsystems workstations and servers. These are employed as operator stations, system servers, file servers, and for interferometer diagnostic data analysis.

Software development for CDS is based on operational requirements. For all real-time DSP applications, code is developed in C and runs on the vxWorks® operating system on VME based PentiumIII processors.

For general network communications and slow controls, the Experimental Physics and Industrial Control System (EPICS), first developed at Los

Alamos National Laboratory, is used. The primary features of EPICS used in the LIGO CDS are channel access, used to communicate information between operators and the VME processors, the graphical user interface features, for development of operator displays, and the back up and restore tools.

For IFO diagnostic analysis and data display, code is developed in C++ for a ROOT based system originally developed at CERN.

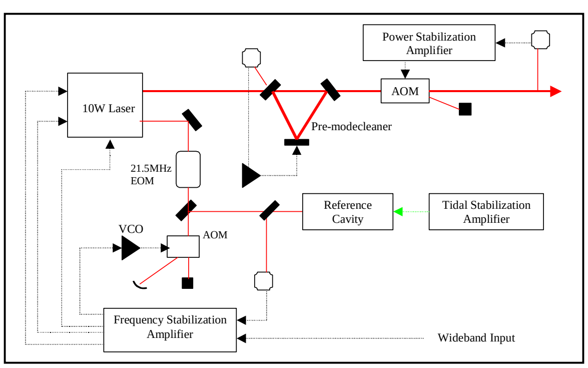
2.2 Vacuum Controls

The vacuum chambers, vacuum tubes, vacuum pumping and all associated sensors and actuators were designed, built and installed under contract by commercial vendors. LIGO provided the control and monitoring systems. This system was designed as a stand-alone system, with three 19" racks of equipment in the corner station and one in each mid and end station. The controls are VME based, using MVME-162 processor boards running EPICS software.

Actual monitoring and control is provided by software written using the EPICS State Notation Language (SNL) feature. For each type of device to be controlled and/or monitored, a generic EPICS database and SNL program was developed and tested, much like a C++ object. A PERL script was then used to generate the unique instances required by each VME processor in the system. A second script generates a startup file based on the contents of this directory. The file is loaded by the VME processor on power up or reset.

2.3 Laser Controls

The commercial laser system is refined in both amplitude and frequency stability by dedicated feedback controls. The entire system is referred to as the Pre-Stabilized Laser (PSL). An overview of the PSL is shown in the following figure.

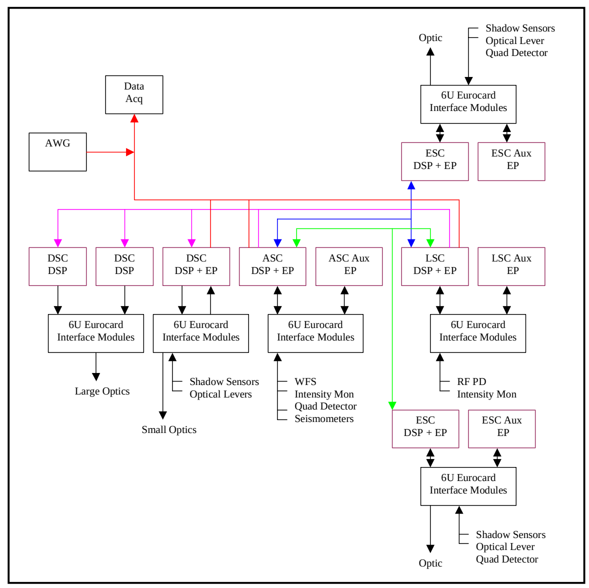


The frequency stabilization servo must initially reduce the intrinsic laser frequency noise by a factor of 1000. To meet the ultimate frequency noise requirements of LIGO, additional frequency stabilization is implemented by a "wideband" correction signal derived from the interferometer driving a dedicated PSL frequency adjusting input.

The amplitude stabilization servo was implemented to provide up to 80 dB of suppression of amplitude jitter inherent to the laser output. Both the frequency and amplitude stabilization are fully controllable through remote user interfaces. Additional amplitude stabilization and isolation of the fundamental optical mode is provided by the pre-modecleaner servo, which functions analogously to a tracking band-pass optical filter.

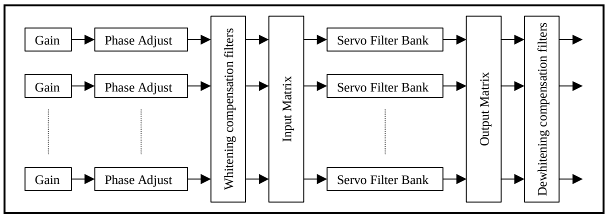
2.4 Interferometer Controls

The interferometer controls that provide for the local damping of optics (Digital Suspension Controls (DSC))[4], Alignment Sensing and Control (ASC) and interferometer Length Sensing and Control (LSC) are provided by MIMO digital servo controllers. All of these control subsystems share common design features.



Each of these digital control systems (DCS) consists of a VME crate with two CPUs, one DSP and one MVME-162 processor running EPICS (EP). The EPICS CPU has a database for communicating information to/from operators, with SNL code providing communications between the EPICS database records and the DSP. Essentially, this SNL code reads/writes local memory mapped to the VME backplane. The DSP updates this memory at 16Hz. The DSP closed loop control code runs at either 16384Hz or 2048Hz, dependent on the servo bandwidth requirements.

While the software for each DCS requires some custom software particular to the application, a fairly standard pattern for control is consistent in all units and code libraries have been written for these standard features. The standard software design is shown in the following figure.



As required, each input sensor signal is provided with a gain and phase shift block, settable by the operator. Some controllers require switchable hardware whitening filters. As these are switched in and out, software filters must also be switched to compensate for the hardware and maintain loop stability.

The input signals are then passed to an input matrix. The matrix parameters can be set by operators, or, as in the case of the LSC, this matrix can be automatically adjusted based on input parameters.

The actual servo calculation filter banks follow the input matrix. These filter banks contain as many as ten filters each. An output matrix and dewhitening compensation filters follow the filter bank. In some cases, the output matrix is actually a matrix of filters.

All of the digital filters in these systems are based on second order section (SOS) IIR filters. A filter bank consists of:

- A primary input, with gain and offset adjusts.
- An auxiliary input for insertion of excitation signals from the GDS.
- Up to 10 filter sections, each of which may be up to an 8^{th} order filter.
- An output inverter stage.
- An output enable/disable.

Initially, the filter coefficients were evolved from modeling of system requirements and then placed in code header files. As is typical, during interferometer commissioning it was found that the theoretical models and real world did not exactly match and the changing of filter coefficients became common. Therefore, the systems have evolved to writing the filter coefficients to separate files, which are user modifiable, and can be uploaded to the DCS. This is done in such a way that filters can be changed "on the fly", i.e. no task restart or CPU reboot is required.

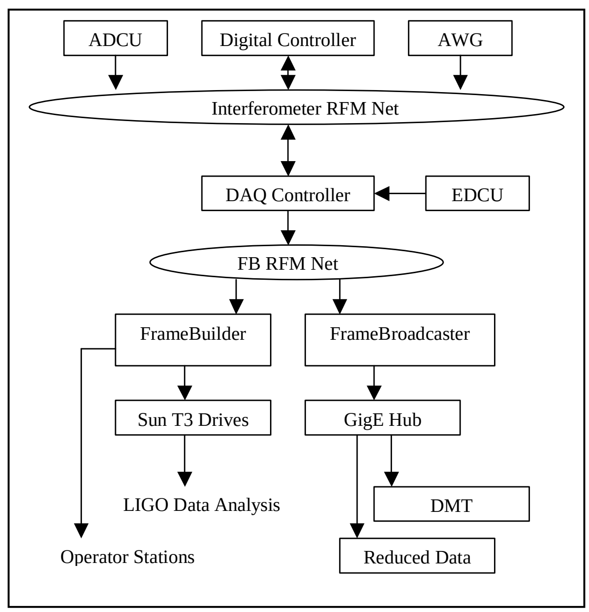
While the DSC, ASC and LSC are individual subsystems of CDS, they must communicate with each other in real-time. Certain sensors are connected to one subsystem, but are also required by another subsystem and control outputs of the ASC and LSC must be fed to the DSC for actual output to the optics. The sensors and control actuators may be up to 4-kilometers away from the DSP performing the servo control functions. To facilitate these communications, the CDS RFM control network is used. Each interferometer control system has three of these networks: one for communication within the corner station and one for communication with each end station. The use of three networks was required to minimize and balance the servo loop phase delay, as the speed to light to the end stations adds 13•sec.

3 DATA ACQUISITION

The LIGO Data Acquisition System (DAQS) provides several key functions:

- DAQS must acquire data from digital controllers and various analog signals from LIGO sensors and record this data to disk for consumption by the LIGO LDAS system for analysis.
- DAQS must broadcast acquired data to LIGO GDS computers for interferometer performance analysis and reduced data set writing.
- DAQS provides the network connections between the GDS Arbitrary Waveform Generator (AWG) and the digital controllers.
- DAQS provides for data transmission to operator stations and graphical tools to display the data.

An outline of the LIGO DAQS is shown in the following figure.



Input to the DAQS can be either as direct analog input signals, via the Analog Data Collection Units (ADCU) or as digital signals from the various LIGO digital servo controllers. These units are interconnected to each other and the DAQ controller via a 1Gbit/sec Reflected Memory (RFM) network. Each processor on the network is assigned a defined block for transferring its data and communicating control and status information with the DAQS controller.

ADCUs are VME crates with a processor, RFM network module, timing system connections, and up to five Analog to Digital Converter (ADC) modules. The ADC modules used in the system contain 32 individual, 24-bit sigma delta ADC with programmable gains. Signals are connected to the ADCU via an anti-aliasing chassis, which contains plug in filters appropriate for the data acquisition rate of a particular channel. Acquisition rates are selectable from 256Hz up to 16384Hz for each channel in steps of 2^n.

The DAQS RFM network also provides a connection for the Arbitrary Waveform Generator (AWG), provided as part of GDS. The AWG uses the DAQS RFM to inject excitation signals into defined points of the digital servo controllers or can directly produce analog outputs via its own Digital to Analog Converters (DAC).

All EPICS channels are available to be collected by the system at rates to 16Hz. As part of the DAQ controller VME crate, a second CPU, the EPICS Data Collection Unit (EDCU), collects requested EPICS channels and stores them in local memory. This memory is then read across the VME backplane by the DAQ controller for insertion into the main DAQS data stream.

The DAQ controller provides for configuration and monitoring of the various DAQS components and reformats data received from the various Data Collection Units (DCU) and transfers it on to the FrameBuilders. Configuration of the system is read from a text file on startup or operator request. This file contains information on all channels to be acquired, including connection points, acquisition rates, connection type, data storage requirements and conversion parameters to engineering units. Acquisition channels may be classified as either permanent connections, which are acquired and stored continuously, or as test points, which are only introduced into the system on operator request and are not stored.

The DAQ controller reads data from the interferometer RFM network and from the EDCU each 1/16 of a second. It reformats the data and then transfers it on to the FrameBuilder and FrameBroadcaster via a second RFM network.

The primary function of the FrameBuilder is to format the data into LIGO standard data frame format and then store that data to disk. The frame format is a standard adopted by the gravitational wave community for the sharing of data. A frame contains descriptor information on all of the data contained within a frame and data for all channels acquired in a given time period. At present, LIGO is producing frames that contain 16 seconds of data. Data written to disk are kept on line for up to three weeks before being archived by the LDAS.

In addition to these frames, which contain all data in raw format, the FrameBuilder also produces what are called trend frames. These are files that contain statistical information about the data channels, at one second, 1 minute and 1 hour intervals.

For delivery of data to operator workstations, the FrameBuilder also provides a Network Data Server (NDS). The NDS provides data on request via a CDS defined lightweight protocol. NDS is used by the DAQS data display software and GDS tools to retrieve information from the DAQS. Data is made available near real-time, at 16Hz update rates, or from LIGO data frames or trend files on disk.

The FrameBroadcaster produces one second data frames. However, rather than storing the data to disk, this unit broadcasts data in frame format over a gigabit Ethernet. This information is then received by several GDS interferometer diagnostic computers, which use the data to characterize various aspects of interferometer performance. The data broadcast is also received by the Reduced Data Set (RDS) writer. This unit records subsets of data and also has the ability to store test point data, which is not recorded by the main FrameBuilder. The RDS is typically used to record short stretches (hours) of data for off-line analysis in support of interferometer commissioning.

4 REFERENCES

[1] D. Sigg, R. Bork, J. Zweizig, “Detector Characterization and Global Diagnostics System of the Laser Interferometer Gravitational-wave Observatory”, Conference proceedings of the Ninth Marcel Grossman Meeting, July 2000.

[2] K. Blackburn, “Advances in Physics, Electronics, and Signal Processing Applications”, World Scientific and Engineering Society Press, 41-46 (2000).

[3] J. Heefner, R. Bork, “The LIGO Suspended Optic Digital Control System”, these proceedings.

[4] Peter Fritschel, Rolf Bork, Gabriela Gonzalez, Nergis Mavalvala, Dale Ouimette, Haisheng Rong, Daniel Sigg, Michael Zucker, "Readout and Control of a Power-Recycled Interferometric Gravitational-Wave Antenna," Appl. Opt. 40 (2001) 4988 - 4998.

TUBT002
physics/0111172

THE SLS BEAMLINES DATA ACQUISITION AND CONTROL SYSTEM

J.Krempaský, R.Krempaská , D.Vermeulen, D.Maden,
T.Korhonnen, W.Portmann, S.Hunt, R.Abela,
PSI-SLS, Villigen, Switzerland,
M.Muntwiler, Physik Institut, ETHZ, Switzerland

Abstract

In the first phase four beamlines have been constructed at the Swiss Light Source (SLS): two for the surface science community, one for powder and surface diffraction and computed micro-tomography, and the last one for protein crystallography. All of them are equipped with insertion devices, which users want to treat as beamline components like a monochromator or experimental station. The beamline control systems are based on the same hardware and software technology as is the machine. This implies extensive use of Personal Computers running Linux RedHat 6.2 and VME systems (PowerPC). The advantage of this choice is a staightforward implementation of the insertion devices into the beamline and experiment framework. Although the experiment Application Program Interfaces differ from beamline to beamline, the standard software technology for linking all sub-systems is based on the Epics toolkit and Cdev. The diagnostic tools provided by this toolkit are being extensively used during the beamline commissioning. Finally we describe some examples of integrating dedicated 3rd party and commercial non Epics software products for experiment control into the beamline control system. Key elements in this domain are CORBA, Python and EPICS Portable Channel Access Server.

1 INTRODUCTION

In the first phase SLS (Swiss Light Source) will have four beamlines. Two for hard X-rays (Protein Crystallography and Material Science) and two for the soft VUV region (Surfaces/Interfaces Spectroscopy and Surfaces/Interfaces Microscopy), all equipped with insertion devices. The commissioning of the first beamline started in April 2001 and by the end of November 2001 all of them delivered beam to their experimental environments. This very tight time schedule clearly indicates that the EPICS control system toolkit provided valuable help in the beamline commissioning. An important aspect of the SLS beamlines is the fact that they benefit from the same EPICS based control system as the SLS machine. This involves both hardware and software and allows smooth integration of several sub-systems (monochromator, insertion device, beam diagnostics etc ...).

2 DATA ACQUISITION AND CONTROL

The controls and data acquisition system of the SLS beamlines is closely integrated with the machine control system. It is based on EPICS, a client-server toolkit with CA (Channel Access) servers running on VME processors (PowerPC), and clients running Linux (RedHat 6.2) or NT PCs [1]. The position and functionality of the VME crates follow the beamline topology (typically front-end, monochomator, beamline optics, experimental station 1, 2...) and clearly determine a beamline sub-system (see Fig. 1). The total number of VME crates per beamline is 4-5. Some sub-systems have been (will be) delivered already with EPICS drivers, for example the Jenoptik Plane Grating Monochromator and COPHEE (COmplete PHotoEmission Experiment) experimental station. Their inclusion into the beamline EPICS environment is straightforward, which greatly facilitates and speeds up the beamline commissioning.

2.1 Usage of synApps package on the SLS Beamlines

The bulk of custom beamline control software is taken from synApps package [2]. It includes software support for motors, scalers, optical tables, slits, multidimensional scans, multichannel analyzers and miscellaneous devices. The support for motors (both stepping and servo motors) provided by the EPICS motor record, fits well with the Oregon Micro Systems VME58 family of intelligent motion controls, generally used for motion control on all beamlines. A monochromator can be considered as a collection of such motors. They can be driven according to special tables or analytic expressions defined by EPICS *calc* records. For example the control of the Kohzu double-crystal monochromator for the Protein crystallography beamline (also included in synApps) is a complex state notation program which drives the motors to required positions.

Another very useful support from synApps is the so called *sscan* EPICS record. It allows for a dynamic configuration of various types of scans based on EPICS process variables. This feature is of particular interest for beamline commissioning where one can perform a run-time configuration of 1D/2D scans with several positioners, detectors and detector triggers. The scan results can be automatically saved from the VME IOC to a file server - another very useful

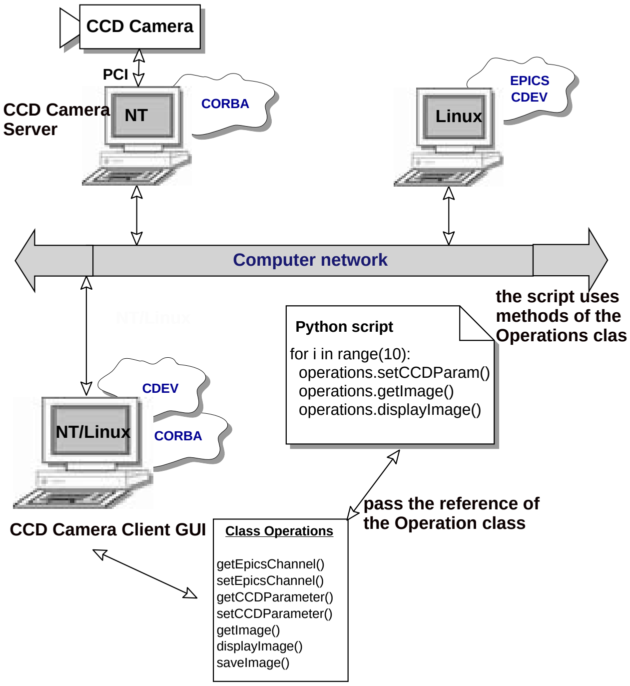


Figure 1: Schematical layout of a beamline experiment. The functionality of the sub-systems resides in VME crates seen on the lower part of the figure. On the upper part there are beamline clients doing the experiments. Since most of the functionality reside in VME crates (or virtual IOCs implemented with Portable Channel Access), an experiment may be implemented as a simple script of well defined operations.

feature called *saveData* from the synApps package (there is one dedicated file server for each beamline). Simply put, one can perform simple or sophisticated scans without involving coding on some clients. All data are kept in VME crates and saved automatically. The only limitation is the size of the *sscan* arrays, which is 16K.

2.2 Usage of SDDS on the SLS Beamlines

The Material Science Beamline at the SLS is currently making extensive use of the SDDS (Self Describing Data Set) [3] utilities and EPICS client-based applications developed at the APS for beamline tuning and data acquisition scans of the powder diffractometer. In particular, the GUI interfaces to the SDDS software, such as quickExperiment [4], have found acceptance by the beamline scientists during the early stages of beam commissioning. At a later date, it is intended that more science-oriented software will be introduced. This will also most likely be coupled with a move to NeXus [5], which is strongly supported by the neutron scattering community at PSI.

3 IMPLEMENTING NON-EPICS HARDWARE

Despite the fact that the experimental stations may be based on a non-EPICS hardware and software, there are several possible ways how to include them into an EPICS system. In this context we will briefly describe a CCD camera distributed architecture, and the EPICS implementation of the Scienta-Gammadata electron analyser for the Surfaces/Interfaces Spectroscopy beamline.

3.1 CCD camera distributed architecture

An experiment or diagnostics based on a CCD camera data acquisition and control consists typically of heterogenous hardware and software. In collaboration with ELETTRA, an open system has been developed, where different software components related to CCD data acquisition and evaluation, are integrated into a single GUI by means of CORBA. A CCD camera CORBA server implements all functionalities of the CCD camera and is running on a Windows NT operating system. The client provides a simple and intuitive graphical user interface giving a user the possibility to control the CCD camera by initializing, setting or getting CCD parameters, or displaying an image. The user interface provides also a utility for simple image processing such as adjusting contrast and brightness, creating a histogram of an image or performing arithmetic operations on images. It is implemented in Java. Typically an experiment based on CCD is a sequence of simple operations for the implementation of which a scripting language is welcome. The bulk of operations on CCD camera and EPICS control system is implemented in a Java class called "Operations" (see Fig. 2). By interpreting the script a reference to this class is passed to the script, which can call the class methods. In such a way a physicist can make an image-based experiment with understandable simple scripts.

3.2 Controlling the Scienta electron analyser

The heart of the experimental apparatus for the Surfaces/Interfaces Spectroscopy beamline is a Scienta-Gammadata electron analyser. The control philosophy for this analyser (as supplied by the company), is a rather monolythic application. On one side it allows to supply additional functionality, but on the other side it didn't fit into a client-server model. This was a serious obstacle not only for implementing experiments, where the analyser is considered to be just one part of the whole experimental setup, but also for synchronising the Scienta scans with the SLS top-up operation mode. Namely, the analyser is supposed to get into a stand-by mode during the top-up injection. Moreover the application was written in Delphi - a clearly non standard programming language for the SLS controls group.

The use of the EPICS Portable Channel Access (ActiveXCA) [6] for invoking the analyser operation-specific functions from the Scienta native code turned out to be the

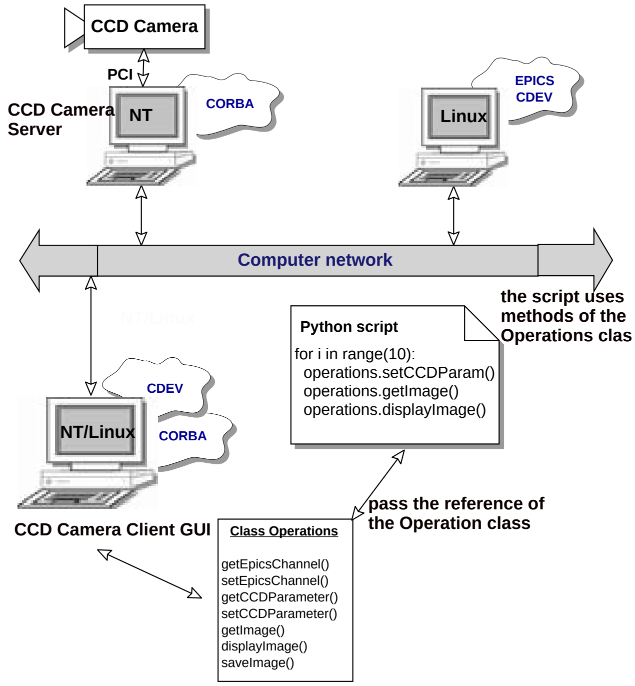


Figure 2: Schematical layout of a distributed architecture for controlling CCD camera including data acquisition (upper part) and handling (lower part). The bulk of operations on CCD camera and EPICS control system is implemented in a class called "Operations". By interpreting the script (here written in Jython) the reference of this class is passed to the script, which can call the class methods containing image acquisition, parameter settings and accessing the EPICS environment.

most simple solution. The Delphi Pascal units for the EPICS CA were provided by means of the ActiveX automation interface. The additional coding required just the initialization of the EPICS process variables and implementing their event handlers. In this way we planted EPICS process variables into this Delphi application, which allows us now to see the Scienta analyser as a virtual EPICS IOC. The event handlers implemented inside the planted EPICS process variables invoke the same functions the user invoke by clicking on the menu-buttons in the Scienta application.

4 DATA STORAGE AND HANDLING

In view of the different amounts and rates of data a flexible scheme for the acquisition, storage as well as handling of the data has been developed. The different beamlines produce data at rates ranging from several Mbytes/hour to 80 Mbytes/sec. The total amount of accumulated data ranges from Gbytes to several hundred Gbytes per day/experiment.

Each beamline has a dedicated file server of up to 300 Gbyte of available storage, which can be easily extended to more than 1 Tbyte. Since the beamline is controlled from a private network, the file server is equipped with two network cards that are connected to the private SLS-network and the

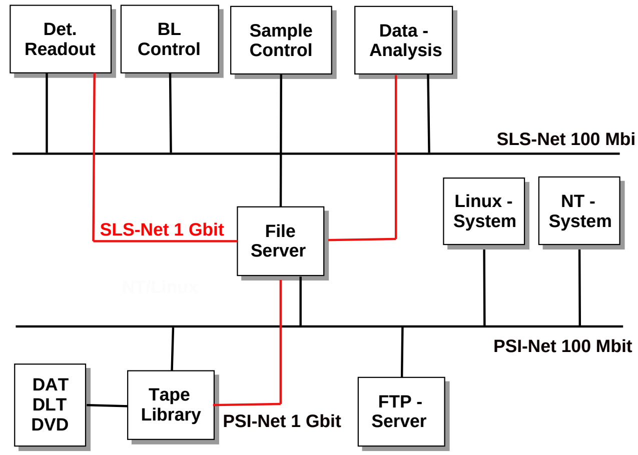


Figure 3: Schematical layout of beamline data storage and handling.

PSI network respectively (see Fig. 3). All data that has to leave the SLS has to go through this file server.

Currently, a tape-library is being set-up as a central medium-term storage facility. It has a storage capacity of 30 Tbytes. Users can, upon request, transfer their data to this medium. Also, they have the possibility to make copies of their data to other mass-storage media like DLT or DAT.

For the large volume data (more than 100 Gbyte) and commercial users, Network Attached Storage (NAS) or hot-swappable disks can be hooked up to the data-acquisition computers.

User access to the computers on the private SLS-net is granted through individual group accounts, which are connected to individual accounts on the PSI-domain through the group-id. The users at the beamline can decide on the level of security for their data.

5 CONCLUSIONS

The choice of the EPICS control system toolkit on the SLS beamlines clearly helps us to meet the tight time schedule of their commissioning. The generic solutions provided by EPICS in principle help us to avoid any coding. Thus the software maintenance across the beamlines is minimal. Another very important aspect is the fact that even beamline operators with no or minimal knowledge about EPICS can actively participate on their commissioning. The goal of the beamline controls group is to keep this philosophy valid also for the experimental stations.

The use of the CORBA architecture turns out to be a good approach for implementing experiments based on several heterogeneous sub-systems. The same applies for the Portable Channel Access (ActiveXCA). It allows to see the special instrumentation functionalities living as EPICS process variables. In either case it is possible to implement various experiments using simple scripts which access instrumentation-specific functionalities distributed on the computer network. The wide variety of APIs (tcl/Tk, IDL, Octave, perl...) for EPICS and CORBA is another benefit for this solution. Moreover, interfacing to EPICS with ActiveXCA opens the doors for Delphi, VisualBasic and VisualC++ developers providing also a "bridge" to COM based applications.

6 ACKNOWLEDGEMENTS

Many thanks go to Tim Mooney from APS for many valuable discussions and suggestions concerning the synApps package. The authors also wishes to acknowledge all contributions from the partners in the ELETTRA beamline control team, and finally special thanks go to SLS control group system and hardware administrators.

7 REFERENCES

[1] S.Hunt, PSI Scientific Report 1999, Volume VII.

[2] http://www.apl.anl.gov/xfd/bcda/documentation.html

[3] L. Emery, "User's Guide for SDDS-compliant...", June 2000, http://www.aps.anl.gov/asd/oag/manuals/

[4] M. Borland and M. Wood, "APS Computing News", Spring 1996: http://www.aps.anl.gov/asd/newsletter/newltr22.html

[5] The NeXus (Neutron & X-ray Data Format) Home Page: http://lns00.psi.ch/NeXus

[6] http://mesa53.lanl.gov/lansce8/Epics/PC/ActiveXCAServer/Default.htm

TUBT003
astro-ph/0111143

INTEGRATION OF THE VIMOS CONTROL SYSTEM

D. Mancini, P. Schipani, O. Caputi, G. Mancini, M. Brescia
Osservatorio Astronomico di Capodimonte, Napoli, Italy

Abstract

The VIRMOS consortium of French and Italian Institutes (PI: O. Le Fèvre, co-PI: G. Vettolani) is manufacturing two wide field imaging multi-object spectrographs for the European Southern Observatory Very Large Telescope (VLT), with emphasis on the ability to carry over spectroscopic surveys of large numbers of sources: the VIsible Multi-Object Spectrograph, VIMOS, and the Near InfraRed Multi-Object Spectrograph, NIRMOS.

VIMOS is being installed at the Nasmyth focus of the third Unit Telescope of the VLT at Mount Paranal in Chile, after a period of pre-integration in Europe at the Observatoire de Haute Provence. There are 52 motors to be controlled in parallel in the spectrograph, making VIMOS a complex machine to be handled. This paper will focus on the description of the control system, designed in the ESO VLT standard control concepts, and on some integration issues and problem solving strategies.

1 VIRMOS PROJECT

The need to collect large samples of objects has become a strong driver in the development of astronomical instrumentation. This is particularly evident for the observation of large samples of galaxies out to very large distances for which tens of thousands of objects are necessary to reach the accuracy needed in the measurement of fundamental astrophysical parameters. Multi-slit spectrographs allow to extract the source signal at much fainter levels than in multi-fiber spectrographs and are therefore now routinely used to reach the faintest magnitudes on several major telescopes.

In 1995, ESO launched a competitive call for proposal to study the feasibility of a Visible and/or Infrared Multi-Object Spectrograph, (VIRMOS), for the VLT. The VIRMOS consortium was awarded a one year contract in parallel with the Australis consortium of Australian institutes.

In October 1996, the VIRMOS proposal was selected by ESO. The project is now proceeding along a fast-track approach, with first light in early 2002 for VIMOS, the first to become operational. VIMOS and NIRMOS instruments allow for a large multiplex gain over a wide field, over the 0.37 to 1.8 µm domain. The unique multiplex gain allows to obtain spectra of up to 840 objects simultaneously with VIMOS, and up to 170 with NIRMOS (10 arcsec slits). An integral field spectroscopy mode with more than 6400 fibers coupled to micro-lenses will be available for VIMOS, covering a 1x1 arcmin² field. VIMOS and NIRMOS will cover up to 4x 8x8 arcmin^2. The instrument is divided into four channels. Each channel is in practice an imaging spectrograph provided with a field lens, a collimator, grisms or filter and a camera, coupled to a 2048x4096 pixel CCD for VIMOS, and a 2048x2048 HgCdTe Rockwell array for NIRMOS.

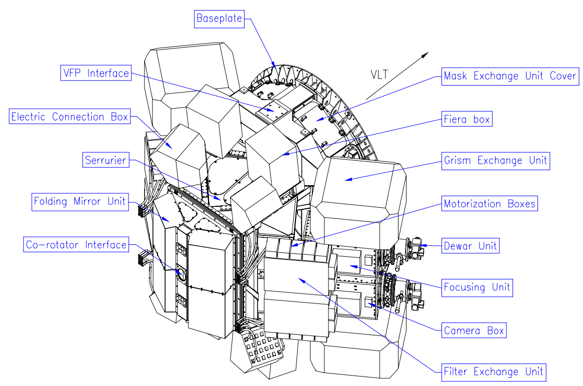


Figure 1: VIMOS spectrograph, CAD view

2 INSTRUMENT CONTROL

2.1 Subsystems

The VIMOS instrument can operate in three different modes:

- Multi-Object Spectroscopy mode
- Imaging mode
- Integral Field Unit (IFU) mode

This versatility implies a quite complex instrument control; 52 motorized functions needed to automate the instrument are actually 52. Most of them are symmetrical functions, identically mirrored on the four channels. The symmetry is not present only in the management of IFU. The electro-mechanical subsystems to be controlled are:

- Filter Exchange Unit (FEU)
- Mask Exchange Unit (MEU)
- Grism Exchange Unit (GEU)
- Mask Shutters Unit (MSU)
- Focusing Unit (FU)
- 1 Integral Field Unit (IFU)

Each unit is a complex electro-mechanical system, requiring many motorized functions.

The FEU is a 10 filter juke-box composed of a linear filter selector and a filter translator which moves the selected filter inside the optical path.

The MEU is a 15 mask exchanger equipped with a linear mask selector, a mask clamp to bring the selected slit mask, a mask translator to carry the mask to the focal plane, and a mask blocker to block the mask in position.

The GEU is a 6 grism exchanger, composed of a grism selector (a rotating carousel), a grism translator to move the selected grism to the optical path and a grism blocker to block the grism in the right position.

The MSU is composed of a couple of curtains able to obscure a selectable part of the field for calibration purposes.

The FU moves a lens in the camera in order to obtain the best focus.

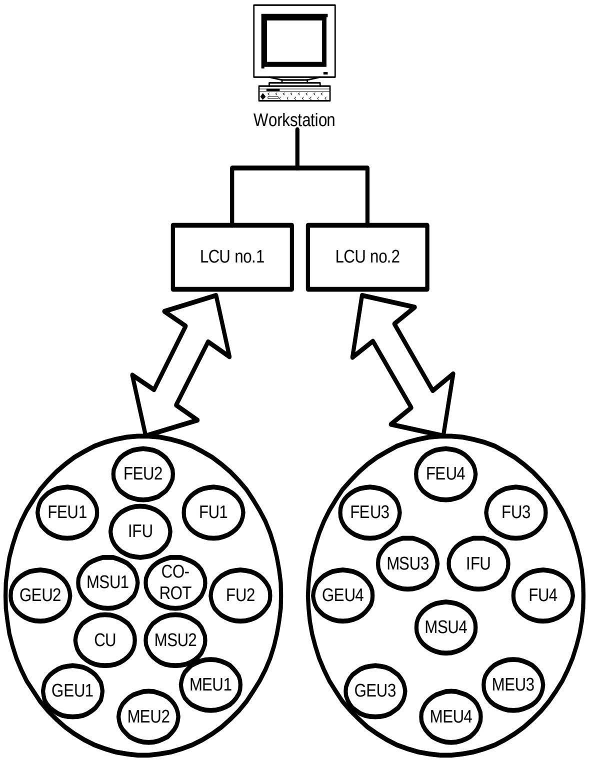


Figure 2: Subsystems division between the Local Control Units

Figure 3: VIMOS in assembling phase at Observatoire de Haute Provence

The IFU is composed, from the control point of view, of an elongator, able to change the magnification; a shutter, able to reduce the incoming light; four masks which are moved in pairs from the parking position to the focal plane when needed.

2.2 Instrument functions control

The specification requirements set to 90 sec the maximum time to change the total configuration of the instrument. This means that all the electro-mechanical subsystems (Filter Exchange Unit, Grism Exchange Unit, Mask Exchange Unit, Mask Shutters, Focusing Units, Integral Field Unit) must work in parallel without cross-interferences.

The motions requiring a fine position control are implemented using stepper motors with internal counting position measurement (therefore no encoders are used).

The motions requiring only a movement between two fixed positions are implemented using DC motors in speed control mode with two hardware limits.

In case some motions are mutually exclusive, they are hardware interlocked.

Besides the coordination and control of each of the 52 motions, other minor functions (beam temperature monitoring, cabinet cooling system management, calibration lamps management, housekeeping control) are performed to guarantee the proper operation of the instrument.

2.3 Low level software

A low level software library called Device Test Software (DTS) has been created to drive the instrument electronics and tune the operation of the instrument. It has been realized by the OAC team and integrated inside the overall Instrument Control Software (ICS) developed by the Observatoire Midi Pyrénées team. This library drives the electro-mechanical functionalities of the instrument, allowing the higher level software to access and control them without taking care of the electro-mechanical details.

2.4 Hardware architecture

The low level software runs on two VME Local Control Units, hosted in two separate control cabinets, each equipped with a Motorola MVME167 CPU card. The division of tasks in two LCUs was an obliged choice due to the high number of systems to be controlled. The operating system is WindRiver VxWorks; the VME crates are equipped with VME ESO standard boards. The two LCUs are managed by an HP-UX workstation. All the VME boards and electronic components have been chosen in conformity with ESO VLT instrumentation standards, in order to ensure compatibility with other ESO applications and simplify maintenance and integration in the VLT environment.

In each LCU the stepper motors are controlled in position mode by four pairs of Maccon Mac4-stp controller + ESO VME4ST amplifier boards, and the DC motors are controlled in speed mode by three ESO VME4SA boards.

It is evident that the two LCUs are highly overloaded with motor control tasks with respect to other similar applications. Some multitasking problems have arisen while testing the instrument in Europe at the Observatoire de Haute Provence, where the problems have been identified and circumvented.

Temperature data are collected from PT100 sensors to a Ester station via a 4-20mA converter, providing information used for automatic focusing to the LCUs via one of the serial ports of an Esd ISER8 board.

The calibration lamps are managed by digital I/O lines through Acromag boards; other diagnostic lines are used to monitor the status of the system.

The cooling system of the control cabinets is ensured by an ESO compliant cooling controller, communicating with the LCU via a serial port of the ISER8 board.

A large cable co-rotator has been necessary in order to drive and protect the high number of cables during the rotation of the instrument at the Nasmyth focus of the VLT. The motion of the co-rotator is slaved to the telescope instrument rotator through a potentiometer which detects the differential motion and consequently drives the co-rotator, so taking it continuously aligned with the Nasmyth rotator. A relay chain is implemented in order to inhibit the motion of both Nasmyth rotator and cable co-rotator in some emergency conditions (emergency keylock, limit switches, etc.).

3 ACKNOWLEDGEMENTS

The authors wish to thank O. Le Fèvre, G. Vettolani and all the VIRMOS consortium staff for their collaboration along the whole duration of the project.

4 REFERENCES

[1] O. Le Fevre, G.P. Vettolani, D. Maccagni, D. Mancini, J.G. Cuby, A. Mazure, Y. Mellier, J.P. Picat, M. Saisse, D. Delabre, B. Garilli, L. Hill, E. Prieto, L. Arnold, P. Conconi, E. Mattaini, C. Voet: “VIRMOS: Visible and InfraRed Multi-Object Spectrographs for the VLT” SPIE Proceedings of "Astronomical Telescopes and Instrumentation", SPIE 3355, pp 8-19

[2] O. Le Fevre, G.P. Vettolani, D. Maccagni, D. Mancini, J.P. Picat, Y. Mellier, A. Mazure, M. Arnaboldi, S. Charlot, J.G. Cuby, G. Guzzo, L. Tresse, G. Zamorani: “The VLT – VIRMOS Deep Survey", Proceedings of the IAP Conference, June 98

[3] Mancini, D., Le Fevre, O., Caputi, O., Ferragina, L., Fiume Garelli, V., Mancini, G., Tommasi Mavar, P., Parrella, C., Perrotta, F., Russo, M., Russo, R., Schibeci, M., Schipani, P.: 2000, “Vimos mechanical and control system design”, in “Optical and IR Telescope Instrumentation and Detectors”, eds Masanori, I., Moorwood, A., SPIE, 4008, pp. 256-267

[4] Le Fèvre, O., Saisse, M., Mancini, D., Vettolani, G., Maccagni, D., Picat, J.P., Mellier, Y., Mazure, A., Cuby, J.B., Delabre, B., Garilli, B., Hill, L., Prieto, E., Voet, C., Arnold, L., Brau-Nogué, S., Cascone, E., Conconi, P., Finger, G., Huster, G., Laloge, A., Lucuix, C., Mattaini, C., Schipani, P., Waultier, G., Zerbi, F., Avila, G., Beletic, J., d'Odorico, S., Moorwood, A., Monnet, G., Reyes, J.: 2000, “The VIMOS and NIRMOS multi-object spectrographs for the ESO-VLT”, in “Optical and IR Telescope Instrumentation and Detectors”, eds Masanori, I., Moorwood, A., SPIE, 4008, pp. 546-557

TUBT004
cs.OH/0111055

OVERVIEW OF THE NSTX CONTROL SYSTEM*

P. Sichta, J. Dong, G. Oliaro, P. Roney, PPPL, Princeton, NJ 08543, USA

Abstract

The National Spherical Torus Experiment (NSTX) is an innovative magnetic fusion device that was constructed by the Princeton Plasma Physics Laboratory (PPPL) in collaboration with the Oak Ridge National Laboratory, Columbia University, and the University of Washington at Seattle. Since achieving first plasma in 1999, the device has been used for fusion research through an international collaboration of over twenty institutions. The NSTX is operated through a collection of control systems that encompass a wide range of technology, from hardwired relay controls to real-time control systems with giga-FLOPS of capability. This paper presents a broad introduction to the control systems used on NSTX, with an emphasis on the computing controls, data acquisition, and synchronization systems.

1 INTRODUCTION

NSTX achieved first plasma in February, 1999. At the time, only ten computing-control subsystems were in use. Today, there are over fifty control subsystems. A diagram of the NSTX control system architecture is shown in fig. 1. The control subsystems can be divided into two categories, 'engineering' and 'physics'. Engineering control systems are used to create and control the fusion reactions that occur in the plasma. Physics systems, also known as diagnostics, are used to gather information about the plasma.

The NSTX device is operated in a 'pulsed' mode. Typically, the duration of an NSTX fusion pulse (a shot) is about 0.5 seconds. The torus' extremely compact center stack assembly needs about ten minutes to cool down between pulses.

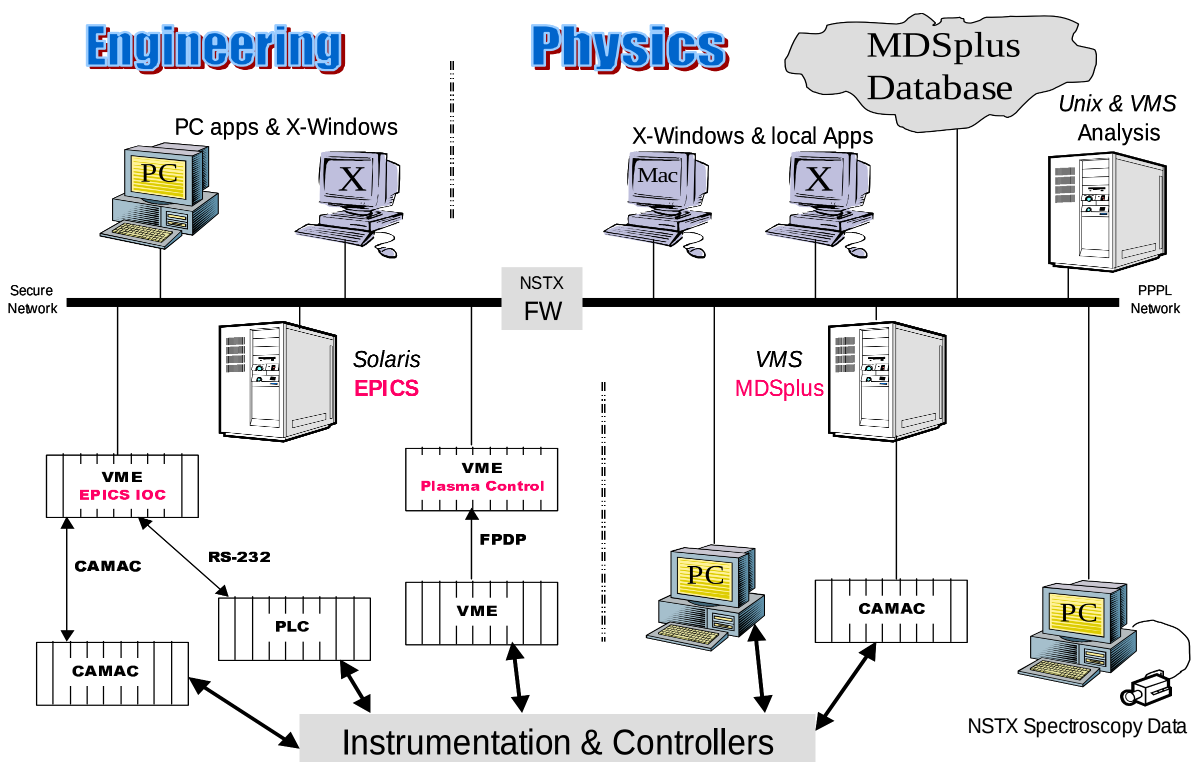


Fig 1 – Architecture of the NSTX Control System. There are Engineering subsystems and Physics subsystems. A variety of operating systems and human interface devices are supported.

* Work performed under the auspices of the US DOE by Princeton Plasma Physics Laboratory under Contract No. DE-AC02-76-CH0-3073.

Typically, the control systems record data at high-speed into 'local' memory during the short plasma pulse. The data is subsequently deposited into the *MDSplus* data management system. Physics analysis codes produce data plots and other scientific results, that can be used to 'tweak' the control systems for the next shot.

The NSTX control system is made up of commercial and collaboratively-developed components. Examples of the hardware include VME single board computers, x-terminals, workstations, and personal computers. Operating systems include Solaris, VMS, MacOS, Windows, and vxWorks. The bulk of the application software is *collaborative software*. This software is open-source, in use on a variety of projects, and is enhanced and maintained through the community of its users.

2 ENGINEERING SYSTEMS

The engineering systems provide the functionality needed to operate NSTX. Examples of these subsystems are the torus vacuum system, the water-cooling system, and the field coil power supplies. The systems incorporate a variety of hardware technologies. Functions that involve personnel and equipment protection typically employ older, well-characterized technologies such as relay logic or hardwired analog processing systems. Non-protective controls often use advanced technology, such as VME, to provide high performance and flexible operation. Programmable logic controllers (PLC) provide a middle ground, offering high reliability and a modern human-machine interface. Most NSTX engineering subsystems are interfaced with the Experimental Physics and Industrial Control System (EPICS), which has served NSTX well in its first two years of operation [1].

The most advanced control system at NSTX is the plasma control system[3]. This system controls the shape, position, and density of the plasma. It is a real-time digital control system that acquires over 200 input signals. Each millisecond, the field coil power supplies and gas injection systems are sent new control commands. The system features a 160 MB/s Front Panel Data Port (FPDP), an 8-processor, 20 giga-FLOP SkyBolt II computer, and real-time fiber optic data links. The system uses the Plasma Control System software, developed by General Atomics, Inc. for the DIII-D tokamak.

3 PHYSICS SYSTEMS

The physics control systems, or diagnostics, measure properties of the plasma. Most diagnostics are passive, in that they cannot affect the plasma. A few provide real-time signals representing a critical plasma parameter, such as plasma density. This signal can be used as a feedback signal for other control systems.

An increasing number of diagnostics are designed, operated, and maintained by non-PPPL scientists at collaborating institutions. PPPL liaison engineers help interface the system with NSTX facility. Considerations include mechanical interfaces, power and grounding, timing, and data management. The latter is very important since high level physics analysis codes sometimes require data from several diagnostics. New diagnostics often use a standard desktop PC. In addition to being inexpensive, the equipment that is connected to the PC often has software provided by the equipment's manufacturer. These attributes can significantly lower the overall cost of the diagnostic.

4 NSTX SOFTWARE

The NSTX software was designed to use commercial and collaborative software, i.e. to minimize new system software development. Collaborative software has permitted NSTX computer support to be highly effective and reliable. Collaborative software packages used on NSTX are the Experimental Physics and Industrial Control System (EPICS), the Model Data System (MDSplus)[2], Scope, the Plasma Control System software (PCS), and the Inter-Process Communication System (IPCS).

4.1 EPICS

The EPICS software is used to integrate most engineering subsystems. It is extensible to many I/O technologies and runs under several popular operating systems. EPICS is used to acquire and archive NSTX engineering data. The NSTX-specific application is presently comprised of 3 IOCs, 135 EPICS process displays, 1,500 I/O points, and 7,000 EPICS records. EPICS is designed for continuous process control applications. 'C' programs and unix-scripts were required to support the special needs of NSTX's 'pulsed' operations and to interface EPICS with other NSTX software.

The EPICS human machine interface is implemented using over twenty x-terminals and PCs. These are used by the operators to monitor all engineering subsystems and, if authorized, control them.

4.2 MDSplus

NSTX data management is based on MDSplus, which is a hierarchically organized database that is created for each NSTX shot. Both raw and analyzed data from the engineering and physics control systems are written into the 'shot tree'. Currently, MDSplus stores 100 MB of data per shot, more than 5000 waveforms and 25,000 parameters. The NSTX MDSplus server runs on VMS. MDSplus clients run under VMS, unix, and

Windows. There are FORTRAN, C, IDL, and Java interfaces to the MDSplus database.

In addition to data storage, MDSplus provides device control capability and an event system. For example, an MDSplus event can be used to refresh a plot with new data, or to control a CAMAC-interfaced instrument.

4.3 Visualization

The main data visualization tools on NSTX are *Scope* and *IDL*. Scope is a 2-D data visualization tool that can plot from one to sixty four signals in an X-window. The tool is designed to access data from the MDSplus shot tree. It has a convenient graphical user interface with the capability to pan and zoom any or all X-Y plots. In response to MDSplus events, it can automatically re-plot a new shot's data from the MDSplus shot tree. Scope runs on unix and VMS.

IDL is often used for complex data analysis and visualization. IDL functions are used to interface with MDSplus. IDL is also used as a basis for an NSTX Physics electronic logbook.

4.4 Web Tools

Many NSTX users interact with the NSTX control systems and data using application programs that use the X-windows interface. New tools are increasingly aimed towards the world-wide-web browser interface. NSTX uses a web interface for several purposes. First, there are 'help' functions such as FAQs, links to local and collaborators' documentation, and training presentations. There are search tools to help the user find signals in the MDSplus database, and plotting and data-export tools. The logbook also has a web-browser interface. Some of the new MDSplus tools use Java.

5 TIMING AND SYNCHRONIZATION

All control systems use the NSTX Central Clock system for timing and synchronization. The system encodes up to fifteen pre-defined timing events, with one microsecond resolution. The Central Clock hardware is made up of CAMAC modules. There is real-time software running under EPICS, and several EPICS displays to configure and control the clock.

Software synchronization at NSTX is based on two software packages, the Inter-Process Communication System (IPCS) and the Event Manager (EVT). IPCS runs on unix and VMS and has both C and FORTRAN application program interfaces. EVT is an event distribution system that is built on top of IPCS. These messaging programs are used to coordinate software (tasks) that are executing on different computers. EPICS is used to interface the Central Clock hardware with these programs.

6 CONCLUSION

The ability to quickly produce and maintain a highly reliable control system by a small group of developers would not be possible without generous support from our collaborators at institutions around the world. The use of collaborative software has helped to make a software environment that spans multiple operating systems and programming languages. Active collaborations also provide an avenue to keep the software compatible with modern computing technologies.

The NSTX control system is rapidly expanding to include new and diverse systems. The trend of new engineering and diagnostic subsystems is toward independent system design and development by 'the experts' in the applicable area of research or engineering, not by a central NSTX engineering group. A key requirement for these systems is their ability to interface with the MDSplus data management system, the Central Clock system, and in the case where integrated systems control is required, with EPICS.

To maintain the reliability of the control system that was built using legacy CAMAC equipment, a program to replace aging equipment with modern hardware will need to be completed. Exploratory work to improve the NSTX Central Clock to provide additional performance and capability is underway. Emerging technologies such as CompactPCI and FPGA devices are currently being explored.

7 REFERENCES

[1] P. Sichta, J. Dong, 'Status of the Experimental Physics and Industrial Control System at NSTX,' *19th Symposium on Fusion Engineering*, Atlantic City, NJ (to be published 2002).

[2] W. Davis, et al, 'The use of MDSplus on NSTX at PPPL,' 3rd IAEA TCM on Control, Data Acquisition, and Remote Participation for Fusion Research, Padova, Italy (2001).

[3] D. Gates et al., 'Initial Operation of the NSTX Plasma Control System,' 27th EPS Conference on Contr. Fusion and Plasma Physics, Budapest (2000).

TUBT005
hep-ph/0110403

REPORT ON THE PCAPAC 2000 WORKSHOP

R. Bacher*, Deutsches Elektronen-Synchrotron DESY, D-22603 Hamburg, Germany

Abstract

In October 2000, the third PCaPAC (PCs and Particle Accelerator Controls) workshop took place at DESY. This paper presents a summary of the workshop. The workshop reviewed existing and new PC-based accelerator control systems, from small-scale to large-scale installations. It demonstrated convincingly the advantage of modern, commercial mass-market products used for accelerator controls. Disadvantages of these technologies were reported as well. Large-scale PC systems inherently bring administrative concerns into the picture. In this vein, special emphasis was given to system administration for distributed systems. A major topic of the workshop was the integration of different control system approaches as well as the integration of different platforms within the same control system. In particular, PC-based concepts offer the simple opportunity to interface to commercial SCADA systems. In addition, large emphasis was given to the presentation of future developments including the next network trends to data exchange via SOAP and XML.

1 INTRODUCTION

The PCaPAC (PCs and Particle Accelerator Controls) workshop began in October 1996 at DESY and was designed to be a biennial workshop. The second workshop was held in January 1999 at KEK and the third took place in October 2000 again at DESY†. PCaPAC is scheduled to be held in the "non-ICALEPCS" years.

PCaPAC is not in conflict with ICALEPCS, although it has a significant overlap in topics with ICALEPCS due to the common focus on accelerator control systems. However, PCaPAC focus particularly on the impact of PCs on particle accelerator controls. The usage of PCs brings especial advantages as well as disadvantages. Therefore, the workshop provides a platform to exchange ideas and experience in the dedicated field of PC-related technologies where trends are changing rapidly and where the pace at which hardware as well as software evolve is very fast.

PCaPAC is designed to be a workshop. Following this approach, no parallel sessions were scheduled to give everybody a chance to attend. The morning sessions were dedicated to oral presentation categorized according to the announced workshop topics. The afternoon sessions were split; a poster session was followed by an open-discussion session guided by a chairman. These discussion sessions picked up open questions left from the morning sessions and provided a forum to discuss special topics raised by the conference attendees or the chairman.

Participation in PCaPAC 2000 reached an all-time high of 127 registered attendees from 43 different institutes and 17 countries.

2 WORKSHOP TOPICS

2.1 Scientific Programme

The scientific programme of the workshop was categorized according the following topics:

Controlling Accelerator Subsystems and Experiments (37 contributions)

- running systems: including commercial or SCADA systems
- peripheral systems: including archiving, alarm, sequencing etc.
- process control: including OPC
- data acquisition
- real-time solutions

Interfacing Accelerator Hardware (2 contributions)

- IO control
- field busses
- device drivers etc.

Accelerator Control Objects and Components (14 contributions)

- object-oriented design and implementation involving OLE, DCOM, ActiveX, Java, Java Beans, CORBA, C++ etc.

Integrating Different Systems (6 contributions)

- operating systems and platforms
- commercial systems, SCADA systems, in-house systems, web-based systems

Control System Architecture (15 contributions)

- methods of distributing control and data exchange mechanisms
- layering models for data flow

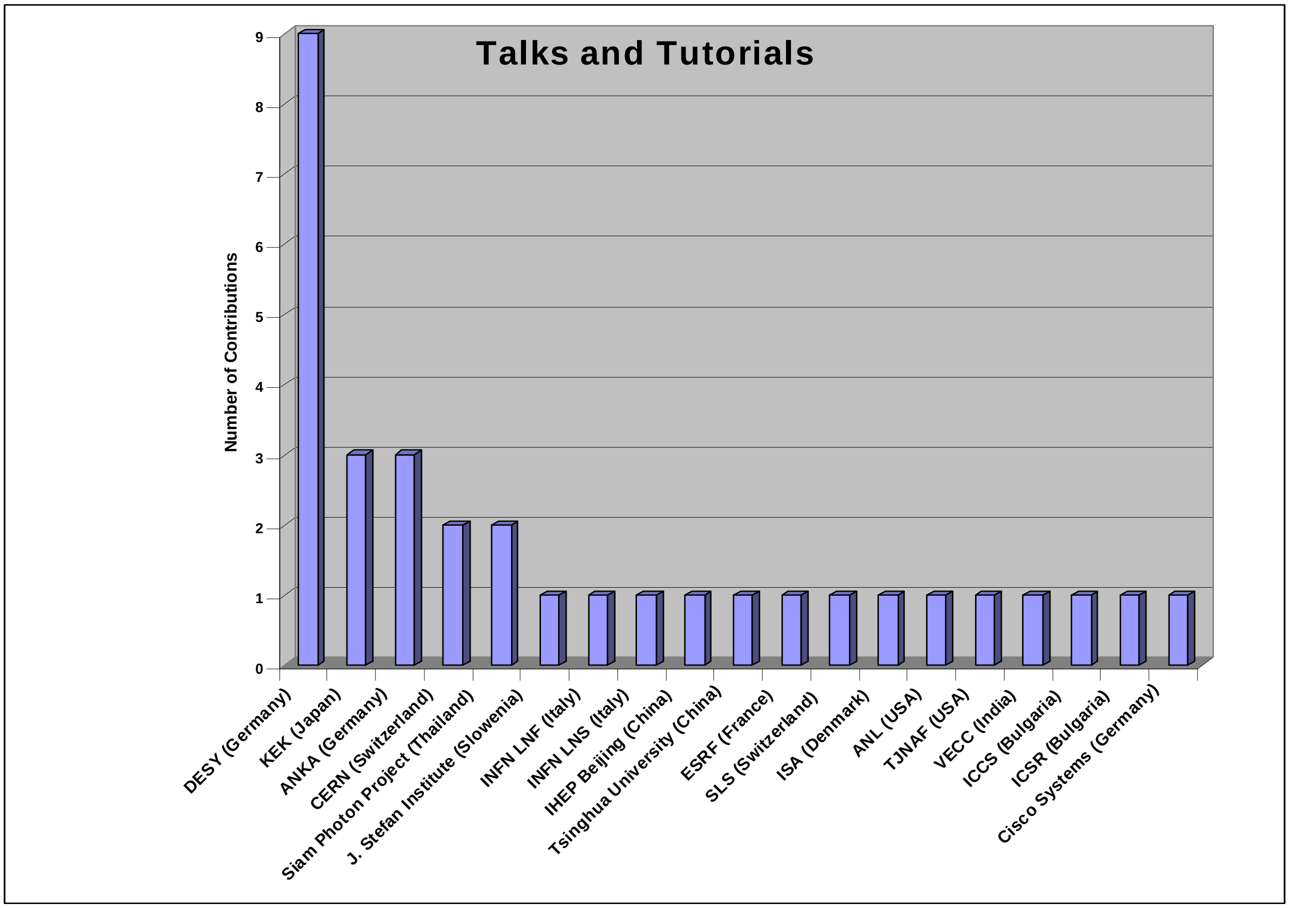


Figure 1: Number of talks and tutorials versus presenting institutes

System Administration and Project Management (7 contributions)

- impact of system administration on machine operation
- control system project management using PC-based tools

Operator Interface (3 contributions)

- man-machine interface
- ergonomics etc.

Future Trends and Technologies (8 contributions)

- XML, 64-bit processors, WAP, data warehousing and mining, blue tooth
- accommodating trends in old systems

In total, 92 contributions (oral: 31, poster: 58, tutorial: 3) were presented (see Fig.1).

2.2 Status and Progress Reports

At PCaPAC 2000, many running accelerators of which the control-system infrastructure was based entirely or in part on PCs were presented. Among these were small systems built and maintained by a few people (e.g. the storage rings ASTRID and ELISA of the ISA Storage Ring Facilities at the University of Aarhus in Denmark), medium-scale systems (e.g. the ANKA synchrotron light source in Karlsruhe in Germany, and accelerators at KEK in Japan) and probably the largest PC-based control systems for the HERA, PETRA and DORIS storage rings and their injectors at DESY in Hamburg (Germany). The volume and success of development work was manifest in a series of corresponding status reports.

The idea of the control system for the synchrotron radiation light source ANKA is to provide a user-friendly control system where "user" means a physicist operating the accelerator and not a computer expert [1]. The control system contains around 500 physical devices (power supplies, vacuum pumps, beam position monitors, RF generators, etc.) The device I/O is handled by self-sufficient micro-controller boards, which connect to a standard LonWorks field bus network. Each branch of the network is attached to a PC. Device servers running on these PCs map the devices and their properties onto 1500 objects and make them remotely available using CORBA. On the client side, the objects are wrapped into specially developed Java Beans called Abeans [2], which provide also a rich framework of tools that clients always need. This allows applications to be written easily, even by nonprogrammers.

COACK (Component Oriented Accelerator Control Kernel) [3] being developed at KEK in Japan in joint venture with IT-industry is a general-purpose kernel for accelerator controls based on Microsoft's COM/DCOM foundation. It consists of different components like a class builder, data cache, message exchanger (peer-to-peer and publish-subscribe type communication), accelerator virtual machine, operation support (scheduling and polling), database (SQL, XML, Excel) and data/command logging, own registry and diagnostic tools (session manager and status display). In addition, support to integrate complete SCADA systems and lab automation software like LabView or HPVEE is provided.

2.3 System Administration Concerns

The distributed nature of PC-based control systems is manifest in the important role that is played by system administration.

The automated installation for Linux PCs (reconstruction farms, work group server and desktop workstations) is done at DESY with YaST, the administration tool of the SuSE Linux distribution. In spring 2000, SuSE has introduced the successor tool YaST2 [4]. DESY took the chance to influence the YaST2 development in an early stage and have SuSE integrate features to turn YaST2 into a network-based configuration management. YaST2 takes a fully modular approach to configuration, installation, and administration of the system. It comes with its own scripting language ycp (YaST control protocol), which has an interface to call other codes, like C, perl, or shell script. YaST2 supports remote, network transparent configuration, installation, and administration of a PC or a group of PCs by users or group administrators. The configuration is stored on a network database. Existing configurations can be cloned to set up clusters of identical PCs. A hierarchical administration structure can be realised.

To manage the Windows-based HERA console workstations at DESY, NetInstall is used for system and application management [5]. The idea is to use the same tool for the HERA consoles as for the desktop PCs at DESY. The benefit is threefold, (1) sharing of knowledge, (2) exporting control software to all office PCs at DESY and (3) importing standard software from central application support. About 50 dedicated control PCs are located in various control rooms. The update of DLLs and other important software is done at logon time. In addition, a check of the control application at launch time for newer versions is performed.

2.4 Integrating Different Control Systems

It is a common and unavoidable experience that different control systems have to be interfaced. Complete commercial SCADA systems as well as in-house developed control systems have to be integrated into other accelerator control systems.

An example is the EPICS-to-TINE translator [6] at DESY. TINE is the dominant accelerator control system protocol at HERA [7]. The translator server resides directly on the EPICS IOC. The EPICS names are mapped to TINE device names. The Channel Access Protocol is by-passed and the process variables are accessed directly using the database access layer including the necessary data type conversions.

2.5 Future Trends and Technologies

PCaPAC has been seen to be worthwhile in discussions involving the technologies of the fast evolving internet.

A. Pace from CERN [8] demonstrated impressively the usage of the Web as a platform-independent application platform and the potential of SOAP (Simple Object Access Protocol) as a Web-wide application protocol based on HTTP and XML. He recommended the integration of the control systems into the Web to make Web technologies the core and not the border of the control systems. Furthermore, he pointed out the possibilities which rendering software provides. This will open control systems not only to traditional Web browsers but also mobile phones, WebTV or any internet enabled device.

3 ACKNOWLEDGEMENT

The author would like to thank P. Duval, I. Nikodem, J. Maaß, R. Schmidt, R. Schröder and W. Schütte for their great enthusiasm and effort organizing the PCaPAC 2000 workshop.

4 REFERENCES

[1] M. Plesko and the KGB Team, Proceedings of PCaPAC 2000 workshop, ID 96 (Managing the Development of a New Control System) and ID 115 (ANKA Control System Takes Control) and references therein

[2] M. Plesko and G. Tkacik, Proceedings of PCaPAC 2000 workshop, ID 89 (How to Build Professional Control System Applications)

[3] I. Abe et al., Proceedings of PCaPAC 2000 workshop, ID 117 (Recent Status of COACK and its Functions)

[4] T. Kleinwort and K. Woller, Proceedings of PCaPAC 2000 workshop, ID 131 (YaST2 Based

Automated Linux Installations at DESY – Taking Central Linux Support to the Next Level)
[5] C. Trachimow, Proceedings of PCaPAC 2000 workshop, ID 136 (System Management in a Windows based Control Environment)
[6] Z. Kakucs, P. Duval and M. Clausen, Proceedings of PCaPAC 2000 workshop, ID 128 (An EPICS to TINE Translator)
[7] P. Duval, Proceedings of PCaPAC 2000 workshop, ID 69 (The TINE Control System Protocol: Status Report)
[8] A. Pace, Proceedings of PCaPAC 2000 workshop, ID 140 (Can Future Web Technologies Profit to Accelerator Control?)

* for the PCaPAC 2000 organizing committee
† http://desyntwww.desy.de/pcapac/

TUESDAY, 27 NOVEMBER 2001

TUC - INTEGRATING INDUSTRIAL SYSTEMS IN EXPERIMENTAL PHYSICS CONTROLS, EXPERIMENT CONTROL SYSTEMS

PURCHASING ACCELERATOR SUBSYSTEMS AS TURNKEY COMPONENTS

S. Hunt, PSI, Villigen, Switzerland

Abstract

Many new accelerator projects are purchasing complex components as turn-key systems rather than building them in-house. If not managed efficiently this can cause problems in integrating these components with the global accelerator control system. Having these components delivered with a 'foreign' control system can cause performance bottlenecks and problems of maintenance. Later replacing these controls is often necessary, but is time consuming and expensive. At SLS we avoided these problems by having the external companies deliver these systems (including Linac and RF) with Epics [1]. To ensure compatibility, we supplied the necessary hardware, and provided the companies with training and support. This ensured a relatively trouble free integration, and contributed to the very fast commissioning of the SLS.

1 INTRODUCTION

When building accelerators and other large scientific projects, such as high energy physics experiments, there has always been a difficult decision if to build components in-house or purchase from industry. The amount of industrial participation often varies, and may include consultancy only, provision of labor, series production, prototype construction, design, requirements capture, or complete turnkey solutions. Systems provided by industry may be of the shelf, modified from standard products, or totally new designs. The scale of components built with some level of industrial participation may be individual small components to complete accelerators.

2 WHY BUILD IN-HOUSE

2.1 Maintenance

Having systems built in-house means the expertise to maintain and develop the systems are readily available. Local staff will have a much deeper knowledge of the system than would otherwise be the case.

2.2 Consistent standards

Following laboratory standards of construction, wiring and documentation, can improve the ease of use of the systems for staff. Local safety procedures, regulations and conventions are more likely to be followed.

2.3 Cost of manpower

In many laboratories the full economic cost of in-house manpower is not taken into consideration when evaluating the cost of projects. Staff may already be in place, which can make it seem cheaper to build in-house.

2.4 Control over schedule

It may be considered dangerous to rely on the supply of schedule critical components from outside sources. It can be frustrating to feel powerless to accelerate progress by directly increasing resources to a project. However building in-house is often no guarantee of delivering on time, and penalty clauses in contracts can act as a powerful incentive for companies to deliver on schedule.

3 WHY BUY FROM INDUSTRY

3.1 Economies of scale

Industrial suppliers may have economies of scale in providing components to more than one laboratory, or to other types of customers. Many multi-use technologies are now used in accelerators and this has an impact on price.

3.2 Lack of expertise or manpower

Laboratories may not have enough competent staff to produce the systems in-house. This is often the case at smaller institutes and new laboratories, a fact they can then use to their advantage by making them more likely to make use of the benefits of using outside companies. Larger established laboratories may have lower staff levels than in the past.

3.3 Price effects of competition

Competitive bidding tends to result in lower prices for systems. Single sources of supply, whether from industry or in-house groups almost always results in higher costs. It is often worthwhile to compare even a preferred in-house price to an external bid.

3.4 Access to new technologies and ideas

Industry may be able to offer alternative approaches and technologies. These may have been developed from projects far removed from the accelerator domain.

4 CAN LARGE SYSTEMS BE DELIVERED WITH INTEGRATED CONTROLS

Accelerator and large experimental physics control systems have largely evolved over the years in a similar direction. This has been mainly towards a distributed computer system, with control and monitoring of signals carried out in a high performance, often real-time, environment.

Many of the controls requirements, even of a modern high performance accelerator, can be carried out using industrial process controls technology. However there is a significant, and perhaps growing class of requirement that cannot. Many of the latest accelerators have very demanding requirements for example for beam positioning and stability, and these requirements translate into stringent controls requirements.

We are therefore faced with the choice of using two systems, one high performance and one low performance, or having one, high performance, system cover all requirements. As signal densities can be very high in high performance systems, cost per channel can be as low or lower than low performance industrial I/O.

5 PROBLEMS WITH INDUSTRIAL PROCESS CONTROLS IN AN ACCELERATOR ENVIRONMENT

5.1 Timing

Many accelerator components have stringent requirements for timing and synchronization. Actions and measurements have to be synchronized to high accuracy, for example for injection, extraction, ramping, or data taking. Often software and hardware timing events are needed, and have to be dynamically modified in a flexible manner. These facilities are not available in PLC systems.

5.2 Data archiving and correlation

Data archiving and trending are available in process control systems, but often with limitations on the number of channels and frequency of update. The lack of accurate timestamp information makes exact correlation difficult. And it is also very difficult to compare data from PLCs with information from the high performance systems.

5.3 Alarms

A single, unified system to monitor, classify, and display alarm information is a vital tool in running a modern accelerator. It may be necessary to monitor tens or hundreds of thousands of channels, often from multiple locations. It might also be necessary to dynamically modify alarm limits, or mask alarms (for instance when changing accelerator operating mode or particle type) at run time. It might be necessary to have an associated first-fault system with mili-second or micro-second resolution.

5.4 Speed, accuracy and resolution

PLC systems typically have analogue I/O with relatively low measurement rates, low resolution, and low accuracy. Modern high performance accelerator controls systems typically have much faster ADCs and DACs, often with 18 or 20 bit resolution, and much better accuracy.

5.4 Special interfaces

Accelerator and large experimental physics control systems need many different interfaces not widely available in industrial systems. Examples of such devices are waveform digitizers, charge ADCs, scalers, high voltage power supplies, etc. These devices are widely available in Camac and VME. Even such common interfaces as GPIB are not well supported in PLCs.

6 EXPERIENCE AT SLS

Due to tight timescale and manpower constraints at SLS, it was decided to purchase a number of large accelerator systems from industry. These included the complete Linac, and the Booster and Main Ring RF Modulator systems. An early decision was made by the SLS controls group to attempt to have these systems delivered by the vendor using our standard controls hardware and software.

6.1 SLS Linac

The complete SLS 100 MeV electron Linac was purchased as a turnkey system from Accel. As part of the contract they agreed to deliver the system using EPICS controls. This was made possible by providing Accel and their subcontractors PPT, with a thorough introduction to Epics tools and hardware used at SLS. The complete system was delivered on time, on a tight time schedule. The low level controls and user screens delivered are totally and seamlessly integrated into the SLS control system, and have needed only very minor modification.

6.2 RF Modulator system

Five turnkey RF modulators were ordered from Thomcast AG. Each modulator has stringent requirements for controls. Thomcast engineers were trained on using Epics with our standard controls VME hardware. After only two days of training they were able to start to implement the RF control systems, using hardware supplied by SLS. Support was available but rarely needed. One comment we received back was that it was very easy to make the transition from programming a PLC system to using Epics. The company has subsequently delivered similar RF systems with Epics controls to the Shanghi Synchrotron Light Source, and has further orders and interest.

6.3 Hydrostatic leveling system

The hydrostatic leveling system was also ordered as a turnkey system. In this case we were unsuccessful in getting the company to adopt Epics as a control system. This was thought to be acceptable as the HLS system is less tightly coupled to the rest of the accelerators as the Linac and RF systems. The systems adopted is based on CAN bus and software on an NT PC. Delays in the control system implementation meant the HLS system was not available at ring commissioning. Not having this system tightly integrated into the accelerator control system has meant difficulty archiving, correlation, and handling of alarms. Two of the twelve machine sectors have subsequently been directly connected in parallel to the Epics control system using our standard VME crates and analogue I/O modules. This process took less than two weeks of work and provides better resolution, less noise and faster update rate. This will probably be extended to all sectors and replace the existing system.

7 LEASONS LEARNED

- Don’t confuse advantages of having industry supply turnkey systems with using low performance industrial (PLC and SCADA) systems
- Don’t underestimate the ability or willingness of industrial partners to use your control system if it is well designed and easy to use.
- Understand the limitations of industrial controls when making your decisions.

REFERENCES

[1] L. Dalesio et al., “The Experimental Physics and Industrial Control System Architecture: Past, Present and Future”, ICALEPCS 1993, Berlin.

TUCT002
cs.OH/0111017

FIRST EXPERIENCES INTEGRATING PC DISTRIBUTED I/O INTO ARGONNE'S ATLAS CONTROL SYSTEM

F. H. Munson, D. E. R. Quock, S. L. Dean*, and K. J. Eder*,
ANL, Argonne, IL 60439, USA

Abstract

The roots of ATLAS (Argonne Tandem-Linac Accelerator System) date back to the early 1960's. Located at the Argonne National Laboratory, the accelerator has been designated a National User Facility, which focuses primarily on heavy-ion nuclear physics. Like the accelerator it services, the control system has been in a constant state of evolution. The present real-time portion of the control system is based on the commercial product Vsystem [1]. While Vsystem has always been capable of distributed I/O processing, the latest offering of this product provides for the use of relatively inexpensive PC hardware and software. This paper reviews the status of the ATLAS control system, and describes first experiences with PC distributed I/O.

1 ABOUT ATLAS

ATLAS is an ion accelerator that combines older accelerator technology with newer technology. The configuration of the accelerator system consists of three ion sources, two injectors, and two LINAC (Linear Accelerator) sections. One of the injectors is an electrostatic Tandem accelerator, which utilizes some of the earliest accelerator techniques. The second injector is a positive-ion LINAC, which uses some of the latest superconducting accelerator technologies. Ions from each injector are transported to the entrance of the first booster LINAC. Ions at the output of the first booster LINAC can then be transported to the entrance of the last booster LINAC, and then directed to one of several target areas. The LINAC portion of the accelerator consists of sixty superconducting accelerating resonator structures.

2 THE ATLAS CONTROL SYSTEM

2.1 System Hardware

An upgrade to the control system was completed in 1998 [2]. Like the accelerator, the control system employs both new and old technologies. The primary interface to various accelerator components is a CAMAC (Computer Automated Measurement And Control) subsystem. While CAMAC is still used extensively in some control and data acquisition systems, and while manufacturers continue to produce CAMAC equipment, CAMAC is considered by many to be an outdated technology. The CAMAC subsystem is configured as a single CAMAC Serial Highway, which currently links 18 crates, and operates at a clock speed of 2.5 MHz.

A second subsystem that is part of the control system is an Ethernet LAN (Local Area Network). This LAN is a hybrid system consisting of 10 MB/s and 100 MB/s segments. The Ethernet subsystem is used to link all of the control system's computers.

At the core of the control system is a Compaq AlphaServer [3]. This computer provides the only link to the CAMAC subsystem. This link is established by a PCI (Peripheral Component Interconnect) to CAMAC Serial Highway interface. The interface and a software driver were acquired from Kinetic Systems Inc. [4]. In addition to the online AlphaServer, the control system consists of a backup AlphaServer, five AlphaStations, and fourteen PCs.

2.2 System Software

The operating system software currently used by the ATLAS control system is Compaq's "OpenVMS" and Microsoft's "Windows NT/2000" [5].

Vista Control Systems' software package "Vsystem" is used to provide the real-time features of the control system. Vsystem is a network distributed control system software that provides distributed database access, and supports CAMAC I/O processing.

Two relational database products play an important role in the operation of the control system. The first relational database is Oracle's "Oracle Rdb" [6]. This database is used to store static information about accelerator parameters and devices that would be inappropriate for storage in the Vsystem real-time database. The second relational database is Corel's "Paradox" [7]. The Oracle Rdb and Paradox relational

* Undergraduate Research Participants

database systems team up to provide the operator with a graphical user interface to archived tune data. These stored data provide the operator with a means to restore the entire accelerator to a configuration used during a previous experiment.

2.3 System Personnel

The control system staff most recently has consisted of one full-time system manager/software engineer, one full-time engineer, and two part-time undergraduate student programmers. This staff is responsible for maintaining all computer systems associated with the control system, the control system LAN, and the CAMAC subsystem. In addition, this group is responsible for all in-house written software maintenance, as well as any requests for new features and necessary upgrades.

3 MOTIVATIONS FOR MODIFYING THE CAMAC SUBSYSTEM DESIGN

While the CAMAC subsystem used at ATLAS has been both reliable and versatile, it has long been recognized that the system has the following inherent disadvantages:

- The serial highway configuration transmits commands and receives responses at one single point of contact. This design establishes a potential constraint to overall performance.
- CAMAC I/O processing is performed by only one computer. As the system grows, future peak demand periods could exceed the processing capability of this single machine.
- Operation of the entire control system depends on the reliability of one computer system. If this system is down, the entire control system is rendered inoperative.

4 ONE POSSIBLE APPROACH

It seems obvious that one solution to ease the burden of one computer system performing all of the CAMAC I/O, and to increase the number of I/O ports accessing CAMAC crates, is to add computer systems with CAMAC interfaces. Borrowing from the concepts of more contemporary control system designs such as the popular control system package EPICS (Experimental Physics and Industrial Control System) [8], local I/O processing could be added to selected CAMAC crates. While probably not practical at ATLAS, in principle, a single computer system could be interfaced directly to each of the 18 on-line CAMAC crates, thus providing local I/O processing for each crate.

Two things make this approach more feasible for the ATLAS control system today than in the past. The first is the low cost of PC hardware and software, and the second is the port of Vsystem to PC operating systems like Windows NT/2000 and Linux [9].

5 A TEST CASE

In the Vsystem environment, database records (Vsystem documentation refers to them as channels) contain all the information necessary to access the I/O hardware. At ATLAS, these records are organized primarily by system or device type into several different databases. Currently in the ATLAS control system, all Vsystem databases reside on the AlphaServer that is connected to the CAMAC serial highway. One of these databases is the cryogenics system database.

Due to the superconducting design of many ATLAS components, the reliability of monitoring cryogenic system parameters is crucial. Since the cryogenic system offered a good opportunity for funneling CAMAC I/O to one CAMAC crate, it was chosen as an ideal candidate for a prototype system.

6 PROTOTYPE IMPLEMENTATION

The following steps were taken to implement the prototype system:

- A PC was acquired to allow for a proof of principle investigation. The PC has been assembled in a rack mount configuration.
- A WIeNeR PCI to CAMAC interface and software driver was acquired for this initial machine [10].
- The Linux operating system and Vsystem were installed on the PC.
- The cryogenics database was then copied to the new PC, and modified to use the new PCI to CAMAC interface connected to the local CAMAC crate.
- The cryogenics graphical user displays were copied to the new PC. These displays, as well as the displays on the main system, were modified to reference the cryogenic database on the PC. This allowed data retrieved from the locally connected CAMAC crate to be displayed on all control system nodes.
- Cryogenics system signal cables were then moved from the serial highway crate to input modules in the new local test crate.

The result is that cryogenic system data can now be retrieved from a CAMAC crate that is not part of the CAMAC serial highway. These data can be displayed on the test PC or on any node that is part of the main system. The complete control system is illustrated in Figure 1.

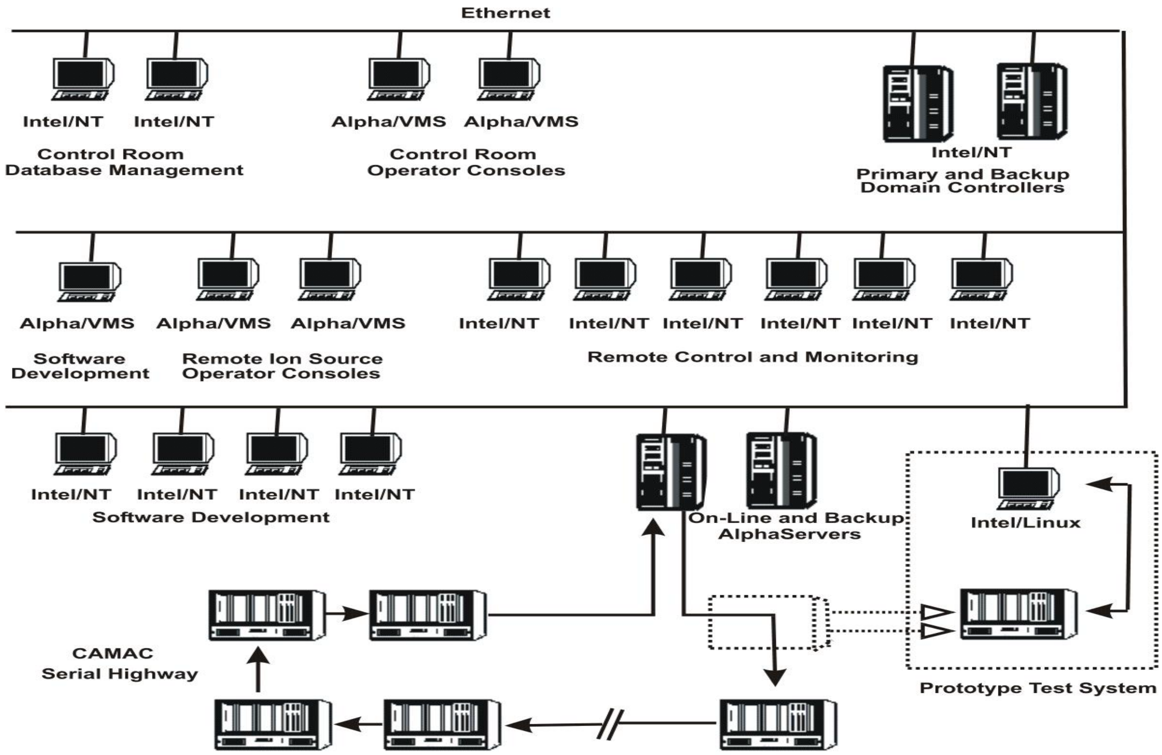


Figure 1: ATLAS Control System Showing Prototype Distributed I/O

7 CONCLUSIONS

It has been demonstrated, through initial experiences at ATLAS, that relatively low cost distributed CAMAC I/O processing is possible within the Vsystem environment. While converting the entire control system to a distributed configuration, as described in this paper, would be a major undertaking, it seems that the future of the ATLAS control system may no longer depend on the CAMAC serial highway as the only option. In fact, the prototype design suggests that the CAMAC subsystem itself is no longer the only option. The prototype system could just as easily have been configured to use a VME (Versa Module Europa) or possibly a VXI (VME eXtensions for Instrumentation) subsystem.

The current plan is to experiment with other bus structures, as well as other PC operating systems. This activity would allow ATLAS staff members to determine which would be most appropriate for the ATLAS control system.

This work is supported by the U.S. Dept. of Energy, Nuclear Physics Div., under contract W-31-109-ENG-38.

REFERENCES

[1] Vista Control Systems, Inc., Los Alamos, NM, USA.
[2] F. Munson, D. Quock, B. Chapin, and J. Figueroa, “Argonne’s ATLAS Control System Upgrade”, International Conference on Accelerator and Large Experimental Physics Control Systems, ICALEPCS ‘99, Trieste, Italy, October 4-8, 1999.
[3] Compaq Computer Corporation, Houston, TX, USA.
[4] Kinetic Systems Corporation, Lockport, IL, USA.
[5] Microsoft Corporation, Redmond, WA, USA.
[6] Oracle Corporation, Redwood Shores, CA, USA.
[7] Corel Corporation, Jericho, NY, USA.
[8] Los Alamos National Laboratory, NM, and Argonne National Laboratory, IL, USA.
[9] Linux, A registered trademark of Linus Torvalds.
[10] WIeNeR, Plein & Baus, Burscheid, Germany.

THE CONTROL ARCHITECTURE OF THE DØ EXPERIMENT

J. Frederick Bartlett, Stanislaw Krzywdzinski, Geoffrey Savage, Vladimir I. Sirotenko, Dehong Zhang, FNAL, Batavia, IL60510, USA

Abstract

From a controls viewpoint, contemporary high energy physics collider detectors are comparable in complexity to small to medium size accelerators: however, their controls requirements often differ significantly. D0, one of two collider experiments at Fermilab, has recently started a second, extended running period that will continue for the next five years. EPICS [1], an integrated set of software building blocks for implementing a distributed control system, has been adapted to satisfy the slow controls needs of the D0 detector by (1) extending the support for new device types and an additional field bus, (2) by the addition of a global event reporting system that augments the existing EPICS alarm support, and (3) by the addition of a centralized database with supporting tools for defining the configuration of the control system. This paper discusses the control architecture of the current D0 experiment, how the EPICS system was extended to meet the control requirements of a large, high-energy physics detector, and how a formal control system contributes to the management of detector operations.

1 THE EXPERIMENT

DØ is a high-energy physics experiment located at one of the two collision points of the 1 TeV proton/anti-proton beam of the Fermilab accelerator. The detector is constructed from multiple layers of sensors: (1) a precision inner tracking section consisting of silicon microstrip cylinders and disks, (2) eight cylinders of longitudinal and stereo scintillating fibers, (3) a super-conducting solenoid providing a magnetic field for the inner tracking layers, (4) an electromagnetic preshower section, (5) a liquid argon calorimeter, and (6) an outer muon spectrometer. The experiment has just commenced its second, extended running period that is expected to last until 2007.

2 EPICS AND ITS EXTENSIONS

Following the first running period, which ended in 1995, the computing policy of the laboratory decreed that future experiment software must be developed from platform-independent components. Since the DØ control group was small and the period before the beginning of the next running period was short, recasting the existing slow-controls system in the new formalism was not practical.

After a brief survey of the field, we selected EPICS (Experimental Physics and Industrial Control System) [1] to provide the building blocks for our new controls effort. The principal reasons for selecting EPICS were (1) the availability of device interfaces that matched or were similar to our hardware, (2) the ease with which the system could be extended to include our experiment-specific devices, and (3) the existence of a large and enthusiastic user community that understood our problems and were willing to offer advice and guidance.

One of the unique properties of the DØ detector interface is the use of the MIL/STD1553B serial bus for all control and monitoring operations with the detector and with the electronics components located in the remote collision hall. Since the detector is inaccessible for extended periods of time, a robust, high-reliability communication field bus is essential. We extended EPICS by providing a queuing driver for MIL/SRD1553B controllers and a set of device support routines that provided the adaptive interface between the driver and the standard EPICS process variable (PV) support records. Once these elements were in place, all of the features of EPICS were available for use with our remote devices.

High voltage channel control is an example of extending the basic PV record support. In this case, building a compound device from individual PV records was not feasible because of the complexity of the HV device and the speed requirements. A generic high-voltage record support module was developed based upon the extended, Harel state machine model [2]. The record support module provides the required, high-level behavior with (1) linear ramping with retries, (2) trip condition recovery, and (3) limits control. Device support modules then adapt the HV record to specific HV devices. Although developed for a specific device, the record support is non-device specific and may be used for other types of voltage generators that require a similar behavior.

Using the EPICS portable channel access server, we have constructed gateways to two other control systems: (1) the SCADA-based DMACS system that manages the DØ cryogenic and gas systems and (2) the accelerator ACNET control system. These links are used for exchange of information only.

3 GLOBAL EVENT REPORTING

Although the EPICS system provides an operator alarm display, alarms from slow controls are not the only, nor, necessarily, the most important events. To address this problem we have developed a separate facility, the Significant Event System (SES) [3], to collect and distribute all changes of state of the detector and the data acquisition system.

Unlike the EPICS alarm facility, in which the operator display explicitly connects to each PV, the SES has a central server that collects event messages from sender clients and filters them for receiving clients. Each EPICS IOC connects to the server via a TCP/IP link and all state changes on that IOC, including alarm transitions, are sent to the server. For large physics detectors with hundreds of thousands of PVs, the savings in connect time at startup can be significant.

The alarm class of SES messages receives special handling in the server. The SES server maintains the current alarm state of the entire detector so that receiving clients are able to obtain the current state when they first connect to the server.

In addition to specialized receiving clients that may connect to the server, there are two standard clients: the SES logger and the operator display GUI. The logger has a pass-all filter and writes all messages to a disk file.

In addition its monitoring and logging functions, the SES system provides the means for distributing synchronizing messages to other components of the online software system. For example, global control of the high-voltage system is accomplished by having the individual detector high-voltage programs connect to the SES server for messages that signal changes in the run state of the data acquisition system.

The SES server and most of the receiving clients have been coded in the Python scripting language, while many of the sending clients are coded in C or C++. We anticipate that, for efficiency considerations, the server may require recoding in C++ at some later stage in the development cycle. API’s for SES clients are available in all three languages.

4 CENTRALIZED DEVICE DATABASE

The EPICS databases that configure the individual Input/Output Controllers (IOC) are flat, ASCII files that are read by the IOC’s during startup. The EPICS system additionally provides a higher-level construct, called a template, which is a parameterized collection of record definitions. Generator files, which reference the templates, supply the parameter values to produce instances of these templated devices. While these collections of files are adequate for EPICS initialization, they are not easily accessible to host-level processes that may require the same information.

To address this problem, the DØ experiment centralized the relevant device information in a relational database (Oracle) [4] and provided a family of scripts, written in the Python language, to manage the transformation between the relational database and the EPICS, ASCII-format files.

By providing scripts for bi-directional conversions, it is possible to edit collections of devices (instances of templated devices) by extracting the parameterized devices to a generator file, modifying the generator file with a text editor, and re-inserting the generator file into the relational database. For large collections of devices, this three-stage process is often simpler and faster than using a database editor directly.

In addition to the database management scripts, a WWW browser interface to the relational database is available for initial definition and modification of the relational database entries.

With control system device specifications now centralized in the relational database, they are easily accessible to other host-level processes. This, in turn, has led to a series of extensions to the original database schema to support the needs of other, controls-related processes. An example is the SES operator alarm display that accesses the central device database for obtaining guidance text and action scripts related to specific EPICS devices.

5 DETECTOR CONFIGURATION MANAGEMENT

One of the most complex tasks performed by the control system is the configuration of the detector for specific run conditions. The set of distinct configurations, both for normal, data-taking and for calibration runs, is very large; and, so, the usual technique of uploading a specific detector configuration, once the required conditions are established, and saving it as a file for subsequent downloading is impractical.

For purposes of configuring the detector, it is structured as a tree with nodes at successively deeper levels corresponding to smaller, more specialized organizational units of the detector. The terminal nodes of the tree are, in nearly all cases, single instances of the high-level (templated) devices discussed in the preceding database section. The intermediate nodes of the tree primarily serve to organize the traversal order of the subordinate nodes since the detector is, in general, sensitive to the order in which devices are initialized. The terminal nodes, called action nodes, manage the configuration of a specific, high-level device.

One level of intermediate node, the geographical sector, has a particular significance. These nodes, in most cases, represent the individual read-out crates of the data-acquisition system and are the lowest level in the tree hierarchy in which the nodes are guaranteed to be functionally independent. The load function for these nodes may be executed in parallel, significantly reducing the total time required to configure the detector.

A single program, COMICS [5], coded in the Python language, manages configuration of the EPICS-accessible part of the detector. The tree nodes, both intermediate and action, are specialized instances of a base node class, which defines most of the methods that characterize node behavior. The detector tree structure is defined by a set of configuration files that are Python program segments which instantiate instances of nodes.

6 CONCLUSIONS

Faced with the task of completely rebuilding the slow-controls system of a complex, high-energy physics detector in a limited time, the DØ collaboration selected the EPICS system to provide the component parts from which the system would be constructed. EPICS, itself, has been extended to support a new field bus, and numerous experiment-specific devices. Our experience with EPICS in building the control system has been an overwhelmingly positive one, although, as with many distributed development projects, we found that the user documentation was often incomplete.

By providing the Python scripting language with an interface to the EPICS channel access API, members of the DØ collaboration have been able to write nearly all of the operator interfaces to the experiment in a high-level, object-oriented language. Because Python is fundamentally object oriented and provides a number of high-level language constructs and because programs written in scripting languages tend to be significantly easier to debug, the development time for building the DØ online system was significantly reduced compared with what would have been required had the C++ language been used instead.

REFERENCES

[1] L. Dalesio et al., 'The Experimental Physics and Industrial Control System Architecture: Past, Present, and Future', Proc. ICALEPCS, Berlin, Germany, 1993, pp 179-184

[2] D. Harel, ‘Statecharts: A Visual Formalism for Complex Systems’, Science of Computer Programming’, 8 (1987) 231-274

[3] G. Savage, ‘Significant Event System Tutorial’, Internal DØ document

[4] S. Krzywdzinski, ‘EPICS Oracle Database Tutorial’, Internal DØ document

[5] J. F. Bartlett, ‘COMICS: DØ Detector Download Tutorial, Internal DØ document

TUCT006

H1DCM - A NETWORK BASED DETECTOR CONTROL AND MONITORING SYSTEM FOR THE H1 EXPERIMENT

G. Eckerlin, M. Hensel, S. Karstensen, Deutsches Elektronen-Synchrotron, Hamburg, Germany
S.K. Kotelnikov, I. Sheviakov, Lebedev Physical Institute, Moscow, Russia

Abstract

From Oct. 2000 until July 2001 the H1 experiment at the e-p collider HERA has been considerably upgraded for higher luminosity running. The required modifications for the H1 detector control system have been the object of the H1 detector control and monitoring project. A variety of subsystems, some from the early 1990s as well as new subsystems with modern design and technology are to be controlled and monitored by a common detector control system. In order to reduce the manpower demand for operation and the burden on the on-site detector experts a common control system is proposed that is based on a commercial supervisory control and data acquisition system. In addition to the typical functionality of a modern control system, the choice for the product is driven by requiring a distributed network based system, support of Linux and WindowsNT, an object oriented modeling of the devices and the possibility of H1 specific implementations via an application programming interface. The control system for a CAEN based high voltage supplies, for the superconducting H1 magnet and for the H1 luminosity system successfully started operation in July 2001. Gradually further subsystems will be included. The aim is to enable a single person to supervise and control the experiment and its data taking and to allow remote access for experts. The basic design of the control system is presented and the experience during implementation and commissioning is discussed.

1 INTRODUCTION

After 8 years of operation the e-p collider HERA at DESY was upgraded for higher luminosity running in a shutdown starting at October 2000. At the same time the H1 experiment [1] was considerably upgraded and modified as well. The upgrade was successfully completed and the machine started up in July 2001, aiming for first collisions by November 2001 and regular luminosity operation from January 2002 onwards. During the upgrade several new subsystems have been implemented into H1 and some older systems have been modified.

To study the required modifications for the control system of the H1 experiment, the H1 detector control and monitoring project (H1DCM) was initiated [2]. Several approaches for a common control system were studied including the option of writing a complete control system from scratch in Java or C++. This finally resulted in the proposal to build the common control system on the basis of a commercial supervisory control and data acquisition (SCADA) system. This was mainly due to the fact that limited personnel would not allow the project team to finish a Java based control system with all necessary ingredients in time for the new data taking period. The choice of the SCADA system used for H1DCM was driven by requiring a distributed network based system, support of Linux and WindowsNT, object oriented modeling of the devices, and an open architecture allowing H1 specific implementations via an Application Programming Interface (API).

An extensive study done at CERN for the Joint Controls Project [3] initiated for the LHC was used as a basis for the decision and PVSS-II [4] from the company ETM in Austria was chosen.

2 REQUIREMENTS

It is vital that a new control and monitoring system should reduce the diversity compared to the old control system. Modern controls should allow automatic action and recovery. Remote monitoring and remote expert intervention will improve the efficiency of the data-taking by faster or preventive action when necessary. Limited resources require the minimization of cost and manpower involved both for operation as well as during the development of the new control system. This results in a list of requirements for the new control system of H1.

- Most front-end electronics and hardware of older subsystems had to be kept in place and reused because it would have been too expensive to replace.
- Out-of-date parts of the control system, like old Apple Macintosh with NuBus, have to be replaced by some more modern systems to assure that the control system will survive at least the next 5 years without major change.
- The implementation of the new controls for older systems should be possible with the old controls in place to make the transition as smooth as possible.
- The design of the control system should allow the reuse of existing software that would otherwise require major efforts to rewrite. Therefore the control system should make use of the API to interface older subsystem controls easily via a well defined interface.

- The common control system should enable the running of the experiment under normal circumstances by one person, should perform standard actions automatically and inform experts and shift persons in case of unusual conditions.
- The system should allow experts to configure the controls for the subsystems easily and expert knowledge should be easy to implement into automatic actions.
- The amount of programming effort during the development of the control system should be kept small by the use of the functionality offered by the commercial SCADA system.

3 DESIGN

The design of the H1DCM system is based on the SCADA system PVSS-II. An overview of the PVSS-II structure is shown in figure 1. It is build as a set of separate processes or tasks, called managers, which communicate via TCP/IP connections. The communication is event driven to reduce network traffic. Around a central event manager which serves as a message dispatcher, several dedicated managers are grouped with specific functionalities, like database, device drivers, control script managers, APIs and graphical user interfaces (see ref.[4] for more details).

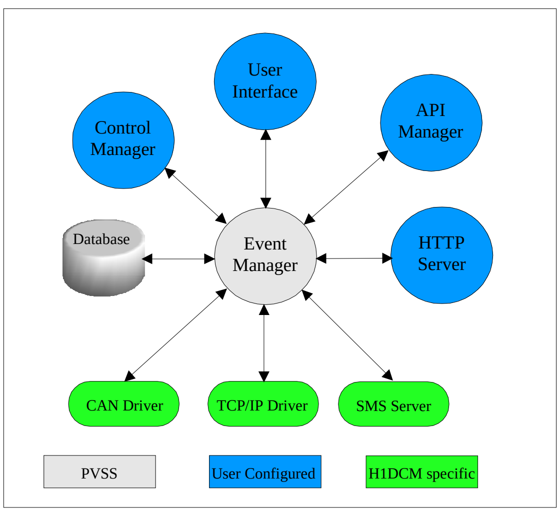


Figure 1: The basic structure of PVSS-II

Within PVSS-II complex devices can be modeled by so-called data point types (DPT) and data points (DP). A DPT defines the structure of a device type, and a data point is an instance of a specific physical device of this type.

By using references to another DPT inside a DPT definition it is possible to build complex devices out of basic types.

The graphical user interface in PVSS-II is built out of panels and the usual items needed for user interaction, like buttons, menus, text fields, tables, graphs. Scripts for complex actions may be attached to panels and the elements inside a panel. The usage of references is also possible for panels, e.g. it is possible to define small reference panels and to build a larger complex panel out of these. This feature enables an effective reuse of code and panels and allows the definition of a common look and feel that can later be easily changed or adapted by changing the basic reference panels.

In addition, PVSS-II comes with the usual functionality of a SCADA system. Furthermore, it allows connection to a remote database, has a built-in HTTP server, automatic backup and can be operated in a redundant configuration.

The H1DCM design makes use of these features to reduce the effort for coding and development wherever possible. It takes advantage of the network based architecture of PVSS-II, which allows all components to run on different machines. This is especially used to distribute the shift and expert user interfaces as well as some drivers and the HTTP server among various PCs. In case of failure of a PC the overall system will stay intact and only a specific part has to be restarted or started on a different PC.

On top of the PVSS-II functionality the H1DCM system implemented a special fieldbus driver for CAN bus modules used for the H1 Solenoid control and a HV client/server which is based on the TCP/IP driver from the PVSS-API package. Extensive use of the control script manager is made to automatically act upon incoming data and to send control messages to the HV server. Two independent HV client/server models have been implemented.

The C based server One HV server is implemented in C and is running on a VME based CPU with either Solaris, LynxOS or Linux. It uses the feature of PVSS to call c-functions inside a control script. The HV client is a PVSS control script and a set of c-functions which use c sockets to communicate with the remote server. The server accepts a set of commands which result in simple write or read actions to the CAEN HV supply via the CAEN VMEinterface A200 [5]. The client is actively requesting the server to read data for monitoring or sends commands to set values or states.

The Java based server A second server was written in Java and is running on a VME based CPU with Linux. It communicates directly with the TCP/IP driver from the API package of PVSS. The incoming messages are ASCII formatted strings with a predefined syntax which are interpreted by the control scripts running in the control manager of PVSS. The server uses a configuration file to load the initial values for the HV system after restart and is monitoring the CAEN system continously. It reports the monitored values and states to the clients and acts on commands received from the clients.

In both cases control scripts are used to process the incoming messages further. Depending on the contents these scripts will automatically set alarms or send messages to other processes or to experts.

The graphical interface is modeled with the PVSS user interface manager. The system allows the display of trends of selected values, like currents or voltages and to send commands to the HV servers for operation, like HV-OFF, HV-ON, by pushing the appropriate buttons.

An automatic control of the HV systems by the H1DCM system is not yet fully implemented. In the automatic mode the system will collect the necessary information from the HERA collider, the experiment components and the DAQ system and will then act on the HV systems. The shift will only be notified in case of unusual conditions or in case an action has to be confirmed or an alarm has to be acknowledged. An automatic sending of SMS messages to experts in case of alarms is implemented.

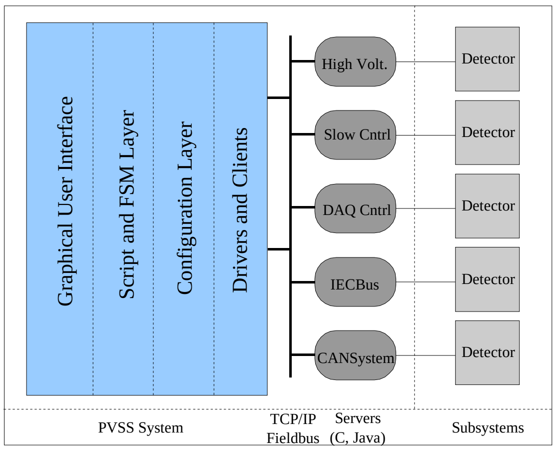


Figure 2: The basic structure of the H1 detector control and monitoring system

Figure 2 shows the overall structure of the H1DCM system. On the right the different subsystems are schematically shown. The connections between the subsystem front-end electronics, which stay untouched for the old systems, and the servers are mainly dedicated point-to-point connections. The servers are mainly located in VME crates. A VME based CPU (currently VMIVME7750 running Linux) is put into the server crate and the server process is implemented. The servers communicate via TCP/IP with the central control system which in turn is divided into 4 logical layers :

- The client layer which communicates with the server and transforms the messages into PVSS internal data formats,
- the configuration layer which enables the detector experts to configure the control system parameters according to the needs of the subsystem,
- the scripts and finite state machine layer which will perform the automatic actions during normal running according to the configuration parameters set by the experts, the external conditions and the requests received from the shift person
- and, finally, the shift control panel, which will enable one person to control and monitor the H1 experiment.

To reduce the effort during development and maintenance the tasks were grouped and responsibilities were combined where possible.

The subsystem experts remain responsible for the detector frontend and hardware.

The server experts, who might be subsystem experts as well, provide the server software.

A few PVSS client experts are responsible for the PVSS client implementation and the definition of the protocol in close cooperation with the server experts and the subdetector experts.

A PVSS script expert is in charge of properly designed and configured scripts and, if needed, of a finite state machine to operate the experiment under normal condition automatically.

One shift panel expert is responsible for the graphical user interface which is provided for the person on shift.

4 CONCLUSION

The control and monitoring of the H1 solenoid was successfully started at the end of the shutdown in July 2001. The operation of the HV systems started in the following cosmic test run with the modified forward tracking chambers and the newly installed central inner proportional chamber. The operation was successful from the start and suggestions for improvements which were raised by the detector experts could be implemented quickly. Additional HV systems were included in the control during the cosmic run which in total lasted 5 days. Including the various systems into the common control proved to be as easy as it was anticipated with the new system and, finally, all tracking chambers that were possible to be operated at that time were monitored and controlled with H1DCM.

The extensive use of reference panels and device types allowed the modification of the functionality in a consistent and efficient way even for a complex running system and kept the effort for maintaining the system low.

Both operating the H1 solenoid as well as the HV systems of the different tracking chambers in H1 proved that the H1DCM system will be capable of taking over the slow control tasks in H1 for the coming data taking period.

5 REFERENCES

[1] I. Abt et al.[H1 Collaboration] , Nucl. Instrum. Meth. A386, (1997) 310.

[2] http://www-h1.desy.de/h1det/h1dcm.

[3] http://itcowww.cern.ch/jcop.

[4] http://www.pvss.com/english/index.htm.

[5] http://www.caen.it.

TUESDAY, 27 NOVEMBER 2001

TUD - PROJECT ENGINEERING AND MANAGEMENT

TUDT001
cs.SE/0111013

QUALITY CONTROL, TESTING AND DEPLOYMENT RESULTS IN NIF ICCS

John P. Woodruff, Drew D. Casavant, Barry D. Cline, Michael R. Gorvad
LLNL, Livermore, CA 94550, USA

Abstract

The strategy used to develop the NIF Integrated Computer Control System (ICCS) calls for incremental cycles of construction and formal test to deliver a total of 1 million lines of code. Each incremental release takes four to six months to implement specific functionality and culminates when offline tests conducted in the ICCS Integration and Test Facility verify functional, performance, and interface requirements. Tests are then repeated on line to confirm integrated operation in dedicated laser laboratories or ultimately in the NIF. Test incidents along with other change requests are recorded and tracked to closure by the software change control board (SCCB). Annual independent audits advise management on software process improvements. Extensive experience has been gained by integrating controls in the prototype laser preamplifier laboratory. The control system installed in the preamplifier lab contains five of the ten planned supervisory subsystems and seven of sixteen planned front-end processors (FEPs). Beam alignment, timing, diagnosis and laser pulse amplification up to 20 joules was tested through an automated series of shots. Other laboratories have provided integrated testing of six additional FEPs. Process measurements including earned-value, product size, and defect densities provide software project controls and generate confidence that the control system will be successfully deployed.

1 INTRODUCTION

The Integrated Computer Control System (ICCS) software for NIF is being constructed by an iterative process that implements and tests specific functional increments as required by project management. Planned overlap between cycles of development allows management to level the effort of design and review, implementation, and test activities to utilize staff efficiently. Performance measures taken during each cycle show progress toward completion. Each increment is first tested in the ICCS testbed. Presently, this is followed by testing in the appropriate prototype laboratories.

2 ITERATIVE DEVELOPMENT CYCLES

Each planning cycle begins by identifying which requirements will be implemented for the increment. ICCS managers review project needs and select from the integrated project requirements documents those functions appropriate for the next development cycle. Availability of testbed resources, project risk resolution and prospects for integration with laboratory equipment influence the selection of requirements that are included.

The product of the requirements phase is an implementation plan: a document that defines the requirements, the changes that have been accepted into the software change request database, new features on the user interface, and dependencies on other subsystems. The implementation plan also establishes the schedule for the increment and defines which of the ICCS subsystems [1] will be deployed at the end of the increment. Earned value accounting allocates value for the product of each engineering phase, so the value accrues steadily through the construction of the increment.

Detailed design work on the increment's components begins when the implementation plan is complete. A review of software designs precedes implementation. Each of the subsystems in a deployment is reviewed prior to code implementation, with reviewers' action items tracked to closure. Included in the documentation presented in a review are Unified Modeling Language representations of the software classes to be built, interface definition language interfaces for CORBA-distributed objects, schemae for database tables, Buhr diagrams [2] showing concurrency, and user interface sketches.

Implementation and unit test are accomplished using an integrated development environment that incorporates version control and change management with the Ada compiler and the Java development suite.

When the several interacting programs that constitute a testable product release are completed, integration and deployment is performed by the configuration management team, independent of the implementers. The configuration manager performs quality control verification by witnessing compliance with the implementation plan. Requested software changes that were completed are documented via the change control database. The source code is stabilized by the version control tool. All versions are confirmed to be consistent, and the build process is repeated under configuration control. Completeness is confirmed by executing a set of integration tests that confirm interface consistency of all the communicating processes. The entire deployment is copied to the test environment and turned over to the test team.

3 TEST PLANNING AND EXECUTION

The test team's responsibility is to confirm that the delivered software correctly and robustly implements the requirements established for the increment. The testers design, execute, and document the results of extensive tests. Test plans and test procedures are written starting with the requirements, and accepted changes are documented in the implementation plan.

One or more tests are conducted for each deployment, with each test typically encompassing numerous test cases. For requirements traceability, test procedures identify the requirements being verified and the relevant procedural steps. Test procedures are redlined as required during test execution to provide an accurate basis for test expansion for subsequent deployments as well as regression testing.

Tests are conducted "offline" in a dedicated Integration and Test (I&T) facility, where representative laser hardware is available for extensive testing. In the I&T facility, the software can be exercised at the limits of its capabilities and error cases introduced to assure robustness. Testers also draft operations manuals and oversee preparation of configuration database instances that support activities in the I&T facility and in laser hardware prototyping laboratories.

Fully integrated deployments can be tested online in a laser laboratory. During both offline and online testing, incidents of noncompliance are recorded and analyzed for trends. Some tests expose defects in software: erroneous requirements, functional errors, errors of omission, and regressive failures are tracked. Software change requests are entered and managed by the SCCB. Hardware defects are documented and tracked using a similar process.

So far, over 650 test incidents have been documented; about three-quarters of these are software issues that result in software change requests (SCRs). Defects that prevent completion of critical testing or operations are classified as urgent and are repaired by the development team as quickly as possible. A patch to the deployment is then issued to fix the problem, and regression tests are performed. Software defect density to date has been approximately 2 functional defects per 1000 lines of code. The success rate for repaired defects is approximately 90%. New defects have been introduced in less than 10% of the patches.

4 OVERLAP OF SUCCESSIVE INCREMENTS

The foundation subsystems – the frameworks and the support layers where commercial off-the-shelf products are installed – lead application development by half a phase. The benefit of this tactic is that new framework functionality is specified ahead of need. Framework requirements are defined immediately after the application subsystems have completed their design reviews (in the preceding cycle), and the implementation of framework enhancements occurs while the succeeding application systems are being specified and designed. This tactic delivers (partially) tested frameworks to application developers just as they begin their intensive implementation activities.

Overlap between cycles is also exploited in the planning process. When subsystem design reviews are complete and the test plans and user documentation are available in draft form, the next cycle of preliminary planning starts. Thus, planners who learn of difficulties in realizing an implementation plan can adjust the next increment accordingly.

5 DEPLOYMENT VARIATIONS

The architecture of the ICCS is conducive to the phased implementation strategy because substantial amounts of functionality can be developed independently for the several functional subsystems. This allows distinct deployments of ICCS components as laboratories become available for online tests. Furthermore, all the subsystems rely on a common set of framework software [3] that itself is undergoing incremental refinement. Status propagation and display [4] are examples of functions that different subsystems implement independently while using the common ICCS framework.

These considerations lead to some tactical decisions about what steps of development should be executed for the different subsystems. Since the ICCS is a loosely coupled collection of subsystems, some deployments include only partial functionality: an example is the target diagnostic subsystem that is very loosely coupled to the laser controls. Standalone tests of target diagnostic functionality have been performed in the absence of other subsystems.

7 SOFTWARE CHANGE MANAGEMENT

Required functionality for an ICCS increment must respond to the project schedule. Management decisions based on the required functionality thus drive the development cycles. Another source of work for the development team is changes that are requested by a variety of stakeholders such as testers who report defects, developers who evolve internal interfaces, operators and project customers concerned with human factors. By convention, these two sources of work are kept distinct since increments of functionality are taken to be purely new code, while changes are expected to modify code that has already been tested. An SCCB manages the disposition of all requested changes.

SCRs are accepted into a managed database from any interested party; most arise either from defects exposed during testing or from evolution within the development team. The data associated with the SCRs allows the SCCB to know the status of each authorized change: what is the next step in satisfying the request, who is responsible for that activity, and when is completion expected? The final step in a change process is regression testing, which occurs when an incremental product is delivered. About 1100 SCRs have been considered since the SCCB was formed, and half of these are still incomplete.

6 EXPERIENCE WITH INCREMENTAL DEVELOPMENT

The ICCS has been under development since project inception in 1998; seven cycles of code release and test have been completed. The size of the successive releases has grown from 89 thousand source lines of code (KSLOCs) to a present inventory of 322 KSLOCs. The estimate of the product size at completion is about 1000 KSLOCs.

Early ICCS releases have been deployed to a variety of different destinations for testing. Because the NIF facility itself is not ready to receive control system software, the project has constructed two successive generations of I&T facilities. The present 2,400-square-foot facility houses 9 Unix workstations and 23 racks of electronics. Generally, one of each type of controls hardware module that will be used in the NIF is represented in the I&T facility. For example, rack-mounted equipment includes data and application servers, network switches, FEPs, programmable logic controllers, timing system components, motor controllers, emulators, and transient recorders. A lab table is used to mount motors, shutters, photodiodes, and other device points for functional testing.

Offline testing accomplished in the I&T facility is, for many subsystems, augmented by testing in dedicated laser subsystem prototyping laboratories. Controls deployed into these labs are integrated with laser hardware, shaking out interface issues and providing operations personnel the opportunity for controls validation and training. Integration tests in the prototyping labs are also used to proof installation and checkout procedures used for controls deployments into the NIF.

The most extensive online test to date has been accomplished in the Front-end Integration System Test (FEIST) laboratory, where the prototype laser preamplifier, input laser diagnostics sensor [5], and timing system are assembled. The control system installed in the FEIST lab contains five of the ten planned supervisory subsystems and seven of sixteen planned FEPs. Beam alignment, timing, diagnosis, and laser pulse amplification up to 20 joules was tested through an automated series of shots on the preamplifier. Other laboratories dedicated to wavefront control [6], pulsed power conditioning, and Pockels' cell testing have provided integrated testing of three additional supervisors and six additional FEPs.

This work performed under the auspices of the U.S. DOE by LLNL under contract No. W-7405-Eng-48.

REFERENCES

[1] L. J. Lagin et al., "The Overview of the National Ignition Facility Distributed Computer Control System," ICALEPCS 2001.

[2] J. A. Buhr, "System Design with Ada" Prentice Hall 1984.

[3] Robert W. Carey et al., "Large-scale CORBA-distributed software framework for NIF controls," ICALEPCS 2001.

[4] Kirby W. Fong et al., "Application Software Structures Enable NIF Operations," ICALEPCS 2001.

[5] R. D. Demaret et al., "Beam Diagnostics Systems for the NIF," ICALEPCS 2001.

[6] Lewis Van Atta et al., "The Wavefront Control System for the NIF," ICALEPCS 2001.

TUDI002
physics/0112051

MANAGEMENT OF A LARGE DISTRIBUTED CONTROL SYSTEM DEVELOPMENT PROJECT*

D.P. Gurd, SNS LANL/ORNL, Oak Ridge, TN 37830 USA
{*} Work supported by the US Department of Energy under contract DE-AC05-00OR22725

Abstract

Building an accelerator at six geographically dispersed sites is quite mad, but politically expedient. The Spallation Neutron Source (SNS), currently under construction in Oak Ridge, Tennessee, combines a pulsed 1 Gev H^- superconducting linac with a compressor ring to deliver 2MW of beam power to a liquid mercury target for neutron production [1]. Accelerator components, target and experimental (neutron-scattering) instruments are being developed collaboratively by Lawrence Berkeley (Ion Source and Front End), Los Alamos (Linac), Thomas Jefferson (Cryosystems), Brookhaven (Compressor Ring), Oak Ridge (Target and Conventional Facilities) and Argonne (Neutron Scattering Instruments) National Laboratories. Similarly, a team distributed among all of the participating laboratories is developing the EPICS-based control system. This paper discusses the management model and strategies being used to address the unusual issues of organization, communication, standardization, integration and hand-off inherent in this widely-distributed project.

1 INTRODUCTION

The SNS control system presents no special or unique technical challenges. It is being developed using a standard, flat, EPICS-based architecture [2], using linux-based upper layer clients, consoles and servers; distributed VME- and VXI-based input-output controllers (IOCs) with Motorola 2100 series Power PC processors and a PLC or field-bus I/O layer for process control-like subsystems [3,4]. The timing and synchronization system is based upon RHIC hardware [5], which implements concepts originated at Fermilab perhaps twenty years ago. The PLC-based personnel safety system [6] is modeled upon a similar system in use at Jefferson Lab. The communication network is based upon now-standard switched Gigabit Ethernet [7]. The most original aspect of the SNS architecture is the use of PC-based "Network Attached Devices" (NADS), developed by the Beam Diagnostics team for beam instrumentation [8]. These NADS are designed to look to the control system like EPICS IOCs.

Implementing these more-or-less conventional systems using teams distributed across the country, belonging to laboratories each of which brings its own culture and approach, does however present a unique and interesting challenge. Conventional management approaches, organization, and communication methods must all be adapted to the realities of the partnership.

There have been successes and failures, but the effort is of interest because it seems likely that for political and economic reasons future large projects will also be built as collaborations. These projects may well be international in scope, and the issues thereby exacerbated. What is learned at SNS should be useful.

2 MANAGEMENT AND ORGANIZATION

The original SNS proposal had each of the participating laboratories responsible for the control system for its individual part of the machine – Berkeley for the Front End controls; Los Alamos for the linac controls; Brookhaven for the Ring controls, etc. While this model insured a tight coupling between individual control systems and the subsystems they controlled by making the partner laboratories responsible for both, there was risk that the ultimate integration of disparate control systems would have been difficult or impossible; and there was nothing in the model that allowed for the development of the "global systems" which are common to all – the network, the timing and synchronization system, the equipment protection system, a common control room, etc.

Eventually, a management model evolved that included the "Integrated Control System" (ICS) at the same organizational and reporting level as each of the six principle facility components – Front End, Linac, Ring, Target, Instruments and Conventional Facilities (the Physical Plant.) Each partner laboratory has a controls team and controls team leader that reports to the central (ORNL) controls team management. The global systems are themselves distributed, but managed centrally from Oak Ridge. This arrangement facilitates standardization and eventual integration, but requires that more effort be made to assure that subsystem developers and partner laboratories pay attention to the requirements, schedule imperatives and integration of their parts of the control system, which have become someone else's responsibility and are in someone else's budget.

One strategy to encourage integration at this level has been to include the subsystem controls schedule as a part of each system "sub-project" schedule, managed and "statused" at the partner laboratories and fully integrated with their schedules. The goal is to make subsystem designers more conscious of the control system support they require, and when. This is always a problem, and it is not clear that we have done any better than is usual. The downside is that there exists no separate control system schedule maintained centrally by the controls team. Understanding and reporting of schedule status and cost performance is a complicated and manpower-intensive effort requiring the integration of six different reports.

An important key to success in any project is good communication. This is rendered even more difficult, and more important, in a collaborative project such as SNS. The controls team has attempted to mitigate this problem with regular (weekly) teleconferences involving the controls team leaders at each of the participating laboratories. Those meeting originally included the use of "NetMeeting," but that became problematic with increased computer security at the DOE laboratories. Indeed laboratory firewalls have made all exchange of technical information awkward and inconvenient at best, impossible at worst. SNS has installed state-of-the-art videoconferencing facilities at each of the partner laboratories. These are used extensively, and are invaluable; however nothing satisfactorily takes the place of face-to-face discussion, and travel is an inevitable but necessary (and expensive) concomitant to successful collaboration. The SNS controls team was able to take advantage of this conference to have 23 members present at a controls team meeting. What might be a weekly occurrence under more conventional circumstances may be the first and last occasion for this group to be together for the duration of the project.

3 STANDARDS

Controls team leadership in the area of standardization has been a model for the rest of the project. Implementation and enforcement of standards in several areas, including software, hardware, screen design and device and signal naming was recognized very early as the linchpin of our integration approach.

3.1 Software

Software standardization has been difficult. In order to insure uniformity across all developed software, the SNS project negotiated project-wide licensing agreements. Some local sales organizations, however, have been reluctant to recognize these contracts, feeling that they have not got their "piece of the pie." This is of course a corporate problem, but one that nonetheless has resulted in delays and frustration for SNS implementers.

The most obvious, most important and most successful standard is the uniform use of EPICS for all subsystem controls. Contrary to tradition even in EPICS laboratories, this includes both the conventional facilities and the target control systems, where integration, often late, loose and ad hoc, was deemed important from the outset. Training was required for both commercial firms and partner laboratories not familiar with EPICS. Current development is taking place under EPICS v3.13, however the target version will be v3.15, which will have a number of capabilities added specifically for SNS [9].

EPICS itself allows a number of choices, and a suite of standard EPICS tools has been selected. This toolkit includes the "Extensible Display Manager" (EDM), developed for EPICS at the Oak Ridge Holifield facility, and which is being further developed in collaboration with the SNS controls team. EDM was chosen for easier maintenance and extensibility than competing EPICS display managers, and tools have been developed to translate screens developed in two of these: "MEDM" and "DM2K."

Working with the operations team, SNS has standardized on layouts and color use for operator screens. The EDM color rules capability facilitates this, allowing predefined colors to be selected by names such as: "linac background."

Linux has been chosen as the operating system for development, as well as console and high-level server applications, and nearly all utilities are available in this environment. Unfortunately VxWorks, the standard EPICS IOC kernel, still must be developed under Solaris, so the standard is not universal.

EPICS is oriented to individual signals, and does not provide a higher-level "device" view of the accelerator convenient to accelerator physics programmers. SNS is using a class library known as XAL [10] to address this requirement. In addition, temporary "ad hoc" programs can be written using Java-based Python scripts or in Matlab, either of which have direct access to any process variable in the control system.

The project early agreed on the use of Oracle™, with the goal of a fully integrated technical database relating device and signal tables for the generation of the EPICS distributed databases, lattice and modeling data for physics use, technical data on all equipment for tracking and maintenance, magnet measurement and other calibration data, a cable database, and more. A success for standardization? Well, not really – at least not yet. The usual difficulties arose. Many specialized databases were already in use. The schema did not correctly address all the issues in all of the diverse areas. Engineers were unwilling to give up control of their already useful databases to a central, and not always responsive, authority. Tools could not be centrally developed at the same rate as tools being developed in parallel at the partner laboratories. The result has been a struggle to incorporate existing databases and, so far, only moderate success in the use of what has become known as the Grand Unified Relational Database (GURD).

The most important tool for standardization and eventual integration of software developed at the partner laboratories is the requirement that all software be maintained and configuration managed out of a CVS repository located at Oak Ridge. The initiative has been moderately successful, and in spite of some resistance to developing in an environment far from home, more and more distributed developments are being deposited in the central repository. This should assure that all otherwise independent developments are using the same versions of the same tools, and avoid a potential integration nightmare when they all come together for final commissioning.

3.2 Hardware

SNS has facilitated the use of hardware standards by means of "Basic Ordering Agreements (BOAs)," which allow all partners, subcontractors and vendors to purchase selected standards at project-negotiated prices. This has worked fairly well, although intervention by the project is frequently required when agents for selected vendors do not or will not recognize negotiated prices as applying to them.

PLCs: The SNS control system makes far greater use of commercial Programmable Logic Controllers (PLCs) than is traditional in EPICS-based systems. PLCs are used for subsystems that must be kept operating whether or not the rest of the control system is needed, such as the cryogenic and vacuum control systems. In addition, the inclusion of traditionally PLC-based process systems such as those for conventional facilities and the target added even more PLC-based systems. SNS selected the Allan-Bradley ControlLogix™ family of PLCs for these applications.

Far more difficult has been the imposition of standards for the programming of PLCs. This is in part because many PLCs are in fact vendor-provided, often with pre-existent software. Related to programming, is the problem of divergent approaches to the use of PLCs. The guideline has been to use PLCs for interlocks only – whatever is needed to keep systems running safely, even when the EPICS control system might be down. All other functionality, such as automated procedures and operator displays, were to be implemented in the IOCs. This practice has not been universally adhered to.

IOCs. SNS has standardized on the Motorola 2100 Power PC series of processors for its distributed IOCs. From this family, an appropriate configuration can be selected for the application. An adapter card allows the same processor to be used for both VME and VXI applications. BOAs have also been established for VME and VXI crates: Dawn for 7 slot VME crates; Wiener for 21 slot crates and Racal for VXI.

Racks. A BOA has also been put in place for standard, 19" equipment racks. These may be configured as required with doors, side-panels and/or other accessories. The concept of a "rack factory" for assembly of equipment racks is not new – in house rack assembly facilities were established at SLAC for PEP II and at Jefferson Lab for CEBAF. SNS has signed a "rack-factory" contract with an electronic assembly company close to the SNS site to allow equipment designers, only if they choose, to have their equipment racks assembled at this facility on a "task-order" basis.

Racks would seem to be simple indeed. However even here imperfect communication exacerbated by distance resulted in a serious misunderstanding of significant consequence. Draftsmen at one partner laboratory misinterpreted the depth of the specified standard rack. As a result, an entire facility was laid out with racks that were too small. When this was eventually caught, all rack space had been allocated, some equipment would not fit the smaller rack and there was no room to add new rows of racks. The resulting compromise was ugly and not entirely satisfactory. This problem would likely not have occurred if all the principals were in one place.

3.3 Names and Database

Perhaps the first standard agreed by the partner laboratories was for signal and device naming. It has also given the most trouble. An apparently simple hierarchical standard was defined as shown below:

SystemName:DeviceName:SignalName

The names were to be mnemonic, long if needed for clarity, and optimized for operations. Instantiation schemes were defined for the linac and ring. Example lists of lattice devices were given. A document was signed and approved by all – so early in the project that no one needed it or used it. A state of euphoric innocence prevailed. Then reality intervened.

The concept hierarchically related each "signal" to a corresponding "device." The originators of this hierarchical idea intended that that device be an "accelerator" concept, such as a quadrupole, or a cavity, or a klystron; and that the signal be one of its observable properties. However signals can reasonably be associated with other types of devices – the transducer that produces it for example, or the VME ADC module that it goes to. All of these devices also need a place in the database for purposes of tracking and maintenance. The naming standard originators intended that these devices would be related to the signal using the relational database (although the original schema did not in fact do this), but that the signal *name* would use the "accelerator" device. As the design proceeded, poor communication, exacerbated by the difficulty of quickly detecting misunderstandings across several national laboratories, resulted in names being created that used any and all possible related devices in the signal name.

Different teams in different laboratories tried to apply the standard to their own subsystems. The scheme, which worked well and easily for lattice devices, was not so easy to apply to off-lattice devices such as a cryoplant or a cooling system. The hierarchical approach was foreign to engineers trained in the process control industry, who wished to relate SNS names to PLC "tag-names" formulated according to industrial standards. Engineers disagreed over what belonged in the device and signal fields, and how to apply the instantiation rules. Finally, a desire to contain all SNS technical information in an Oracle-based relational database, and to use this database (among other things) to produce the distributed control system database, imposed new requirements of parsability on the names that had not been considered in the standard.

By now, "official" names using several different interpretations of the original standards document appeared on drawings, screens and in documents and prototypical databases. This situation still prevails. Some names have been changed to conform to the intent of the standard, but this has not been done where an adverse schedule impact would have resulted. It may become necessary to make some changes later in the project, which will be both expensive and painful.

4 INDUSTRIAL PARTICIPATION

A controversial goal of the SNS approach was to fully integrate the "Conventional Facilities" controls from the outset. It is often the experience that these physical plant control systems (HVAC, power, etc) are provided by the general building contractor using technology quite different and incompatible with the accelerator control system. Later, during operations, it is found that process variables from these systems are needed in the control room, for observation or for correlation, and they are not easily accessible. Not without considerable controversy and opposition, SNS mandated that the conventional facility controls should be implemented from the outset in EPICS. To conform to traditional practice, this EPICS-based system would use PLCs at the I/O layer, and be implemented by a commercial contractor familiar with industrial control systems.

The Sverdrup Technologies controls team based in Tullahoma, Tennessee was awarded this contract. A weeklong EPICS training session was set up at Tullahoma, and this team is currently developing the distributed databases and human interface screens using the SNS-standard EPICS tools. As EPICS is based upon the same ideas as industrial control systems (the "I" in EPICS stands for "Industrial") the Sverdrup team seems quite at home in this environment, and is progressing well. They are familiar with PLC technology as well, and so can produce a fully integrated system, top to bottom. It is the plan to use the same approach and team to deliver both the EPICS and PLC portions of the Target control system. In addition to the obvious advantage of seamless integration of conventional with accelerator controls, this approach has made available an experienced commercial EPICS-trained team, which will be available later in the project to assist when there are resource shortages or schedule "crunches."

For more on the details of this arrangement, both technical and contractual, see reference [3].

5 HANDOFF

A particularly interesting challenge for the collaboration is the development and eventual implementation of plans to hand over to the SNS engineers and physicists at Oak Ridge complex subsystems developed at the partner laboratories. This applies to all subsystems, and is especially challenging for controls, where the systems include both hardware and software that might never have been fully integrated where they were designed. The project has developed a "Lead, Mentor, Consult" model for the handoff process, in which the partner laboratory responsible for the design of a subsystem takes a lead role for the design and for the installation and testing of the first subsystem; then allows Oak Ridge personnel to install and test the next subsystems, while taking an active mentoring role; and finally returns home to leave installation and testing of later subsystems to Oak Ridge personnel, while remaining available for consultation if needed. A detailed installation plan is in place that adopts this approach.

Two "facts of life" have made it difficult to implement this plan in an entirely rational and consistent manner. First, the SNS budget plan did not adequately account for anticipated pre-operations expenses. This resulted in pressure to move some money from the partner laboratories to SNS, thereby compromising their “lead” and “mentor” functions in some cases. Secondly, each of the partner laboratories have interpreted their responsibilities under this plan somewhat differently. The result is that while the controls team at Oak Ridge still expects considerable help with installation and testing of Linac subsystems, it expects to be more on its own in installing the Ring. These variations in approach have made it difficult to plan for staffing levels during the installation phase. One very great advantage of the collaborative model for project building is that the partner laboratories can serve as both source and sink for the extra staffing requirements of the construction phase. This benefit is being somewhat reduced by the need to increase staffing at Oak Ridge for pre-operations, at the expense of partner laboratory staff.

6 CONCLUSION

Construction at the Spallation Neutron Source site in Oak Ridge is proceeding on schedule. The “Front End” building will be complete at the end of May, and the Ion Source, RFQ and Medium Energy Beam Transport systems will be delivered in June and July of 2002. Installation of Linac components will begin in the fall of the same year. The project appears to be on track to deliver its first neutrons in December of 2005.

The control system should be ready to support installation, testing and commissioning of the various subsystems as they are delivered. The Front End already operates with beam at Berkeley, using a prototypical EPICS control system. “Hot Model” tests were supported by the controls team at Los Alamos, and prototypical controls subsystems are under development at the other partner laboratories. The refrigeration plant will be installed in the summer of 2002, and an EPICS control system will be ready. Control for “conventional facilities” is being developed under contract by a commercial vendor using EPICS, and each building, when handed over, will include an easily integrated EPICS control system for heating, ventilation, air conditioning, power, etc.

Like the facility itself, the control system for the SNS is being developed by a multi-laboratory collaboration. This presents unique management and organizational challenges. Attempts have been made to address these challenges in various ways, and with varying degrees of success. We are winning some, and losing some, and learning as we go; but the most important thing that we are learning for future projects is that there is no fundamental reason one cannot build and integrate a complex control system using many widely distributed partners. And have fun doing it.

REFERENCES

[1] SNS Website: http://www.sns.gov/

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

[3] This Conference: TUAP055. Use of EPICS for High-Level Control of SNS Conventional Facilities. J.K.Munro, Jr., J.C.Cleaves, E.L.Williams, Jr., D.J.Nypaver (ORNL), K.-U.Kasemir (LANL) and R.D.Meyer (Sverdrup Technologies)

[4] This Conference: WEAPO13. The SNS Cryogenic Control System: Experiences in Collaboration. W.H.Strong, P.A.Gurd, J.D.Creel and B.S.Bevans,

[5] This Conference: FRAT001. The SNS Timing System. B. Oerter, J.R.Nelson, T.Shea, C.Sibley

[6] SNS Preliminary Safety Assessment Document. Internal SNS Document.

[7] This Conference: TUAPO55. Plans for the Spallation Neutron Source Integrated Control System Network. W.R.DeVan and E.L.Williams

[8] PAC 2001 Proceedings, Session ROAB. SNS Beam Instrumentation and Challenges. T. Shea

[9] This Conference: THAPO14. Next Generation EPICS Interface to Abstract Data. J.Hill and R. Lange

[10] This Conference: THAPO60. SNS Application Programming Plan. C.M.Chu, J.Galambos, J.Wei and N.Malitsky

TUDT003
physics/0111079

CONTROL SYSTEM DESIGN PHILOSOPHY FOR EFFECTIVE OPERATIONS AND MAINTENANCE*

M. H. Bickley, K. S. White, Jefferson Laboratory, Newport News, VA 23606, USA

Abstract

A well-designed control system facilitates the functions of machine operation, maintenance and development. In addition, the overall effectiveness of the control system can be greatly enhanced by providing reliable mechanisms for coordination and communication, ensuring that these functions work in concert. For good operability, the information presented to operators should be consistent, easy to understand and customizable. A maintainable system is segmented appropriately, allowing a broken element to be quickly identified and repaired while leaving the balance of the system available. In a research and development environment, the control system must meet the frequently changing requirements of a variety of customers. This means the system must be flexible enough to allow for ongoing modifications with minimal disruptions to operations. Beyond the hardware and software elements of the control system, appropriate workflow processes must be in place to maximize system uptime and allow people to work efficiently. Processes that provide automatic electronic communication ensure that information is not lost and reaches its destination in a timely fashion. This paper discusses how these control system design and quality issues have been applied at the Thomas Jefferson National Accelerator Facility.

1 INTRODUCTION

Jefferson Laboratory uses a control system based on the Experimental Physics and Industrial Control System (EPICS). At the lab, a number of different physical plants are controlled with EPICS, including an electron accelerator, a free electron laser, a helium liquification plant, and three experimental end stations[1]. In order to control the plants, the lab utilizes more than 50,000 I/O control points on 100 front-end computers. Operating these machines around the clock involves many operators and support personnel. As the number of hardware devices and users increases, the effort needed for coordination and communication can increase exponentially.

To combat these potential inefficiencies, Jefferson Lab has developed its control system with a philosophy that emphasizes the importance of enabling all users to work more effectively. This paper examines three broad categories of control system usage: operations, maintenance, and development. For each category, the paper will discuss administrative choices, support tools and aspects of control system design that contribute to improving user effectiveness and streamlining communications.

2 OPERATIONS

2.1 User Interfaces

The most basic interaction between users and the control system is through synoptic displays. The displays allow operators to monitor control system data points and modify machine parameters. When there is no consistent user interface design, the job of working with the control system is more difficult. At Jefferson Laboratory, several steps were taken to enable operators to work more effectively with MEDM, our synoptic display program. Interface developers have tools and an administrative framework that gives them the freedom to innovate, while ensuring that their products integrate well with the rest of the control system.

The first piece of the administrative framework was the development of interface standards. To ensure consistency, interface developers are given a palette of colors including obvious choices, for example red to indicate alarm conditions and yellow to indicate warnings. When screen items are user-modifiable their background is light blue, and if they are for display only they are dark blue. All interfaces include a color-coded title bar. This gives users a quick way to identify the system the screen controls.

To ensure that user interfaces support operations well, machine operators do a significant fraction of the interface development. They typically begin with screens provided by the device control developer. These screens are useful for managing a single device, or for use by experts, but are often too detailed to be

* This work was supported by the U.S. DOE Contact No DE-AC05-84-ER40150

used for operations. An operator modifies the low-level control screen and generates higher-level interfaces that, for example, show how the device integrates into the control system. An interactive GUI builder provides operators with the ability to easily develop interfaces optimized for their needs. The original expert screens remain available for debugging.

As an additional aid to interface development, Jefferson Laboratory has created a library of object code that is used to generate synoptic display files programmatically. The library is useful with systems that include a large number of similar devices. For these systems, high-level displays can contain thousands of EPICS Process Variables. Rather than building screen files tediously by hand, one can easily write a program that generates a detailed screen using input from a data file. This is so simple that screen designers can invest the effort to create a very useful screen layout; confident their work will not have to be redone if the machine hardware is modified.

By using standards to ensure consistency, and providing tools for operators to customize screens for their needs, the task of interacting with the control system is simplified. Users can spend their energies understanding the meaning of the data, rather than deciphering its presentation.

2.2 Communications

Operating and maintaining Jefferson Lab's accelerator involves over 200 people in various groups. Coordination of such a large number of people can be problematic, as a great deal of information must be organized and made available to all staff members. Good communication is vital in using the control system to support smooth operation of such a large machine. To facilitate this, the laboratory has developed an electronic logbook (Elog) that is closely integrated with other control system tools. The Elog is used to ensure that all staff have easy access to information about the status of the accelerator.

An electronic logbook is superior to a paper version in several respects: It allows many people to generate and read entries in parallel. An electronic logbook can be updated via many methods, including automated generation of entries, web-based interfaces, and with customized log entry tools. The Jefferson Lab Elog can also be examined with many tools, including search engines, database forms, and a standard web browser.

For operations, the automatic log entry feature is an important capability. A significant fraction of the operators' time is spent recording information, to provide an accurate record of machine operations. Automated logging tools enable operators to efficiently make detailed entries with minimum distractions. These entries usually occur as a secondary output of a program that changes machine parameters or makes a measurement. This saves the operations staff time, since they do not have to make manual entries.

The most commonly used log entry tool is a custom interface named "dtlite". One of its features is the ability to capture X-windows screens. If an operator wishes to record the information presented in any window, he can associate the screen with a log entry. This enables him to show exactly what he sees at any time for later analysis. When encountering a problem, the operator can save the information quickly, with two mouse clicks, and move on to other tasks.

On occasion, while running the accelerator, the operations staff will identify a problem. They collect information about the problem, generate an Elog entry containing a text description and graphics if needed, and can, with dtlite, ensure that the problem will be addressed for maintenance. The same information that is placed in the Elog is added to a trouble-tracking database, called NEWTS. By selecting the appropriate button, the NEWTS entry can automatically record machine downtime associated with the reported problem. The operators do not need to duplicate any data entry efforts, because the Elog, NEWTS and downtime systems are integrated.

Jefferson Lab has used the Elog and other communication tools to help all control system users to be more effective. By ensuring that there is a consistent effort to address user communication issues during software design, the control system continuously enhances operational communication.

3 MAINTENANCE

Scheduled maintenance activities at Jefferson Laboratory include work to upgrade and enhance hardware or software, or to add or remove components. The lab has few scheduled maintenance opportunities, so it is vital to work efficiently. The result is that there is a strong motivation to do as much work as possible during a short period of time. It is also important to ensure that different maintenance tasks do not interfere with each other. For example, if one group is upgrading hardware in a portion of the machine, an unrelated software enhancement in the same area should not be attempted at the same time. A single person coordinates all accelerator maintenance activities to ensure that conflicts are minimized.

All support staff interested in performing repair, maintenance or upgrade work, submit their proposed activities to the maintenance coordinator. Control system tools are an integral part of this coordination effort, by enabling users to generate work plans that are automatically forwarded through the system.

Software maintenance is organized through the development of written test plans generated with web-based tools, which include several different templates. The templates provide button selections for common information and procedures for standard processes. The author lists the enhancements that will be achieved with the modification, what features it has, the expected test duration, and the steps required to install the upgrade and roll it back in the event that there are problems. The test plan is submitted, becomes viewable via the web, and an email notification is sent to a reviewer, who determines if plan is complete and reasonable, and either approves or returns it. If a test plan is returned, the reviewer indicates the deficiencies, and the author receives an email notification so that the plan can be improved. When approved, the test plan information is forwarded by email to the maintenance coordinator, and the work is scheduled. For each test plan completed during a maintenance period, an Elog entry is made that includes the test results The Elog then has a complete record of all of the maintenance activities.

As mentioned earlier, the Jefferson Laboratory control system spans a number of different physical plants. The control system is segmented so that each of the separate plants can function independently. The separate segments called "fiefdoms," make it possible to provide a consistent suite of tools while maintaining each plant's independence. Each fiefdom has all of the software and hardware required to function in isolation from the others. The segmentation is especially important during maintenance periods, because it enables portions of the control system to be unavailable in one fiefdom without impacting others. For example, if operating system patches must be installed, the modification can be made on one fiefdom at a time, and other fiefdoms can continue to operate, available for other maintenance work. During normal operation, all fiefdoms are accessible to all others. This makes it possible to centralize software development efforts in one fiefdom. When software is ready to be exercised operationally, it is easily distributed to the destination fiefdom. Support for the control system segmentation is built into the software development and operational tools used in the control system, making the segmentation transparent in normal operation.

4 DEVELOPMENT

A suite of tools is available to help software developers work with the control system. A primary goal of any operational control system is to enable developers to easily install software for testing and roll back these changes. At Jefferson Lab, where testing facilities are limited, the operational machine is usually the integration test bed for software enhancements. To minimize the impact of testing on availability, the lab uses a well-designed version management system for its low-level applications[2].

The purpose of the tools is to assist the developers in organizing and managing their applications while maintaining flexibility. A well-defined framework enables programmers to have a great deal of flexibility with the application implementation details. This is important, because the scope of various applications can be so different, ranging from an application that controls a single device with a few I/O channels, to one that drives an RF system with thousands of control points. A well-designed structure enables all to coexist, and aids in support efforts. The organizational similarities between applications provide a basis of understanding for all developers. This makes it easier for on-call staff to support a large body of applications. Also, because of the commonalities, a new developer can more quickly come up to speed.

The tools support versioning of applications, and the associated installation and rollback needed for support of the operational software. The computer scientist can create new application versions as needed, and designate which versions are operationally valid, using the test plan tools described above. The list of valid versions is stored in a database, accessible through web-based tools. Any software developer, can, with information provided by the database and using the standard tools, select a different software version.

Each step of the software development process is documented, with support of the available tools. This documentation is automatically available via the web, so that any control system user can find design and release notes for any version of the software. The easy accessibility helps users to understand the capabilities of the control system software.

CONCLUSIONS

A control system that is highly effective does more than simply provide device control and user interfaces for operating a complex, automated machine. A good control system enhances the productivity and quality of the work of all users, including operators, maintainers and developers. It provides mechanisms for supporting communication and coordination between people. This enables users to focus on the task at hand, rather than spending their energies in bookkeeping and other important but mundane tasks. The effective control system provides consistency in the appearance and behavior of tools for users, helping them to work more efficiently. Finally, it can also facilitate the work of enhancing and maintaining the accelerator by simplifying the work of technical support staff and operators.

REFERENCES

[1] K.S. White, M. H. Bickley, "Control System Segmentation", PAC 01, Chicago, USA, June 2001.

[2] S. Schaffner, M. Bickley, et. al., "Tools for Application Management at Jefferson Lab", ICALEPCS 99, Trieste, Italy, October 1999.

TUDT004
physics/0112026

A GUERILLA APPROACH TO CONTROL SYSTEM DEVELOPMENT

J. Dovc, G. Milcinski, M. Plesko, J. Stefan Institute, Ljubljana, Slovenia

Abstract

We present our experiences in managing the development cycles of the control systems for ANKA and the ALMA Common Software. Our team consists practically only of undergraduate students. Stimulating and rewarding the students with cutting-edge technologies and travel to conferences like this and installation fieldwork are an important positive factor in raising their motivation. However, building any system with a group of inexperienced students is quite a challenging task. Many problems occur with planning deadlines and missing them, organizing and managing development, sources, and documentation and also when dealing with conventional program management rules. To cope with them, we use many tools: CVS for versioning and source archiving, Bugzilla for keeping our bugs in order, a to-do list for managing tasks, an activity log and also many other programs and scripts, some found on the Internet and some made by ourselves. In the end, we had to become organized like a professional company. Documentation and demos can be found on our homepage: http://kgb.ijs.si/KGB. Because of powerful intranet/web front-ends of all those tools, our Internet pages are the central resource for developers, who work mostly off-site.

1 INTRODUCTION

The synchrotron light source ANKA has been built at the FZK (Forschungszentrum Karlsruhe), Germany. The control system has been outsourced through a commercial contract to the JSI (J. Stefan Institute) in Ljubljana, Slovenia. There, the KGB group (Kontrol Gruppe für Beschleuniger) has been founded in 1996 by one of the authors of this paper (M.P.) with the very naïve idea that a group of motivated, responsible and skilled students without previous experience in accelerator control systems can build such a system from a ten-pages long wish-list of features and ideas.

2 WHY GUERILLA?

It had been clear from the beginning of the ANKA project that the tight project schedule and low budget required an innovative approach both in technology and management. The well-organized groups at the FZK and JSI had experienced engineers who had enough other work to do and were not particularly motivated to take on another project, which did not have clear specifications at that time. We decided that we should bypass rigid organizational structures and use highly motivated people in order to save development time that we were short of.

That called for a guerilla approach (figure 1): Apart from finding people that are waiting to be motivated, sufficiently small tasks had to be defined that could be done in one guerilla action.

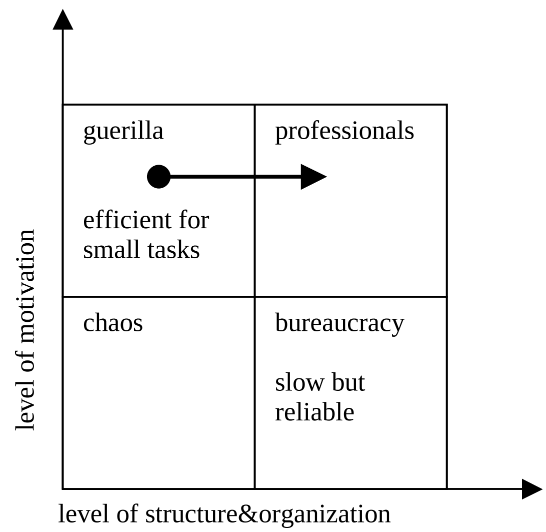


Figure 1: Different organizational forms based on team motivation and organization. We have started at the top left and are moving towards the top right.

2.1 The Guerilla Force

The only group of potential team members that might easily be motivated were students we knew from summer jobs. So we started to build our guerilla force, first with one student, then recruiting new members as we met more excellent students of physics, mathematics, electronics and computer sciences with affinity to programming. Whenever the opportunity arose to get a competent person, we took all efforts to ensure that (s)he would come to us. Initially it was not easy to find and to convince people to join us, but it got better as the group had results to show and became sufficiently large that members brought in their friends and classmates.

To make it more compelling for students, we decided to make a control system design based on the Internet and general network technologies, which were becoming very popular at that time like Java, CORBA and the LonWorks fieldbus.

We used any other way of motivating as this was the most important factor in raising the productivity: standard financial compensation, trips to conferences, the chance to work abroad, MP3 music with good speakers and the possibility to use broadband internet access from the institute. Of course, there are no algorithms to deal with people. Each individual is different and the project leader had to adapt his style and rewards to the person and not the other way around, in order to get the most out of that person.

Managing guerillas, who are practically volunteers, is quite challenging in that the leader has no formal leverage over the team. That's why in the end human resource management becomes more important than project management.

2.2 The Guerilla Tasks

As the students mostly could not work on a regular basis, sufficiently small tasks had to be made that had to have well-defined and long-term stable interfaces, both in hardware and software. The design was crucial, because a good design allowed to dissect the project in small manageable tasks. It was also important that the assembly of the pieces would not pose extra work. Clear interfaces helped us here.

This concept actually fit well together with the architecture of the control system of ANKA [1]. Because ANKA was a low budget project, it was necessary to offer not just a control system with a low purchase price, but also with:

- low maintenance cost
- low upgrade cost
- low failure rate

i.e., a system with low total cost of ownership (TCO).

In retrospect we can say that there were only two main reasons why we have successfully finished the project. One was horizontal division of tasks and responsibilities: each team member got a specific module to master and to implement, instead of being responsible for the control of a particular kind of equipment, which would correspond to vertical division. Many modules were generic and thus not related to specific equipment, or even to the control system and were used to hide complex details from casual programmers, e.g. the Abeans [2,3] Java-CORBA wrapper or the wizard that generates server code. Actually, most of the time the students didn't even know or care about the details of the controlled devices, whose functionality had been described with the CORBA interface definition language (IDL) according to the wide-interface approach: Each device is represented by a specific CORBA object which has its properties and methods reflecting the state and functionality of the device, (see [3] for more discussion on wide versus narrow interfaces). Since everybody had the guidelines set by all those wide interfaces, there was little that people needed to know to use each other's data as the most information comes from the data themselves. Using this approach, every property has to be well defined in advance and this required a lot of time during the design phase. But this time pays off when programming applications, writing documentation and maintaining the control system. Other benefits of this approach are described in [1].

The other reason for success was pure luck that we got some ingenious students right at the beginning of the project. They programmed the core modular components or developed and programmed the modular I/O boards, respectively, and helped to attract other good students.

2.3 Getting the Guerilla Organized

The whole approach probably would not have worked, if we had worked inside of ANKA, because the temptation for continuous improvements, postponing deadlines, not to mention ignoring writing documentation, would be too high. Being responsible to a customer, however, forced us to behave more professionally. We started to implemented several software engineering tools and made our team members use them consistently, which is described in detail in another paper [4]. We have done some improvements, e.g. CVS is now used consistently everywhere, we use documentation templates and UML for designs, but most of the tools are still heavily used.

Apart from inside pressure, outside factors affected our decisions to strengthen our organizational structure. In 1999-2000 we developed a core control system for a commercial company and now, started in 2000, a similar cooperation takes place with ESO (European Southern Observatory) for the ACS (ALMA Common Software).

There were subjects that we never even dared to approach, e.g. C/C++ programming rules. The C++ server was developed by one person, so he had his own rules but at least nobody had others – until we started the project for the commercial company with two new programmers, who couldn't even agree on variable naming, not to mention that the company had a third view. The situation with Java was better as general Java conventions are quite strong. Also, those who used Java learned it after joining the team from the same books and Suns examples and tutorials.

2.4 Project Management

Planning was done by the project leader. He also designed the whole control system and distributed the tasks. He used mainly to-do lists by using the features of MS Word in outline view, which we found very useful. The project manager was a charismatic student, whose task was to check the status of the work once a week. Although the productivity of developers increased before the deadline – the effect of last-minute panic, the project manager had to keep reminding them that the deadline was approaching. The most effective solution was to call them one by one, as it was nearly impossible to schedule a meeting with all students attending.

The schedule had to be adjusted to the dynamics of student life (exams, lectures, girl/boyfriends, etc). Most of the work was done during weekends, public holidays, and semester breaks. One of the consequences was that students mostly programmed at home. So we really had to have good means of communication. We used mobile phones (everybody had one), mailing lists for each subproject and weekly meetings.

Initially the hardest work for the project manager was to collect all the corresponding documentation. But by now everybody has learned that documentation is important and some write docs already in advance as specifications. We have close to 1000 pages of manuals for ANKA.

Testing on the other hand was quite successful from the onset. Because the majority of the software we had to test were GUIs, we could not easily use any automatic procedures to test our software. To assure the quality of the products, developers cross-tested the products of each other. That stimulated competition, which is another motivating factor for testing and finding bugs, which we manage using a Web bug reporting and bug tracking tool called Bugzilla.

The main resource of information was the KGB homepage both for the group and for our customers. The publicly available pages contain the complete documentation (manuals, specifications, design documents, white papers, etc.), FAQ, conference articles, presentations, and downloadable demos. The internal pages contain the address book of all the members, archives of mailing lists, and to-do lists. Other Web-based tools that we use heavily are:

- CVS for versioning of the programmers' code and all documentation (also binary formats such as MS Word) and as a repository from which backups are made
- Activity log: a set of simple Perl scripts, where the developers reported their activities
- Bugzilla (http://www.mozilla.org/bugs/)

Although used very efficiently, there is a negative aspect of having all tools on the Web: people who work at home are not online all the time. We are therefore looking for a tool that would cache all entries and synchronize then automatically when going online.

3 GUERILLAS TURN PROFESSIONALS

Even after having gotten used to all the project management and software engineering, the team members still are highly motivated. We are using formal document templates, design with UML, talk patterns, apply modular testing and have fun! We have even started JavaAcademy – a training course for newcomers to screen for the best candidates. It appears that we are reaching towards the upper right region in figure 1. Our guerilla approach pays dividends, because it is much easier to get a motivated but chaotic group organized then to motivate an organized team of dull individuals.

So we have a veteran team with an average age of 22. The oldest members have already graduated and left and we would lose all our investment if we let this trend to continue. Our institute cannot hire the whole team; therefore we have created a spin-off company for developing and installing control systems for accelerators and other large experimental facilities (www.cosylab.com). True to the research community that we grew in, the vision of the company is to make a living with our work instead of selling software licenses. And true to our philosophy of high motivation, all initial employees are co-owners.

Now the real management only starts. Will it turn from guerilla to professional? We will know when the company appears at a future ICALEPCS conference as sponsor.

4 ACKNOWLEDMENTS

We thank the Forschungszentrum Karlsruhe for placing the initial contract to us, giving us thus the chance to create this excellent team. We also thank the ESO, in particular Gianluca Chiozzi, for teaching us programming and project discipline. We found that it isn't so difficult after all, although Gianluca might argue that we are still far from it.

REFERENCES

[1] J. Dovc et al., ANKA Control System Takes Control, PCaPAC 2000, Hamburg, October 2000

[2] I. Kriznar et al., The Object Oriented Approach to Control and Machine Physics Calculations with Java Technology, ICALEPCS 01, SLAC, Dec 2001

[3] J. Dovc et al., New Abeans for TINE Java Control Applications, ICALEPCS 01, SLAC, Dec 2001

[4] J. Dovc et al., Managing the Development of a New Control System, PCaPAC 2000, Hamburg, October 2000

TUESDAY, 27 NOVEMBER 2001

TUAP - POSTERS

TUAP001
cs.SE/0111045

THE OVERVIEW OF THE NATIONAL IGNITION FACILITY DISTRIBUTED COMPUTER CONTROL SYSTEM

L. J. Lagin, R. C. Bettenhausen, R. A. Carey, C. M. Estes, J. M. Fisher, J. E. Krammen, R. K. Reed, P. J. VanArsdall, J. P. Woodruff, LLNL, Livermore, CA 94551, USA

Abstract

The Integrated Computer Control System (ICCS) for the National Ignition Facility (NIF) is a layered architecture of 300 front-end processors (FEP) coordinated by supervisor subsystems including automatic beam alignment and wavefront control, laser and target diagnostics, pulse power, and shot control timed to 30 ps. FEP computers incorporate either VxWorks on PowerPC or Solaris on UltraSPARC processors that interface to over 45,000 control points attached to VME-bus or PCI-bus crates respectively. Typical devices are stepping motors, transient digitizers, calorimeters, and photodiodes. The front-end layer is divided into another segment comprised of an additional 14,000 control points for industrial controls including vacuum, argon, synthetic air, and safety interlocks implemented with Allen-Bradley programmable logic controllers (PLCs). The computer network is augmented asynchronous transfer mode (ATM) that delivers video streams from 500 sensor cameras monitoring the 192 laser beams to operator workstations. Software is based on an object-oriented framework using CORBA distribution that incorporates services for archiving, machine configuration, graphical user interface, monitoring, event logging, scripting, alert management, and access control. Software coding using a mixed language environment of Ada95 and Java is one-third complete at over 300 thousand source lines. Control system installation is currently under way for the first 8 beams, with project completion scheduled for 2008.

1 INTRODUCTION

This paper presents an overview of the NIF ICCS. The NIF contains 192 laser beam lines that are focused on an inertial confinement fusion (ICF) capsule at target chamber center [1]. Each beam requires alignment, diagnostics, and control of power conditioning and electro-optic subsystems. NIF will be capable of firing target shots every 8 hours, allowing time for the components to cool sufficiently to permit precise realignment of the laser beams onto the target.

The NIF requires integration of about 60,000 atypical control points, must be highly automated and robust, and will operate around the clock. Furthermore, facilities such as the NIF represent major capital investments that will be operated, maintained, and upgraded for decades. Therefore, the computers and control subsystems must be relatively easy to extend or replace periodically with newer technology.

The ICCS architecture was devised to address the general problem of providing distributed control for large scientific facilities that do not require real-time capability within the supervisory software. The ICCS architecture uses the client–server software model with event-driven communications. Some real-time control is also necessary; controls requiring deterministic response are implemented at the edges of the architecture in front-end computer equipment. The software architecture is sufficiently abstract to accommodate diverse hardware, and it allows the construction of all the applications from an object-oriented software framework that will be extensible and maintainable throughout the project life cycle. This framework offers interoperability among different computers and operating systems by leveraging a common object request broker architecture (CORBA). The ICCS software framework is the key to managing system complexity.

A brief summary of performance and functional requirements follows in Table 1.

Table 1: Selected ICCS performance requirements

Computer restart	< 30 minutes
Post-shot data recovery	< 5 minutes
Respond to broad-view status updates	< 10 seconds
Respond to alerts	< 1 second
Perform automatic alignment	< 1 hour
Transfer and display digital motion video	10 frames per second
Human-in-the-loop controls response	within 100 ms

Summary ICCS functional requirements:

- Provide graphical operator controls and equipment status.
- Maintain records of system performance and operational history.
- Automate predetermined control sequences (e.g., alignment).

- Coordinate shot setup, countdown, and shot data archiving.
- Incorporate safety and equipment protection interlocks.

2 ARCHITECTURE

The ICCS is a layered architecture consisting of FEPs coordinated by a supervisory system (Figure 1). Supervisory controls, which are hosted on UNIX workstations, provide centralized operator controls and status, data archiving, and integration services. FEP computers incorporate either VxWorks on PowerPC or Solaris on UltraSPARC processors that interface to over 45,000 control points attached to VME-bus or PCI-bus crates respectively. Typical devices are stepping motors, transient digitizers, calorimeters, and photodiodes. FEP software provides the distributed services needed to operate the control points by the supervisory system. Functions requiring real-time implementation are allocated to software within the FEP (or embedded controllers) that does not require time-critical communication over the local area network. Precise triggering of 1600 channels of fast diagnostics and controls is handled during a 2-second shot interval by the timing system, which is capable of providing triggers to 30-ps accuracy and stability [2]. The software is distributed among the computers and provides plug-in software extensibility for attaching control points and other software services by using CORBA.

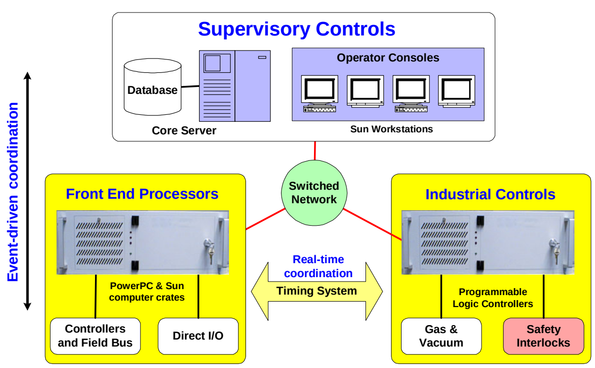


Figure 1: Control system architecture.

Operator consoles provide the human interface in the form of operator displays, data retrieval and processing, and coordination of control functions. Supervisory software is partitioned into several cohesive subsystems, each of which controls a primary NIF subsystem such as alignment or power conditioning. Several databases and file servers are incorporated to manage both experimental data and data used during operations and maintenance.

FEPs implement the distributed portion of the ICCS by interfacing to the NIF control points. The FEP software performs sequencing, data acquisition and reduction, instrumentation control, and input/output operations. The software framework includes a standard way for FEPs to be integrated into the supervisory system by providing the common distribution mechanism coupled with software patterns for hardware configuration, status, and control.

The front-end layer is also divided into another segment comprised of an additional 14,000 control points for industrial controls including vacuum, argon, synthetic air, and safety interlocks implemented with Allen-Bradley PLCs. The segment consists of a network of PLCs that reside below the FEP layer and attach to field devices controlling vacuum systems for the target chamber and spatial filters, argon gas controls for the beam tubes, and thermal gas conditioning for amplifier cooling. This segment also monitors the independent safety interlock system, which monitors doors, hatches, shutters, and other sensors in the facility.

ICCS is further divided into subsystems to partition activity and ensure performance. There are ten supervisor software applications that conduct NIF shots in collaboration with 17 kinds of FEPs. Software applications in the NIF control system subsystems partition the control system into loosely coupled vertical slices consisting of a supervisor and associated FEPs that are easier to design, construct, operate, and maintain.

- All supervisors are controlled by a Shot Director, which is responsible for conducting the shot plan, distributing the countdown clock, and coordinating the other subsystems.
- The beam control supervisor provides coordination and supervision of laser wavefront control and laser component manual and automatic alignment and optics inspection [3].
- The laser diagnostics supervisor provides functions for diagnosing laser performance by collecting integrated, transient and image information from sensors positioned in the beams [4].
- The optical pulse generation supervisor provides temporally and spatially formatted optical pulses with the correct energetics and optical characteristics required for each of the beams.
- The target diagnostics supervisor coordinates the collection of data from a diverse and changing set of instruments.
- The power conditioning supervisor manages high-voltage power supplies that fire the main laser amplifiers.

- The Pockels cell supervisor manages operation of the plasma electrode Pockels cell optical switch that facilitates multipass amplification within the main laser amplifiers.
- The shot services supervisor provides monitoring of industrial controls and integrated timing systems.
- A final supervisor interfaces to a computerized Laser Performance Operations Model (LPOM) simulation, which is being developed to guide laser setup of laser operating parameters [5].

3 COMPUTER SYSTEM AND NETWORK

A computer network interconnects approximately 750 systems, including embedded controllers, 300 FEPs, supervisory workstation systems, and centralized servers [6]. All systems are networked with Ethernet to serve the majority of communication needs, and ATM is utilized to transport multicast video and synchronization triggers. CORBA software infra-structure provides location-independent communication services over TCP/IP between the application processes in the workstation supervisors, servers, and FEPs. Video images sampled at a 10-Hz frame rate from any of 500 video cameras will be multicast using direct ATM data streams from video FEPs to any operator console.

4 ICCS SOFTWARE FRAMEWORK

The ICCS is based on a scalable software framework that is distributed over supervisory and FEP computers throughout the NIF facility [7]. The framework provides templates and services at multiple levels of abstraction for the construction of software applications that distribute via CORBA. Framework services such as alerts, events, message logging, reservations, user interface consistency, and status propagation are implemented as templates that are extended for each by application software. Application software is constructed on a set of framework components that assure uniform behavior spanning the FEP and supervisor programs [8]. Additional framework services are provided by centralized server programs that implement database archiving, name services, and process management.

5 SOFTWARE DEVELOPMENT AND TESTING

The ICCS incorporates a mixed language environment of Ada95 and Java using CORBA and object-oriented techniques to enhance the openness of the architecture and portability of the software. The object-oriented design is captured in Unified Modeling Language using the Rational Rose design tool that maintains schematic drawings of the software architecture, while source code is managed by the Rational Apex tool.

The strategy used to develop the NIF ICCS calls for incremental cycles of construction and formal test to deliver an estimated total of 1 million lines of code [9]. Each incremental release allocates four to six months to implement targeted functionality and culminates when offline tests are conducted in the ICCS Integration and Test Facility. Tests are then repeated online to confirm integrated operation in dedicated laser laboratories or ultimately in the NIF. Process measurements including earned-value, product size, and defect densities provide software project controls and process improvements that generate confidence that the control system will be successfully deployed.

6 SUMMARY

Construction of the NIF ICCS incorporates many of the latest advances in distributed computer and object-oriented software technology. Primary goals of the design are to provide an open, extensible, and reliable architecture that can be maintained and upgraded for decades. The control system is being developed using the incremental approach to software construction. Eighteen (of over 40 planned) releases of software have been deployed and tested since 1998, resulting in over 325 thousand lines of source code that is capable of supporting initial laser operations. Control system installation is under way for the first 8 beams, with project completion scheduled for 2008.

This work performed under the auspices of the U.S. DOE by LLNL under contract No. W-7405-Eng-48.

REFERENCES

[1] E. Moses et al., “The National Ignition Facility: Status and Plans for Laser Fusion and High-Energy-Density Experimental Studies,” ICALEPCS 2001, San Jose, CA, November 27-30, 2001.
[2] R.A. Lerche et al., “The NIF Integrated Timing System - Design and Performance,” ICALEPCS 2001.
[3] Lewis Van Atta et al., “The Wavefront Control System for the NIF,” ICALEPCS 2001.
[4] R.D. Demaret et al., “Beam Diagnostics Systems for the NIF,” ICALEPCS 2001.
[5] Chris Haynam et al., “Computational Modeling in Support of NIF Operations,” ICALEPCS 2001.
[6] R. M Bryant et al., “NIF Control Network Design and Analysis,” ICALEPCS 2001.
[7] Robert W. Carey et al., “Large-scale CORBA-distributed software framework for NIF controls,” ICALEPCS 2001.

[8] Kirby W. Fong et al., “Application Software Structures Enable NIF Operations,” ICALEPCS 2001.

[9] John P. Woodruff et al., “Quality Control, Testing and Deployment Results in NIF ICCS,” ICALEPCS 2001.

TUAP002
physics/0111202

A CONTROL SYSTEM OF THE JOINT-PROJECT ACCELERATOR COMPLEX

J. Chiba, H. Fujii, K. Furukawa, N. Kamikubota, H. Nakagawa, N. Yamamoto
High Energy Accelerator Research Organization (KEK), Tsukuba, Japan 305-0801
H. Sakaki, H. Yoshikawa
Japan Atomic Energy Research Institute (JAERI), Tokai, Japan 319-1195

Abstract

The current status of the control system for a new high intensity proton accelerator, the JAERI-KEK Joint Project, is presented. Phase 1 of the Joint Project has been approved and recently started its construction at the JAERI site at Tokai. The first beam commissioning is scheduled in 2006. In parallel with it, a 60-MeV Linac is now being constructed at the KEK site at Tsukuba for R&D purposes.

Recently the Project has officially decided to use the Experimental Physics and Industrial Control System (EPICS). Under the EPICS environment, we are challenged with implementing the Ethernet/IP network for all communication, even at the level of end-point controllers which are currently connected via a field bus. In order to realize such a system, three new controllers (PLCs, WE7000 stations and general-purpose Ethernet boards) are being developed. A prototype EPICS driver for the PLCs works fine and is used to control the ion-source at the KEK Linac.

1 THE JOINT-PROJECT

As reported in the previous conference [1] the two projects, JHF at KEK and NSP at JAERI, were merged and formed a new project, the JAERI-KEK Joint Project. Phase 1 of the project consists of three accelerator facilities (400-MeV Linac, 3-GeV Rapid-cycling Synchrotron, and 50-GeV Synchrotron) and two experimental facilities (Material and Life Science Facility at 3 GeV and Nuclear and Particle Physics Facility at 50 GeV). The construction of Phase 1 will be completed in 2006. The schedule of Phase 2 is not known yet, in which a 600-MeV Super Conducting Linac and Accelerator Driven Transmutation (ADS) Experimental Facility will be constructed.

We are planning to make a central control system with which not only the accelerator but also the beams and targets in the experimental facilities can be monitored and operated in order to reduce the operational costs including labor.

In addition to the facilities at Tokai mentioned above, we are constructing a 60-Mev Linac at the KEK site. It is comprised of an ion source, a low-energy beam transport (LEBT), 3-MeV RFQ Linac, a medium-energy beam transport (MEBT), a 50-MeV drift-tube Linac (DTL), and a 60-MeV separated DTL (SDTL). The DTL and SDTL will be transported and re-installed at Tokai after successful beam tests.

2 DESIGN OF THE CONTROL SYSTEM

2.1 Introduction

Before forming the Joint Project, the use of the EPICS toolkit at the JHF accelerator had been decided after intensive discussion among the JHF control group members [1]. Although there were some complaints against the use of EPICS, we recently decided to use it in the Joint Project. Since the Joint JAERI-KEK construction team have been formed only a half year ago, concrete rules needed for the control system design are not determined yet except for the use of EPICS.

In the following subsections, we describe our present thought on the control system in 5 components; (1) device control, (2) network, (3) operator interface, (4) interface to beam simulator and (5) database.

2.2 Device control

In EPICS, all device controls are carried out through input/output controllers (IOCs). At present, only VMEs under the vxWorks operating system can be used as an EPICS IOC. Many VME modules are commercially available for processing normal digital and analogue signals. EPICS drivers are already available for many of those modules. Therefore, we don’t need to worry about devices which require only those signals. However, there are many devices which need special treatments.

PLCs are cheap and reliable in a sense that they keep running under any trouble in computers or networking, and therefore it is preferable to use PLCs for some critical devices such as an ion-source. How PLCs communicate with IOCs is one of the issues we have to determine early. Our answers are as follows:

- Use Ethernet TCP/IP. Among PLCs commercially available, presently only FA-M3 PLC by Yokogawa support data transfers protocols we required, we force equipment groups to use it if they need PLCs.
- Assign an identification number (ID) to each PLC. It is dangerous to rely only on the IP address. Before any communication with a PLC and at any data (command) transfer to a PLC, check the ID to make sure you are communicating with an appropriate PLC.
- No individual commands for operation. Only data transfer from/to PLC memory is supported. Any device operation should be done though contents of memory in a restricted address range.
- No direct write to PLC memory. An IOC sends 3-word data (an ID, a memory address and content) to a communication area in the PLC memory. Then, a PLC ladder program moves the content to an appropriate address after checking its ID.

A prototype of the PLC EPICS driver has been written and successfully used for the ion-source operation at the KEK 60-Mev Linac, although all the rules mentioned above are not fully implemented yet in the prototype driver. A detailed description on the device driver may be found elsewhere [2].

Choice of waveform digitizers is another important issue we have to consider for device control. As reported in the previous conference [3], our choice was the WE7000 measurement station by Yokogawa. Its EPICS driver has been partially completed and tested. For this station, the same TCP/IP Ethernet communication method is used as that for the PLCs. A detailed description is given again in Ref. [2].

For precise current control of power supplies, we should avoid extending an analogue signal in a long distance, and instead, should use digital data links. Since Ethernet network runs all over the area, we chose Ethernet for the digital data link, avoiding another type of field bus so that all of the data communication is unified. By doing so, we can trace back all data through the network which will significantly help us for operation diagnostics. Also it reduces the number of stock items which may reduce the operational cost. To achieve the unified Ethernet links, we are developing an Ethernet interface board which will be used primarily in the power supplies for quadrupole magnets at the DTL. The board is designed considering more general uses and may be embedded in other devices.

However, some devices required for some equipment may communicate with our control system only via a method other than Ethernet. For example, most of low-price power supplies have a GPIB or RS232c interface for remote access. In such cases, we will use Ethernet interface such as a GPIB LAN gateway for GPIB and a terminal server for RS232c to minimize the area covered by field buses.

2.3 Network

Redundant gigabit Ethernet with fiber optics linked as a "star" starting from the central control room (CCR) to each accelerator (Linac, 3-GeV RCS and 50-GeV PS) and experimental facility is the backbone of the control network. From each backbone station, star-like 100baseFX optical fiber cables extend to network switches over the facility. Most of the network end nodes such as IOCs are 100baseTX. Some nodes in a high EMI noise levels such as near klystrons or high-powered pulsed power supplies should be 100baseFX.

As described in previous subsection, we are going to use Ethernet links instead of other field buses. Therefore the total number of IP addresses required for the control system could easily exceed a few thousands. Therefore it is impossible to use a global IP address space. Instead we are going to use a Class B private IP address space and use a router to communicate outside the network.

We have not figured out how the network should be divided into subnets. The EPICS channel access (CA) protocol requires the network broadcast. Therefore all EPICS nodes should be in one subnet. Nodes for diagnostic purpose should belong to a different subnet because they must be free from the EPICS subnet. Such diagnostic nodes include a communication port of a traffic watcher (of course, it should also have a port in the EPICS subnet to spy on network traffic) and an IOC’s CPU console (RS232c through a terminal server). Those two types of nodes are in general close each other in space, and therefore it is better to implement the virtual-LAN technique to reduce cabling and number of network switches.

2.4 Operator interface

We will use one of the standard EPICS OPI tools for the EPICS operation, but it is not specified yet which one should be used. Also there are two choices; (1) OPI applications run directly on a PC Linux or (2) OPI applications run on an X-client machine with an X-terminal (or a PC with X-server). The former choice is good for network traffic but not good for maintaining applications. The latter is just the opposite. Another type of operator interface is necessary which is related to the issues described in the next subsection.

2.5 Interface to accelerator simulator

In order to operate the accelerator stably, it is important to simulate the beam before parameters are physically changed. Accelerator physicists are responsible for developing such a simulation code, but the control group should help them to include the EPICS interface and operator interface in the code. SAD code [4] used at the KEKB accelerator is a candidate in the Joint-Project. Since SAD code used at KEKB already has the graphical user interface (GUI) and the EPICS interface, we don't need to worry so much about the interfaces if SAD is chosen as a simulation code. However, SAD must be significantly modified or new simulation code should be implemented for the Linac simulation. If new code is chosen, we have to provide the GUI and EPICS interface similarly to the SAD case.

2.6 Database

For steady and reliable operations of the accelerator, use of proper database system is indispensable. We have just started to design various kinds of databases necessary in our system.

3 PRESENT STATUS

At KEK, an EPICS application server (HP-UX, HP 9000 model D380/2) and several VME IOCs (Force PowerCore 6750) have been installed primarily for software development.

Development of an EPICS device driver for the WE7000 measurement station (Yokogawa) is being carried out. So far we have developed EPICS device support for 3 WE7000 modules; WE7111 (100MS/s digital oscilloscope), WE7121 (10MHz function generator) and WE7271 (4ch 100kS/s waveform digitizer). It basically works fine. However, concurrent use with MS Windows applications doesn't work for unknown reasons. Because the WE7000 has been developed for MS Windows OS, there are fruitful applications available under Windows system. Those applications are very useful for the maintenance of the WE7000 stations and developments of EPICS applications. We must solve this problem soon because the concurrent use with Windows is indispensable.

The present version of the EPICS driver for Yokogawa's FA-M3 PLC supports only simple input/output operations. The rules for the PLC mentioned in Sec. 2 are not implemented yet.

The EPICS system will also be used for the operation of the 60-MeV Linac. At present, only the ion-source control which uses a MA-F3 PLC is under the EPICS environment.

The HP EPICS server, VME IOCs, PLCs and 3 operators PC terminals form a Class C network with private IP address. Operators can access the KEK-LAN via routers (2 of freeBSD PCs). The HP server also has a network port to KEK-LAN through which we access to the system from outside of the accelerator building.

At JAERI, an HP EPICS application server and several VME IOCs are installed. A test stand for klystron power supplies employs the EPICS system. An EPICS driver for a VME pulse motor controller module is now under development.

4 ACKNOWLEDGEMENTS

The authors would like to thank all members of the Joint-Project team. They also acknowledge members of the KEKB ring control group for their advices and suggestions.

REFERENCES

[1] J. Chiba, K. Furukawa, E. Kadokura, N. Kamikubota, H. Nakagawa, K. Nigorikawa and Y. Yamamoto, "A control system of the JHF accelerator complex", ICALEPICS'99, Trieste, Italy, 1999, p672.

[2] K. Furukawa, J. Chiba, N. Kamikubota, N. Nakagawa, "Implementation of the EPICS software support for network based controllers", in these proceedings.

[3] K. Furukawa, J. Chiba, N. Kamikubota, N Nakagawa, "Network based EPICS drivers for PLCs and measurement stations", ICALEPICS'99, Trieste, Italy, 1999, p409

[4] http://acc_physics.kek.jp/SAD/sad.html (KEKB SAD homepage)

TUAP003
hep-ex/0111076

PRESENT STATUS OF VEPP-5 CONTROL SYSTEM

D.Yu.Bolkhovityanov, R.G.Gromov, E.A.Gousev, K.V.Gubin, I.L.Pivovarov, O.Yu.Tokarev
BINP, Novosibirsk, Russia

Abstract

This report concerns the present status of VEPP-5 control system. The control system hardware consists of CAMAC blocks, a set of crate controllers based on INMOS transputers and ICL-1900 architecture processor Odrenok, and Pentium-based workstations. For small tasks simple serial CAMAC controllers are used. For slow controls of power supplies the CANBUS is begun being used. The workstations are running Linux and are connected via local net using TCP/IP. Odrenok crate controllers are joined into other local net and are used for control of equipment in high voltage pulse condition (klystron gallery). Transputer crate controllers are linked directly to the server computer and are used for high performance diagnostics (BPM). The three-level software complies the so-called "standard model".

1 INTRODUCTION

1.1 General design

The VEPP-5 forinjector is a large installation that includes: DC-gun, klystron gallery, power supply system, bunch compression system, thermo system, BPM and others [1]. Operation is pulsed with repetition rate from 1 to 50 Hz. The control system of the VEPP-5 injection complex has a standard three-level model[2] (fig. 1).

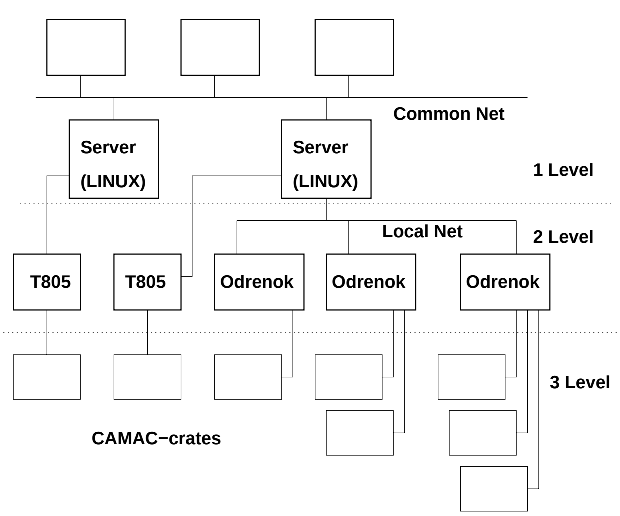


Figure 1: Control system network.

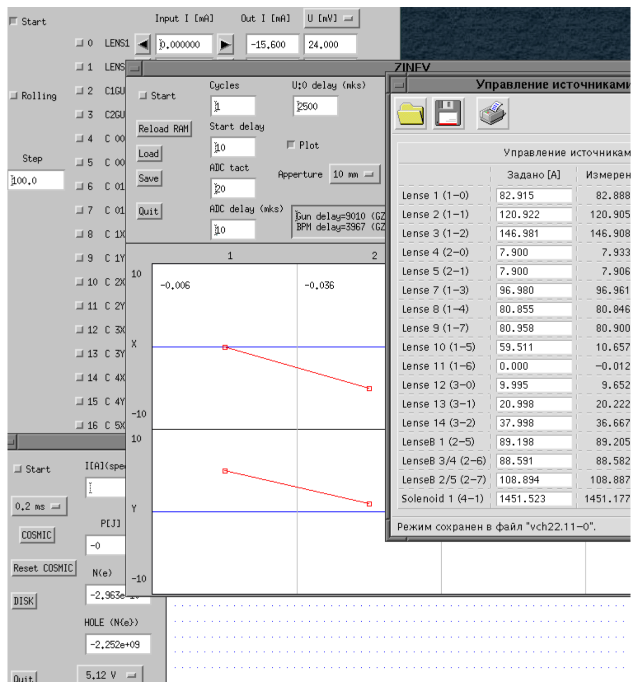


Figure 2: Several generations of high-level control software.

The lowest level is composed mainly from CAMAC electronics. The second level is a set of CAMAC crate controllers with supplementary low-level software. There are several types of intelligent crate controllers based on INMOS T805 transputer, ICL-1900 Odrenok and Motorola 5200.

The high level consists of a server which runs under Linux (RedHat-7.1). All client programs have access to the low level only via the server, which performs initial loading and initialization of the crate controllers and supplementary programs.

1.2 Software design

Historically there have been several generations of control system software on VEPP-5. The very first programs were simple – they implemented both client interface and hardware access. This approach is still used in some tasks[1].

However, we quickly switched to three-level architecture, because it is much more suitable for control tasks. First ver-

[1] A Russian proverb says: there's nothing more constant than temporary.

sions were bound to specific controllers – separate software for Odrenok and T805. They are still in use, but a general version was developed which is able to serve various types of controllers simultaneously (see Fig.3). The design of the general version was greatly influenced by design principles of X11.

The new version uses the same mechanisms (which are dictated by the nature of the controllers) as previous versions, described in sections 2 and 3.

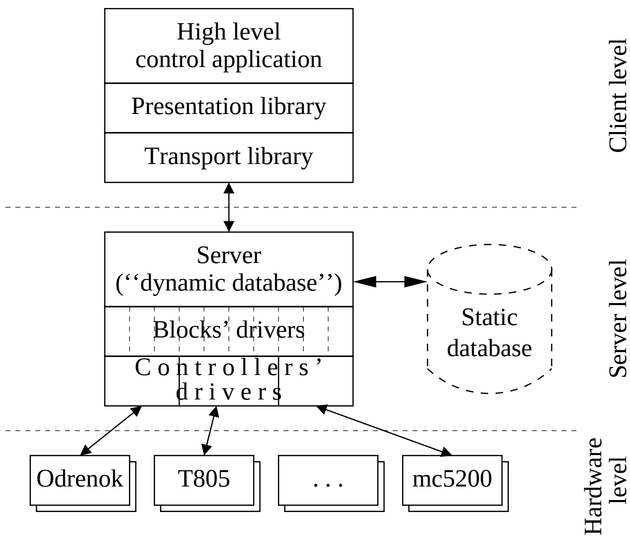


Figure 3: Structure of the current clients ↔ server ↔ hardware system.

High-level programs use Motif for the client interface. A special library was designed, which builds control windows “on the fly” from a database description, thus significantly reducing application creation time & costs. (An example of such a window can be found at the right of Fig.2.)

2 ODRENOK ↔ SERVER COMMUNICATION

Odrenok is the most popular crate controller in BINP. It has the ICL-1900 instruction set supplemented with commands for CAMAC bus access and vector operations. The new CAMAC adapter was developed to have a possibility to connect CAMAC with PC via Ethernet. The CAMAC-Ethernet adapter provides a speed of data exchange up to 400kB/sec [3]. It is sufficient for real-time network operation. The server workstation has two Ethernet cards: one is for communication to the institute network and another is for the local Odrenok net.

The ODOS (ODrenok Operation System) protocols use Ethernet packets of non-standard type, therefore the server requires I/O facilities for these packets. It is implemented by a kernel module which provides sending, receiving and waiting functions via a standard socket interface; the module is supplemented by client libraries and utilities (Fig. 4).

However, usage of home-made protocol have shown many inconveniences. So, this year UDP support was added

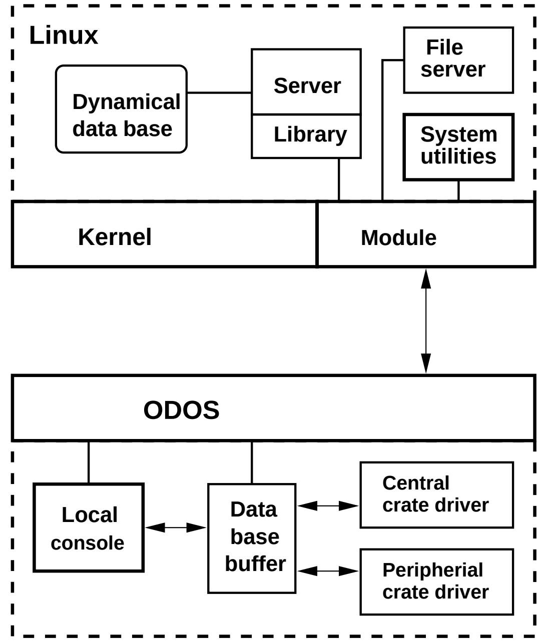


Figure 4: Linux ↔ ODOS Software structure.

to ODOS, which enables to use standard Unix communication interface and solves many other problems (routing between networks, kernel upgrades, stability).

3 TRANSPUTER ↔ SERVER COMMUNICATION

Crate controllers based on INMOS T805 Transputer [4] have been developed in BINP as high performance devices for data acquisition systems. This type of controllers is a suitable tool for wired BPM and pick-up electronic control. This is because T805 has a powerful floating point unit. In this case all calculations are performed on the 2nd level and the calculated data are sent into the 1st level software.

Transputer controllers have no OS (despite the fact that transputers are excellent for parallel tasks), so the following set programs had to be made (See Fig. 5). First, a dispatcher performs communication between the local transputer net and a set of block drivers. Second, an event handler (“PINT”) which enables drivers to read data when it is ready. Finally, a set of drivers for individual CAMAC blocks (1 driver process per block).

4 MOTOROLA CRATE CONTROLLER

The Odrenok crate controllers are very old. Production of T805-based controllers had ceased. So, we have to find another crate controller.

This year we began using another one, based on Motorola 5200 processor. 100MB Ethernet is used for communication to host computers.

This controller runs uClinux [5] – a Linux clone designed for processors without a Memory Management Unit

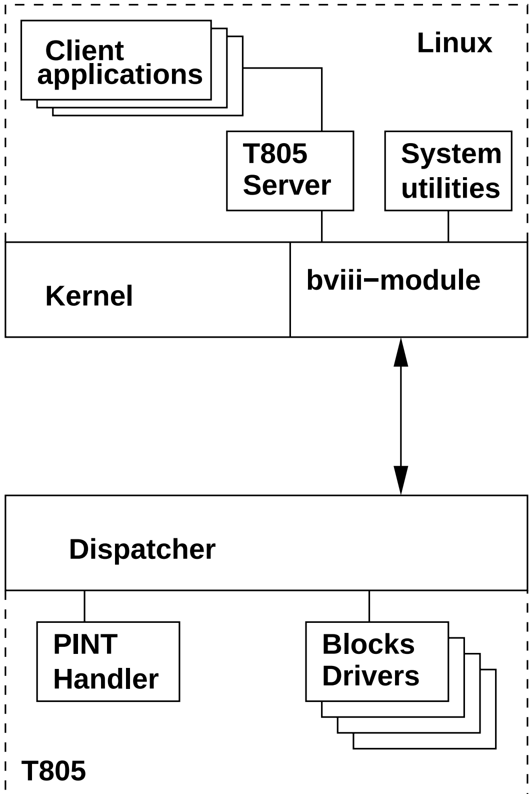


Figure 5: Linux ↔ T805 Software structure.

(MMU). Having Linux in both workstations and crate controllers significantly eases the life. On the other hand, lack of MMU makes multitasking and multithreading too tricky.

So, using ARM, PowerPC or an x86 clone in CAMAC controller could be much better choice, but 5200 was chosen mainly because of abilities of BINP electronic design department.

5 FUTURE DEVELOPMENT

Currently the information about hardware structure and knowledge of how it is mapped to "physics" information is spread along the various parts of software – in the server config files, hardcoded into application programs, etc. This is extremely inconvenient and error-prone, so we switched a significant part of manpower to design a database, which will contain all this information (a so-called "static database"). The database is based on PostgreSQL, but for more flexibility all pieces of the control software will access it through the server.

In the last several years VEPP-5 began to use other hardware in addition to CAMAC. In this process we tried to employ the most standard interface as possible. The ultimate goal is to replace all custom-made PC↔hardware communication boards with standard ones, such as Ethernet.

For slow controls of power supplies CANBUS devices are used. However, this decision is still half-CAMAC-based – the CANBUS controller is itself a CAMAC block. This was done because it was impossible to find PCI CANBUS controllers with open specifications. The CANBUS hardware fits nicely into our software architecture, so we'll widen its use.

Some devices (like TV cameras) require very high bandwidth, which can't be obtained from CAMAC. So, our lab designs TV camera with 100MB Ethernet interface. Having negative experience with non-standard communication protocols, we decided to implement transport protocol over UDP.

Since Ethernet chipsets are very cheap, even not-so-demanding hardware is being implemented with Ethernet as communication media. Thus Ethernet extends its presence as one of control system's low-level buses. But various types of hardware (crate controllers, TV cameras, slow devices) will be physically connected to separate buses.

Currently the VEPP-5 control system is being designed and tested on the forinjector, and in the future it will also be used on damping ring.

6 REFERENCES

[1] A.V.Aleksandrov et al., "Test of electron Linac for VEPP-5 Preinjector", Proc. EPAC'2000, Vienna, 2000, http://accelconf.web.cern.ch/accelconf/e00/PAPER/MOP4A05.pdf

[2] Götz et al., "Experience with a standard model'91 based control system at the ESRF", Proc. ICALEPS'93 (Berlin), NIM 352A (1994) p. 22-27

[3] A.N.Aleshaev et al., "VEPP-5 clystron gallery control system", Proc. PAC'99, New York, 1999, http://bnlinfo2.bnl.gov/cgi-bin/pac99/displaypaper.cgi?MOP41

[4] INMOS Limited, "Transputer Instruction Set", Prentice Hall, 1988, ISBN 0-13-929100-8.

[5] Lineo, Inc., "Embedded Linux/Microcontroller Project", http://www.uclinux.org/

TUAP004
cs.DC/0111028

THE ESRF TANGO CONTROL SYSTEM STATUS

J-M Chaize, A. Götz, W-D. Klotz, J. Meyer, M. Perez, E. Taurel and P. Verdier
ESRF, BP220, Grenoble, 38043, FRANCE

Abstract

TAco Next Generation Object (TANGO) is an object oriented control system toolkit based on Common Object Request Broker Architecture (CORBA) presently under development at the ESRF. In this paper, the TANGO philosophy is briefly presented. All the existing tools developed around TANGO will also be presented. This includes a code generator, a Web interface to TANGO objects, an administration tool and an interface to LabView. Finally, an example of a TANGO device server for OPC device is given.

1 INTRODUCTION

The task of building a control system in today's world has been heavily influenced by the ever-increasing choice of commodity off the shelf products. Many of the control problems (hardware and software) have been solved and can be bought ready to use off-the-shelf. However the products have to be integrated in order to form a control system. System integration is therefore one of the main tasks of a control system builder today. TANGO has been developed with system integration as one of its main design goals.

In TANGO system integration is achieved by wrapping. Wrapping means inserting a layer of software between the product to be integrated and the system in which it has to be integrated. The wrapper layers runs on the product platform and communicates with the control system via the network. The wrapper software needs to be multi-platform, network based and language independent. TANGO has chosen CORBA as its wrapper software.

2 THE CORBA MIDDLEWARE

CORBA is what the software industry calls a *middleware*. This means that it is a layer that allows end-user applications to communicate with other end-user applications or with utilities while hiding all the communication protocols. The CORBA definition uses the object approach to deal with the communication problem. Object interfaces are defined using a language called Interface Definition Language (IDL). An object interface defines all the kinds of requests that the object supports coming from the external world. CORBA defines language mappings from IDL to the main programming languages. Various commercial and non-commercial implementations exist for CORBA for all the mainstream operating systems.

3 TANGO PHILOSOPHY

The TANGO philosophy could be summarized in the following points:

- Hide CORBA details from the end-user application and from the device access software programmer. This is achieved by providing programmers with a device pattern for implementing new control classes and an Application Programmer Interface (API) for implementing physics applications.
- Define only one type of network object. This means only a single IDL file and only a single type of object to support as far as the communication layer is concerned. All controlled objects will inherit from this base class. This ensures all objects support the same basic interface and functionality. This uniqueness of object control interface allows writing of generic applications which can be used whatever the controlled device is.
- Group controlled objects in processes called *device server*s. All device server processes have the same architecture based on a well-defined device pattern.
- Keep a high degree of flexibility by using a database for device specific information and description.
- Use only freely available software.

4 THE TANGO OBJECT INTERFACE (IDL)

Only one IDL file is defined as there is only one interface to support. Actions are performed on devices by executing commands. Each command is defined by its name and has one input and one output parameter (which could be void). These parameters must be one of a list of 20 data types supported by TANGO. For commands without an input parameter, results are returned from the hardware itself or from a data cache continuously filled by a polling mechanism. Commands are sent to the device via a *command_inout* operation defined in the IDL. The use of CORBA Any objects allows the same device operation to transmit different data types. Commands can be executed synchronously or asynchronously. Asynchronous commands have to supply a callback object to receive the answer. Devices also support a list of attributes which can be read or written. Like commands, attribute values can be read from the hardware or from the data cache. Device also supports describing calls like *command_list_query* or *get_attribute_config* which enable generic applications to deal with any kind of device.

5 DEVICE SERVER PROCESS

A device server is an operating system process with one or several user implementations of the TANGO device pattern. Following a predefined *main()* or *winmain()* structure, the device access software programmer merges all the device pattern implementations s/he wants to run within the same process. Device name and number for each device pattern implementation are defined in the database and are retrieved during the process startup sequence. Multiple instances of the same device server process may run within a single TANGO system. Each instance has an instance name specified when the process is started. The process executable name/instance name pair uniquely defines a device server.

6 TANGO STATUS

Since the last ICALEPCS meeting in Italy, the following features have been added to TANGO:

- Full support for Windows. It is now possible to run a TANGO device server in an MS-DOS window, as a stand-alone application with its own windows using MFC or the Win32 library, or as a service
- Full support for TANGO attributes
- Polling threads in device server processes that enable data caching for slow devices
- The Java and C++ API's
- A device server code generator called pogo
- A TANGO administration tool called Astor
- A Web generic application to ease device testing called Jive
- A LabView interface

These new features are described in the following chapters.

7 DATA TYPES AND API

TANGO supports a fixed set of data types for transferring data using commands. All simple types and sequences of simple types are supported. In addition, TANGO supports two mixed types (sequence of strings and longs, and a sequence of strings and doubles). For attributes, only four types of data are supported. These types are short, long, double or strings (in zero, one or two dimensional arrays)

TANGO clients can be programmed using only the CORBA API. However, CORBA knows nothing about TANGO and programming at this level means that clients have to know all the details about CORBA programming. The TANGO philosophy is to hide these recipes in an API. The API is implemented as a set of C++ classes or as a Java package. This API also implements automatic reconnection between clients and servers in case of server restart or front-end computer reboot.

8 POGO: A TANGO DEVICE PATTERN GENERATOR

In order to ease the work of device access software programmers, a TANGO device pattern code generator has been written. This generator is called *pogo*. It is a graphical tool built using. Once commands, attributes, properties and device states have been clearly defined, it is possible to use pogo to generate a complete framework of the device pattern implementation. Using a user-friendly graphical interface, the programmer can enter all the commands, attributes, properties and states. The tool will generate C++ or Java classes and an HTML framework for documentation. Obviously, the code specific to the device is still in the programmer's hands! Pogo is also able to analyze a device pattern implementation that it has not generated or to analyze a device pattern that has been modified by the programmer if s/he follows some very basic rules. This allows the usage of Pogo from the beginning to the end when developing a device pattern implementation.

9 JIVE: A GENERIC WEB TANGO DEVICE INTERFACE

Jive is a basic tool used by each TANGO programmer or user. It is made of two different parts, which are:

- A generic device menu. Once a device has been selected, using device configuration commands, Jive displays the list of commands/attributes supported by the device. The user is able to execute any command or to read/write any attribute.
- A graphical interface to the TANGO database. Jive offers a graphical interface to all the database functionality, like defining new properties, updating properties values, attaching a new device to a device server process, browsing devices and properties, etc

The two parts of this tool are written using Java and a servlet/applet pair. The graphical possibilities offered by HTML are not well adapted to the needs of such a program. Therefore, an applet has been written. The drawback of the applet is its loading time and all its security restrictions. In this architecture, the applet communicates with a servlet running within an Apache Web server. Communication with the TANGO device is done by the servlet. The applet communicates with the servlet using the traditional doPost/doGet Hyper Text Transfer Protocol (HTTP) requests. This has a timing penalty but access time is not a key point when testing a device, checking its functionality or browsing the database.

10 ASTOR: ADMINISTRATING A COMPLETE TANGO CONTROL SYSTEM

Once the control system has several device servers an administration tool is needed. Within the TANGO toolkit this tool is named *Astor*. It allows easy starting/stopping of device server processes even on remote hosts. A specific device server called Starter achieves this. The starter device supports commands to start/stop device server processes running on the same host. For correct usage of Astor, each host involved in the control system must have a Starter device server.

For a correct start up/shutdown of a complex TANGO control system, a level can be associated to each device server process. These levels are stored in the database. Starting or stopping the control system using Astor will ensure the sequencing of actions according to these levels.

11 LABVIEW INTERFACE

TANGO has been interfaced to the LabView G language. A dynamically loaded library (dll) on Windows and a shared library on Unix have been written to convert the LabView G programming types to TANGO types. G programs use the shared library Virtual Instrument (VI) to call TANGO. LabView programs written for one platform also run on the other platforms. The LabView TANGO interface is presently supported on Windows, Linux and Solaris.

12 EXAMPLE OF AN OPC DEVICE SERVER

Most Programmable Logic Controller (PLC) suppliers propose an OLE Process Control (OPC) interface with a data server (SDS) running on Windows NT. For a full distributed control system, we need to access this SDS easily from anywhere on our network. The idea was to write (with the POGO pattern generator) a TANGO device server able to read/write data in a PLC through the SDS.

This TANGO device pattern implementation gets the PLC’s I/O addresses as properties from the TANGO database. It could also be used as a low level device pattern implementation for a higher level device server for a specific PLC and/or controlling many PLCs distributed around the accelerator.

13 CONCLUSION

Nowadays, several TANGO device servers are used in the day to day running of the ESRF control system. With the help of the backward compatibility provided by the TACO API’s, it was possible to smoothly incorporate them in our running control system.

TANGO is actually available on 4 platforms presently: Linux (Suse), Windows NT, Solaris and HP-UX. C++ and Java are supported for client and/or server. It uses CORBA release 2.3.

The TANGO IDL has been written since the beginning of the project. Some of the features foreseen in this definition are still missing. This includes the security aspect, the grouped calls and the event system. This is our future task.

Even if CORBA has a steep learning curve, it is easy to use for building simple types of network objects like our Device. The performance of CORBA is more than enough for an object oriented control system. The paradigm of device oriented access has again proved to be very powerful and adapted to control system problems.

TANGO offers significant improvements compared to TACO, e.g. its support for modern protocols (IIOP) and languages (Java and C++), immediate reconnection, and openness to emerging Web technologies.

THE INTEGRATED HV, LV AND LIQUID RADIATOR CONTROL SYSTEM FOR THE HMPID IN THE ALICE EXPERIMENT AT LHC

G . De Cataldo*, D. Di Bari, A. Franco, E. Nappi, INFN sez. Bari and Dep. of Physics,
Via G. Amendola 173, 70126 Bari, Italy
E. Carrone, M. Davenport, A. Di Mauro, P. Martinengo, F. Piuz, E. Schyns, CERN CH 1211
for the ALICE collaboration

Abstract

The complexity and the underground location of the new generation experiments (ALICE, ATLAS, CMS and LHCb) at the CERN Large Hadron Collider (LHC) requires a reliable and user friendly control system to operate such large detectors remotely. Control system experts at CERN are deeply involved in developing the JCOP (Joint Controls Project) 'Framework'[1], a software running in the PVSSII SCADA[1] (Supervisory Control And Data Acquisition) system, that will provide a homogeneous and ready to use tool for the control system developers of the LHC experiments. The High Momentum Particle Identification Detector (HMPID) [1], one of the ALICE[2] sub-detectors, is being equipped with a Detector Control System (DCS) developed within the JCOP Framework. In this paper the basic features and the first results of the DCS prototype are presented.

1 INTRODUCTION

The HMPID is a proximity focusing Ring Imaging Cherenkov (RICH) detector. It consists of a C_6F_{14} liquid radiator and a multiwire proportional chamber (MWPC) with a pad segmented photo-cathode coated with a thin CsI layer acting as photon converter. The MWPC is continuously flushed with CH_4 at NTP, while a CAEN SY1527 crate with the A1821P board, provides 2.1 KV to the anode wires. Seven modules (1.5x1.5 m^2 each) arranged in a cross-like geometry constitute the final detector.

In order to design and implement the HMPID DCS it has been subdivided in five subsystems: High Voltage (HV), Low Voltage (LV), C_6F_{14} Liquid Circulation System (LCS), physical parameters (e.g. pressures and temperatures) and CH_4 gas system. To ensure a safe and easy integration of the control of each subsystem in the HMPID DCS, and later in the ALICE DCS, each individual process is represent as a finite state machine from whom a control program can be deduced. Ref. [2] reports on preliminary studies on GRAFCET[2] (GRAphe Fonctionel de Command Etape/Transition), a graphical language that has been adopted to represent each subsystem, as well as the entire HMPID, as a finite state machine and to deduce the relevant control programs. The key features of GRAFCET are the capability to deal with synchronised parallel processes (Master-Slave configuration) and concurrent sub-processes in a main process. Its application, tested on the LCS [2], was successful although its conversion to the control program (instruction list) for a Siemens PLC, has been done via a dedicated algorithm[2]. The first HMPID DCS prototype operates on the HV, LV and LCS subsystems. The remaining devices will be integrated in the next version. As for the LCS, GRAFCET has been adopted for the design and implementation of the LV control system since the control device of the LV power units are also based on the Siemens PLC. The bubble chart, the more detailed GRAFCET diagram and the preliminary results on the sensitivity of the LV control program are reported in Ref. [3].

Due to the simplicity of the HV process (many controls on voltage, current, board temperature, etc, are already implemented at the level of the crate and board), its representation has been accomplished just through a bubble chart diagram from where the relevant control program has been written in the PVSSII environment.

2 THE INTEGRATED HMPID CONTROL SYSTEM

The DCS prototype is intended to allow the HMPID operator to run and monitor the detector.

Figure 1 shows the Graphic User Interface (GUI) which integrates the HV, LV and LCS sub-systems. This task has been accomplished by exploiting the JCOP Framework which acts as an interface between the PVSSII data points structure and the user application. It provides the user with an environment where logical and physical devices can be defined as well as their connectivity to provide data to the control program and/or directly to the GUI.

* Corresponding author

[1] http://itcowww.cern.ch/jcop/welcome.html

[2] http://richpc2.ba.infn.it

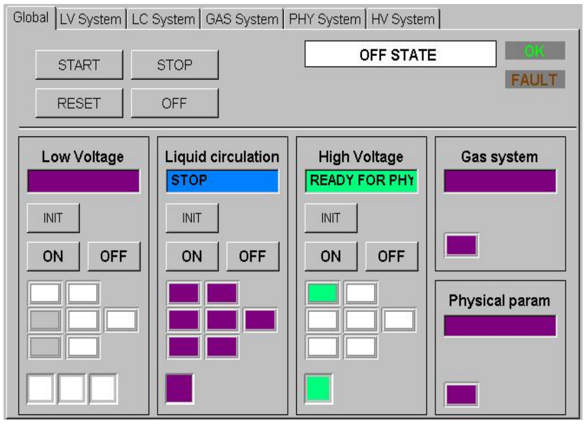


Figure 1: Main panel of the HMPID DCS

The HMPID can be operated respectively through the Start, Stop, Off and Reset action buttons, located on the panel upper-left zone. The lower part of the GUI is used for monitoring via code colors per detector module and per device. For test or debugging purposes each HMPID sub-system can also be operated independently once the main control has been released by activating the OFF button. If a fault status is shown on one module, then clicking on it, a table containing the related list of set and monitored parameters is presented.

If a detailed sub-system monitoring is required, in the top-edge of the GUI there are five tabs. By selecting each of them, the related panel appears.

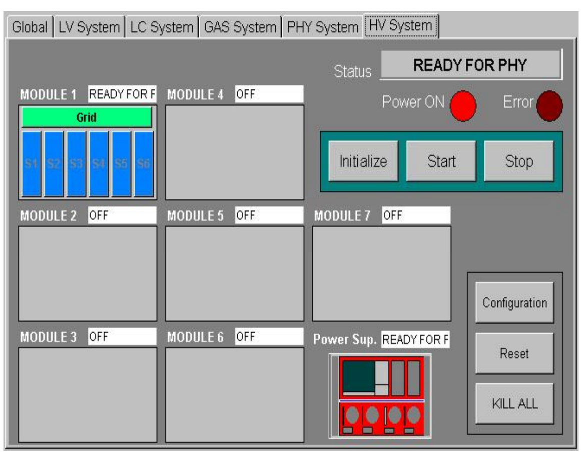


Figure 2: HV Control System panel

Figure 2 shows the HV panel representing all the seven HMPID modules. To deal with some possible HV failure conditions, the MWPC of each module has been split in seven (S1-6, Grid) segments. Each segment is individually powered by a dedicated A1821P board channel. From the panel, clicking on one module's segment, the monitoring and archiving of the voltage and the current values can be inspected (see fig. 5).

In order to modify the detector HV-configuration, by means of the relevant button (bottom-right of the panel), a configuration file, can be uploaded in the PVSS environment. This file contains the HV segments to be switched ON/OFF and related values can be set, e.g. current-limits, voltages, V ramp up V ramp down.

Once the HV configuration has been accomplished, the action buttons Initialize, Start and Stop, operates the HV subsystem. These actions represent a subset of those reported in the bubble chart in the figure 3. A software engine written in the PVSSII language (a script language similar to the interpreted C) makes the system loop through the states, provided the transition conditions are fulfilled. The transitions marked with a "*", are crossed automatically as soon as the actions in the previous state are accomplished.

The bottom-right part of the panel shows the CAEN SY 1527 monitoring icon. Clicking on it, the related panel with all the crate relevant parameters and the housed boards (HV hardware configuration), is presented to the user.

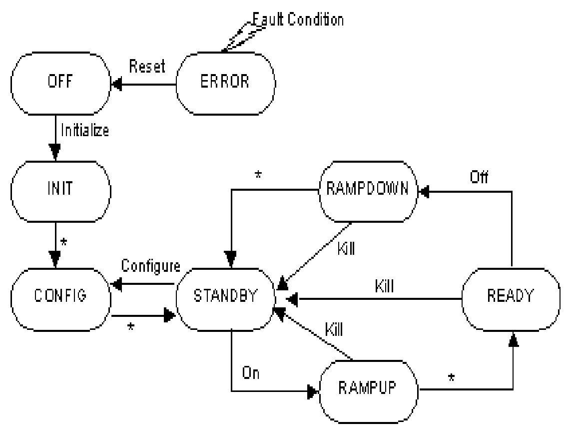


Figure 3: The bubble chart of the HV system.

The data exchange between the HV device and the control program is based on the CAEN OPC (OLE (Object Linked and Embedded) for Process Control) server and the PVSSII OPC client.

Figure 4 shows the GUI diagram of the LCS.

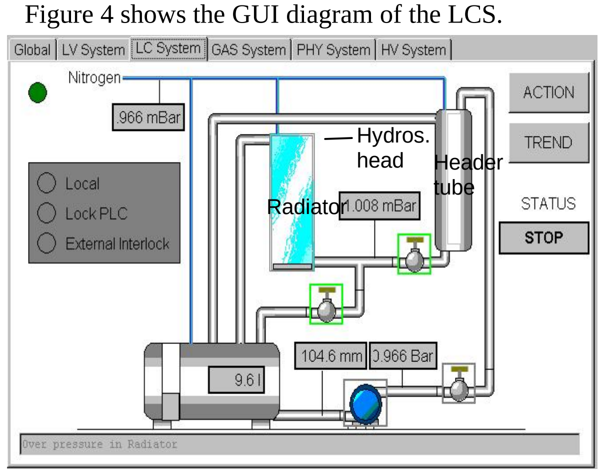


Figure 4: Schematic of the liquid circulation system

It is intended for expert users for detailed monitoring and operation purposes.

From a main tank the C_6F_{14} is pumped via a filter (not shown in the figure) into the header tube where the liquid reaches a maximum level slightly higher than the upper profile of the radiator (hydrostatic head). This allows the radiator to be filled by gravity flow and the liquid to be purified by means of molecular sieve filters while circulating. Clicking on the action button, a panel with an active bubble chart appears and allows the user to operate the system. All the actions are linked to the control program which has been deduced from the GRAFCET diagram and downloaded into the control device, the Siemens PLC S300 series. It consists of CPU, ADC, relays and Digital I/O modules. The program also allows the user to select the LCS manual operation mode, where all the actions are under the direct control of the user. Activating the trend button on the LCS panel, the parameter measurements and data archiving are started (see fig. 6). The data exchange between the control program in the PLC and the GUI in the PVSSII environment, is ensured via the Siemens OPC server and the PVSSII OPC client.

Selecting the LV tab, an operating and monitoring panel similar to the HV, appears. For details on the LV process control refer to Ref. [3].

3 MEASUREMENTS AND FIRST RESULTS WITH THE DCS

Although the HMPID DCS prototype allows the simultaneous control of the HV, LCS and LV sub-system, the LCS and LV systems have been separately tested in the DCS laboratory, and the HV device during a test beam. In figure 5 is reported the monitoring on voltage (red curve) and current (blue curve) drained by the HMPID PROTO-3 during test beam at the T10 PS facility at CERN.

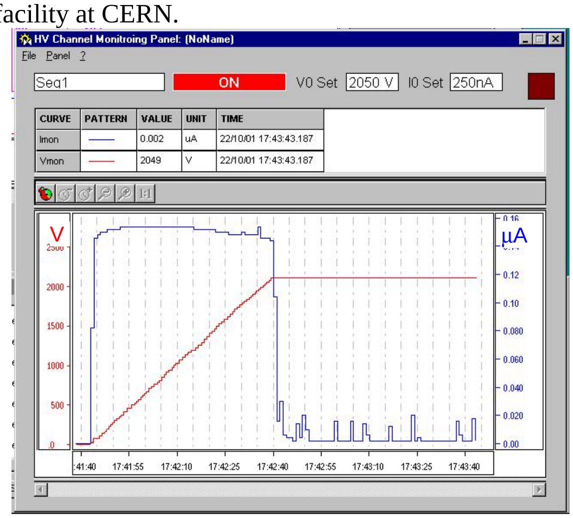


Figure 5. The HV ramp up (red curve) and current monitoring (blue curve) on the HMPID during the T10 test beam at the CERN.

The sensitivity and effectiveness of the CAEN 1527 HV crate and the A1821P (12 ch 3KV, 2 nA current resolution) board, have been verified and found to comply with experiment specifications. Although not yet optimized the DCS version clearly shows the current peaks in the detector in coincidence with the beam spills.

In Figure 6 a complete cycle for the radiator filling and purging is reported. The red curve represents the C_6F_{14} level in the main tank calculated with the pressure measured at the tank output (Rosemount sensor Mod 1151DP3E22), while the blue curve represents the pressure measured at the input of the radiator.

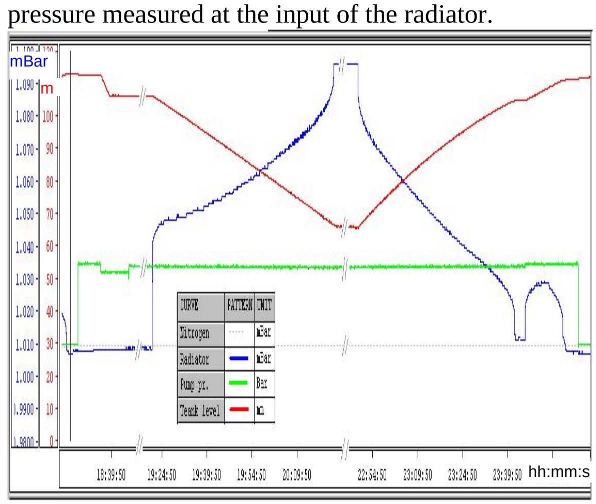


Figure 6: Radiator filling-purging cycle.

The bump at the end of the blue curve is due to the separate purging of the header-tube. Both pressure-sensors are read via a Siemens SM321 8ch 12bit ADC module ensuring a relative sensitivity $\Delta P/P = 3*10^{-4}$ on the relevant full scale values.

4 CONCLUSIONS

The first integration of the HMPID DCS in the JCOP Framework, running in the PVSSII SCADA system, has been accomplished. By means of a preliminary main panel the detector has been operated and monitored. The data collection via the CAEN and Siemens OPC server has proven to be effective and the device sensitivity adequate.

REFERENCES

[1] F.Piuz et al., “The CsI-based ring imaging detector for the ALICE experiment: technical description of a large prototype”, NIM A433(1999) 222

[2] G De Cataldo,, “The control system for the HMPID in the ALICE Experiment at LHC”, LEB 2000, Kracow, September 11-15, 2000.

[3] E Carrone, “Design Specifications and test of the HMPID’s control system in the ALICE Experiment”, LEB2001,Stockholm, Sept.10-14, 2001

TUAP008
physics/0111201

THE ADVANCED LIGHT SOURCE ACCELERATOR CONTROL SYSTEM AT TEN YEARS FROM COMMISSIONING*

A. Biocca, W. Brown, E. Domning, K. Fowler, S. Jacobson, J. McDonald, P. Molinari, A. Robb, L. Shalz, J. Spring, C. Timossi, LBNL, Berkeley, CA 94720, USA

Abstract

The Advanced Light Source [1] was commissioned 10 years ago using the newly constructed control system [2]. Further experience with the control system was reported in 1993 [3]. In this publication, we report on recent experience with the operation and especially growth of the computer control system and expansion to accommodate the new superconducting bend magnets and fast orbit feedback for the ALS electron storage ring.

BACKGROUND

The Advanced Light Source is a third generation 2 GEV Synchrotron radiation source with ports for 60 beamlines. ALS was commissioned in 1990 with the control system reported in [2]. This system by S. Magyary et al. was based on nearly 500 custom made Intel 80186 based microprocessor controllers (Intelligent Local Controllers or ILCs), connected via a bidirectional optical fiber interconnect carrying modified SDLC to an Intel Multibus based central memory architecture (Collector Multi-Module CMM/Display Multi-Module DMM) with dedicated links to 50 Mhz 486 PC's running Windows 3.11. Accelerator Controls communication with experimental beamlines and endstations was via EPICS CA (Experimental Physics and Industrial Control System Channel Access [4]) to process variables instantiated in a multiprocessor VME crate interfaced via Bit-3 bus couplers to the Multibus shared-memory core. VME based EPICS systems were also deployed in the storage ring for vacuum readouts and the early beamline controls were also EPICS based. The initial networks were 10 megabit thicknet wiring to a Cisco 7500 series router with a 100 megabit FDDI link to the LBNL site central network.

Software and Data Flow

Computer control of the ALS is distributed amongst embedded processors deployed near the hardware connected remotely to display computers in the control room.

Control room supervisory and status applications run on a dozen PCs, collectively known as 'console computers' and are developed with a plethora of languages including Microsoft Visual C and Basic, Borland's Delphi and National Instruments' Labview. More active control programs, such as orbit feedback, are developed primarily with MathWorks' Matlab and are deployed on Sparcstations running Solaris.

Programs and applications for hardware control run on embedded processors in one of two configurations: 1) Programs running on the ILC are developed in C or PL/M and are downloaded over the serial network. A configuration database is also loaded to describe the devices to be controlled. 2) The VME and Compact PCI (cPCI) subsystems run EPICS on VxWorks and applications are a combination of real-time databases and C code.

Data and control flows between the console computers and the process controllers follow two distinct paths. The original path traverses a massively parallel serial RS485 and optical fiber network using a simple SDLC based protocol. Data is polled continuously to the central memory (CMM) from the 500 ILCs over 50 multi-drop links. From the central memory, data is polled on demand to the console computers across another set of dedicated serial links. As new accelerator systems are added, and for older systems needing higher performance, the original data path is replaced by a dedicated Ethernet subnet using the EPICS Channel Access protocol.

These different data paths present a challenge for software development on the console computers. Ideally, the differences should be transparent, and significant work has been done to achieve this goal. A dedicated multiprocessor EPICS based Input/Output Controller (IOCs) is bus-coupled to the central memory making the network of ILC data available via the Ethernet. Many applications for the console computers have been developed using an interface library, deployed as a windows Dynamic Link Library (DLL), with an API designed for the original serial network.

* Work supported by the Director, Office of Energy Research, Office of Basic Energy Sciences, Material Sciences Division, U.S. Department of Energy, under Contract No. DE-AC03-76SF00098

Changing and re-testing all these programs to use the channel access API is not feasible due to the large effort required. Instead a compatibility library has been deployed that mimics the behavior of the original library. Since linking is dynamic, applications don't have to be rebuilt.

Both the compatibility library and the newly developed applications use a C language API called Simple Channel Access (SCA), which was designed to make EPICS Channel Access client development less complex. SCA is now widely used throughout the facility, both on Windows and Unix, and has been linked with software development packages such as Matlab and Labview.

CURRENT CONFIGURATION

The Console Computers have been upgraded several times, first to Windows NT 3.51, and later to NT 4.0, and they are now in the process of being migrated to Windows 2000 Professional. The hardware has also been updated several times and now all console PC's are at least 400 MHZ Pentiums. Additionally, three Sun Workstations have been added to the console to handle EPICS based operator interfaces for the Damping Systems and to run Matlab based control software. Matlab has become the high-level programming environment for ALS control and many of the accelerator physics experiments.

Servers

PC and Unix servers have been upgraded several times, from the original Digital 486 and Sparc 2/IPX's to more current PCs and Sun Netra's. The more significant change in the server arena was the consolidation of file services into Network Attached Storage. A Network Appliance 720 performs this duty with 300GB of FibreChannel disks and a 1 Gigabyte network attachment. Backups are to a locally attached SCSI DLT library. This serves both NFS and CIFS clients and runs no user code so availability is extremely high. Rebooting servers no longer takes file systems down.

Equipment Protection System

Equipment Protection Systems (EPS) at the Light Source are implemented using Modicon PLC type equipment. A Modbus serial interconnect has been implemented to EPICS VME crates and a read-only program polls selected PLC memory locations into EPICS Process Variables, making them available to the system and Data Archiving. This facilitates documentation of beam time usage and aids in fault analysis.

Data Archiving was implemented initially using an EPICS diagnostic tool that was not designed for continuous production operation. A more reliable Archiving engine was implemented based on our local SCA library that has proven itself in several years of use. Web and application program interfaces allow data to be mined for many purposes.

Hardware Controllers

Power Supplies and Instrumentation interfacing in the original controls were implemented via the custom manufactured ILCs. They continue to be used, but in applications requiring greater performance they are being replaced with a Compact PCI based solution. Compact PCI was chosen after a careful evaluation of all requirements including the compatibility required with the installed facility racking and cabling. Compact PCI offers significant rear I/O and adequate capability configured in a 3U rack package, matching the existing equipment racking and cabling. COTS (Commercial Off the Shelf) hardware was employed as much as possible, but to meet requirements, a slightly customized commercial enclosure coupled with a custom rear I/O PC board, and a custom Industry Pack (IP) board allowed us to meet ALS requirements with excellent performance at reasonable cost. The IP board carries four 16 bit analog input channels, two 16 bit analog outputs, and two bytes of binary I/O. A pair of these IP cards exceed the I/O capability of one ILC, and four will fit on each IP carrier board. They were designed in-house and assembled by outside vendors. The cPCI CPU selected is a 300 Mhz PowerPC board from Motorola (MCP750) running EPICS on VxWorks. The IP Carrier is from SBS GreenSpring. Fifteen of these chassis are deployed, enough to handle all of the power supplies in the Storage Ring. Presently only the correctors and the main strings are under cPCI control and readouts of some of the beam position monitors, the rest will be moved as needed. Twelve of the cPCI chassis were installed with their own private 100 Megabit network subnet and a fast orbit feedback system with a 1 khz update rate is in development, with beam position information sharing to take place via multicasts over this network [5].

Facility Networks

The original network was based on a Cisco 7509 router. A separate subnet was employed for each beamline sector area, and one for the accelerator controls. This has been upgraded to a Cisco 8530 primary router with 40 Gigabit backplane and 1 Gigabit connections, and a 7513 router was configured as a redundant backup. Managed switches are deployed at each facility sector, and several in the accelerator controls subnets. Unmanaged switches are deployed only to operationally non-critical subsystems and hosts. The number of accelerator controls subnets was increased to three, one each for controls and services plus a separate one for the fast orbit feedback controls. Additional access security is provided by configuring the routers to limit connectivity of the control networks to local subnets. The legacy thicknet network cabling is in the process of being decommissioned, and all ports are being converted to Category 5 10/100 Base-T or better. Wireless 802.11B networking has been installed and is used for debugging from laptops.

Devicenet

The recent replacement of three ALS bend magnets with superconducting units necessitated 18 bit control precision on their power supplies. To achieve this and minimize noise, etc., it was decided they be digitally controlled, and DeviceNet was selected as the interface. After a failure to get the SBS IP DeviceNet card to work properly we changed to an SST 5136 VME board for production operation. Labview DeviceNet boards were used for testing and development.

CORE UPGRADE

The Controls Core Upgrade is currently in progress. The goal is to retire the difficult to support core components of the control system. These have been identified to include the Multibus-1 and –2, the PC SDLC links, and the bi-directional fiber link equipment. The methodology is to develop an EPICS Ethernet to ILC link adapter (hardware, software, and compatibility libraries), and then deploy these throughout the facility, connecting to the copper ILC links already in place. The adapter under development now consists of a VME CPU and an SDLC capable multifunction serial I/O board on IP (both commercial products). The software to communicate with the ILCs is in development now. This project will also require new Operator Knob Panels; a prototype has been developed based on optical encoders, LCD displays, and an Atmel AVR microprocessor.

Current network upgrades include increasing the number of 1 Gigabit links to hosts and retiring all remaining thicknet network cabling.

The corrector magnet power supplies are presently controlled with 16 bit DACs, but slightly smaller step size is needed to minimize beam motion during global orbit feedback. A new rear I/O PC board is under test that adds a second DAC to each control for finer resolution. This fine DAC is to be controlled by the feedback algorithm, while the setpoint is placed in the coarse DAC. A prototype has been constructed and software development for testing is underway.

FUTURE IS NETWORKING

Clearly the Network will continue to be upgraded as beamlines have increasing bandwidth requirements, and the core will likely go to 10 Gigabits and beyond. Host connections of 1 Gigabits will become commonplace, and the performance and reliability of network equipment will improve.

One industry trend in I/O is toward the use of direct Ethernet in the whole price range of devices. When we were selecting DeviceNet for our SuperBend controls one goal we had was to choose a field bus that would facilitate low cost I/O. At that time Ethernet did not meet the cost requirement. Now Ethernet devices are under $100 and an Ethernet protocol stack will fit in a $7 embedded CPU. This low cost makes it possible to build a distributed control and data acquisition system using off-the-shelf modules that have Ethernet interfaces, placing them near the controlled equipment to minimize cabling, and use standard networking to tie the system together. In essence the whole system becomes a high performance network coupling various types of computing and I/O resources together. The bandwidth of this network is sufficient to do things that previously required special and costly single-vendor interconnects. As links between processors reach 1Gigabit and TCP/IP QOS (quality of service) become mature it will be possible to meet more system requirements with standard networking, at lower initial and life cycle cost.

REFERENCES

[1] http://www.als.lbl.gov
[2] S. Magyary et al, 'The Advanced Light Source Control System,' Nuclear Instruments & Methods in Physics Research A 293 (1990) 36-43, North Holland.
[3] J. Young et al., 'Status Report on the 'Advanced Light Source Control System, 1993' Nuclear Instruments and Methods in Physics Research A 352 (1993) 112-113.
[4] http://www.aps.anl.gov/epics
[5] Steier et al, 'Design of a Fast Global Orbit Feedback System for the Advanced Light Source', PAC 2001 TPAH009.

TUAP013

SLOW CONTROL FOR MICROMEGAS AND DRIFT CHAMBERS

F. Gougnaud, R. Azoulay, P. Deck, E. Delagnes, A. Lotodé, Y. Queinec, H. Ren{*}
CEA/Saclay, DAPNIA

Abstract

A new spectrometer is being built for the COMPASS particle physics experiment at CERN. Saclay is in charge of the construction and the operation of two types of gaseous detectors: Micromegas microstrip detectors and Drift chambers [1], [2]. The operation of the detectors requires the slow control of high voltages and currents, and of the thresholds for the Front-End. This control system has been implemented using the Experimental Physics and Industrial Control System (EPICS) Software tool kit. For the high voltages, the system CAEN SY127 was chosen and has been controlled since March 2000. For the thresholds of the Micromegas chambers, a specific VME hardware and a JAVA graphical interface were developed to configure and monitor the thresholds.

1 INTRODUCTION

As part of the COMPASS collaboration at CERN, Saclay is in charge of the building of 2 types of gaseous detectors: 12 Micromegas microstrip chambers and 2 Drift chambers. The slow control for these detectors consists of the control of high voltages and of the discrimination thresholds and is based on EPICS with vxWorks on the VME side and Solaris 8 on the SUN workstation side. At present, this control system is carried out for six Micromegas and one Drift chambers and run during three months last summer. We have used one VME for the Micromegas chambers and one VME for the Drift with a PPC MVME2431-1 CPU on each VME.

2 HIGH VOLTAGES

For the high voltages, it was decided to use the CAEN system, model SY127 because this system has been known to the physicists for a long time and consequently we had several CAEN crates at hand. This system is specifically designed to power detectors such as PM and chambers. Each CAEN crate can house 10 modules and each module has four channels. For each high voltage channel, the user has to give a “Ramp-up” and a “Ramp-down” value and a current limit value. The high voltage will linearly increase or decrease with time, the rate being determined by « Ramp-up » or « Ramp-down » parameters. If a channel draws a current larger than the programmed limit, the channel is signaled to be in overcurrent. Thereupon, a number of actions can be optionally taken. In our case, the channel is switched off. The high voltage will drop to zero at a rate determined by the value of ramp-down for that channel.

In our experiment, two CAEN crates are used. One for the Micromegas chambers and PM, and one for the Drift chambers. For the Micromegas chambers, CAEN modules have a voltage range of 2kV with a current limit of 40µA and have a current resolution of 10 nA. For the photomultipliers, 2 modules A333 (4kV, 2mA) are used. Another CAEN crate controls the high voltages for the Drift chambers with modules A234 (2KV, 200 µA). All these CAEN modules have a maximal voltage protection.

In the past, the physicists controlled locally on the CAEN crate or remotely from a terminal with a RS232 link. Now, the high voltages are controlled from a VME via a CAEN communication module in the VME, model V288, and a CAENET link.

We obtained the EPICS software for CAEN SY127 from SLAC. This software was first written at Jefferson Laboratory. There are MEDM displays for the control of the values of « Ramp-up », « Ramp-down », trip, current limit, voltage limit, and others displays for the real readback of the current and the voltage. For safety reasons, we have separated the displays dedicated to the controls of the high voltages and the displays for the monitoring. In total, at present there are 36 high voltage channels. Next Spring, 6 new Micromegas chambers and a second Drift chamber will be installed. Hence, the number of channels will double. There will be 2 CAEN crates serially. At present, we are limited by the rate of scanning of the channels to one per second. Practically, we get the current for all the channels only once per second.

The EPICS tools Channel Archiver, xarr and Striptool allow to control effectively the high voltage system. Striptool is a real time tool and the Channel Archiver permits to diagnose the behavior of the high voltages in retrospect.

3 MICROMEGAS THRESHOLDS

3.1 Hardware

Each Micromegas chamber has up to 1024 strips by plane. These strips are linked by 16 to a small card that houses each a full custom ASIC called SFE16 designed in our department to answer to specific need for Micromegas chambers. This chip SFE16 contains 16 channels with a low noise preamplifier. Each channel allows to operate one strip. This chip provides also a setting of the thresholds common for 16 channels and a logical OR for the same 16 channels. The slow control is in charge of the command control of the thresholds and of the read-out of the OR to keep the thresholds system independent from the acquisition in any circumstance. The threshold's setting depends on the position of the channel in the detector. Hence, this writing has to be done chip by chip.

A dedicated bus was developed to supply the power to the cards SFE16 and to carry the signals. Around each Micromegas chamber there are 2 busses of 32 SFE16 cards. Consequently, for 12 chambers there are 24 busses. At the head of each bus, a buffer board was designed to do an electric adaptation of the signals. A VME interface board was also built to adapt the signals from the VME to the buffer card and a binary input output VME board ADAS ICV196 is used to write and read bit patterns.

The chip SFE16 is composed of 4 registers with 6 bits for the address, 4 bits for the subaddress and 8 bits of data. One of the registers is the value of the threshold. The other registers are electronic parameters.

The busses are written in parallel and the bit patterns serially. Hence, the purpose of the software is to write 24 bit patterns for the 24 busses of the 12 chambers and these 24 bit patterns are constructed with the same bit of the same register of the same chip of the different busses. To build these specific patterns, a convenient method was to use a file in which all the values of the 32 chips of the 24 busses are written.

The bit patterns are written through a VME binary input output board ADAS ICV196 that has 96 channels. One port of 32 channels is used to write the pattern, another port to read the pattern of bit. There is only one channel for the CLOCK signal and one channel for the LOAD signal.

3.2 Software

The thresholds are written at each spill to solve some problems of discharges of the detectors. On the VME side, an EPICS SNL program receives an interrupt and writes the different registers every 14 seconds (Fig. 1).

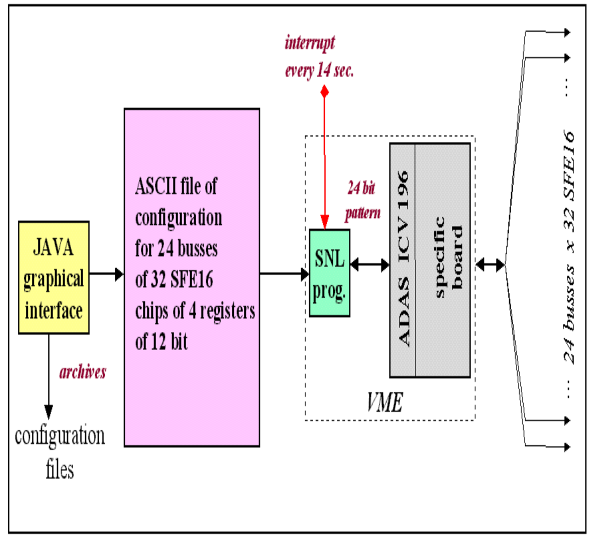


Fig.1 Software for thresholds of Micromegas

A JAVA (JDK 1.2) graphical interface was developed and has two goals. One goal is the "on-line" configuration of the thresholds for all the chambers. Another goal is archiving and restoring configurations. The interface gives the possibility to modify online the threshold of one SFE16 (Fig. 2) or to set the threshold for several SFE16 at the same value or to configure several chambers in the same way (Fig. 3). A specific display is also available for the mapping of all the registers.

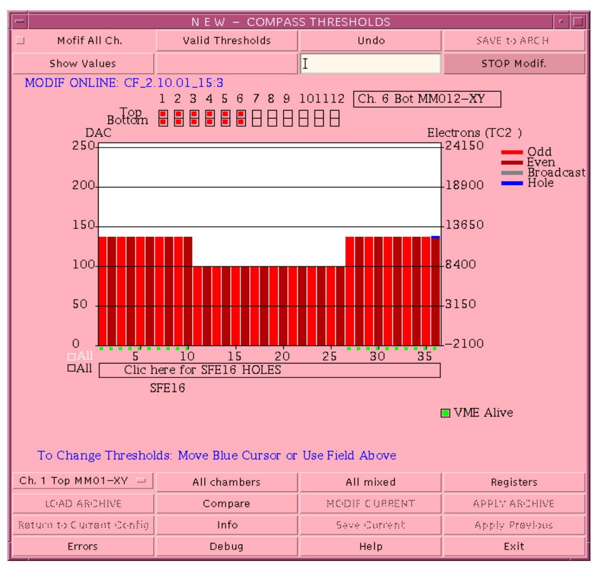


Fig. 2 Display to control 32 thresholds on one bus

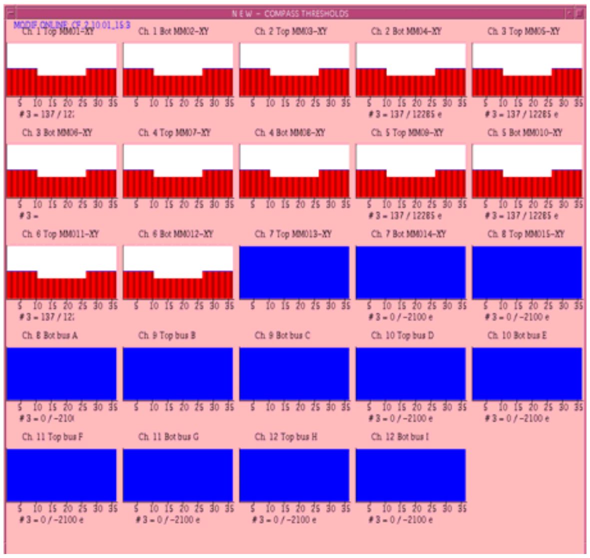


Fig. 3 Display for 12 busses for 6 Micromegas chambers

4 THRESHOLDS FOR DRIFT CHAMBERS

This control consists in sending a negative voltage comprised between 0 and –1600 mV with a step less than 10 mV. The ADAS VME DAC board ICV712 fulfills these needs. For the software, we had only to configure an EPICS database and to create a MEDM display. Therefore, this simple application was rapidly done.

5 LOG MESSAGE QUERY

This program was developed in JAVA and is a real time log message graphical tool. It works from an ascii file gathering all the errors. It provides functions of search. It is useful to query error log messages by giving name of hardware, time of log message: date or time period. This tool also backs up files of log messages.

6 FUTURE DEVELOPMENTS

One future development is the counting of the OR output by the SFE16 chips. The chosen VME board is the SIS3801 multiscaler from SIS GmbH. Another one is the communication software between EPICS with Channel Access and PVSS2 with DIM that would allow to communicate with the COMPASS general control.

7 CONCLUSION

For all the VME ADAS boards, the EPICS drivers were already written. In our control group, it was the first time we had to control CAEN high voltages and the EPICS software we got, contributed to control our detectors since June 2000. The addition of graphical JAVA tools for thresholds and log error messages has answered all the needs of the physicists.

REFERENCES

[1] COMPASS Proposal, CERN/SPSLC/P297, March 1,1996

[2] Micromegas as a large microstrip detector for the COMPASS experiment D. Thers et al. NIM A 469(2001) 133-146

{*} Institute of High Energy Physics at Beijing

TUAP014
nucl-ex/0111003

A SLOW CONTROL SYSTEM FOR THE GARFIELD APPARATUS

M. Giacchini, F. Gramegna, INFN, Legnaro (PD), ITALY 35020, EUROPE
S. Bertocco, SISSA, Triste, ITALY 34014, EUROPE

Abstract

The major part of the GARFIELD apparatus electronics are monitored and set up through a slow control system, which has been developed at LNL. A software package based on Lab View has been dedicated to the setting and control of 16 channels of integrated Amplifiers and Constant Fraction Discriminators. GPIB controllers and GPIB-ENET interfaces have been used for the communication between the Personal Computer and the front-end of the electronics.

1 INTRODUCTION

In the last decade more and more sophisticated and complex apparatuses have been built [1] to meet modern nuclear physics requirements. These detectors (see Fig. 1) have been designed in order to both cover a large fraction of the solid angle, with a high granularity in order to study precise angular correlations between the reaction partners, and to be capable of providing good energy and charge (and/or mass) resolution for a complete identification in a wide range

More compact electronic modules have been therefore developed following the growing complexity of the apparatuses. A simplification of the hardware (no elipot, no manual control, few trimmers etc.) permits the handling of a larger number of channels per module (16-32). But the integration of modules must not detract from all those characteristics necessary for nuclear physics studies, as far as linearity, resolution, possibility of setting different values of parameters like gain, shaping time, polarity etc. The read-out and process of a large (500-1000) number of channels, with almost the same performance as standard modules, has become a requirement.

For this reason several multi-channel analog and digital modules have been newly developed [2]. A remote control system becomes indispensable to set- up the module parameters and to keep them under control during the measurements.

We will describe in this paper the slow control system of the GARFIELD apparatus, based on a LabView platform, which has been developed for the CAEN C208 Constant Fraction Discrimination and the CAEN N568 Amplifiers [3] using the KS 3988 GPIB Crate Camac Controller [4] and C117B High Speed Caenet Camac Controller.

2 THE CONTROL SYSTEM

2.1 The design

GARFIELD [5] is a composite and multipurpose apparatus made by different detector groups:

- an annular three stage telescope, divided in eight sectors along the azimuthal direction, made with Ionisation Chambers, Silicon detectors (300 μm thickness), CsI(Tl) crystals.
- a TOF system based on three Position Sensitive Parallel Plate Avalanche Counters (20x20 cm^2), for fragment mass measurements in dissipative collisions.
- two drift chambers, placed back-to-back with respect to the target, each of which contains about 90 micro-strip on glass gas detectors and about 90 CsI(Tl) crystals.

Close to 500 preamplifiers are used for the read out of the detectors and their signals are fed into the 16 channel shaping amplifiers CAEN mod. N568B. For each input signal three outputs are provided: two correlated linear outputs, for which shaping time, gains output polarities, pole zeroes can be set by an external control and one fast output at fixed gain and polarity. The linear signals are then fed into Analog-to-Digital Converters, based on VME/FAIR bus [2], while the fast output has to be sent to the 16 channel CAEN Constant Fraction Discrimination Mod. C208.

A good and efficient control system is therefore necessary to handle the high number of channels: a LabView application has been developed for this purpose.

The experimental environment is distributed as follows: the apparatus is located in the experimental area, in a scattering chamber under vacuum. All the electronics (up to the acquisition front-end) are mounted in the experimental area, close to the scattering chamber. The “acquisition data” room is connected to the experimental area via Ethernet, both from the acquisition system point of view (Optical fibers), and from the slow control system (coaxial cables).

2.2 Network Idea

The control system has been developed taking into account the possibilities offered by a network environment, like for example the utilization of Internet Tools to display the status of the system and to notify possible problems and faults.

The system controls about 500 channels for both the Constant Fraction Discrimination channels and the Amplifiers and it runs on a Window 2000 PC, connected to the hardware in the experimental area through a GPIB-ENET interface. This interface transparently handles the data transfer between an Ethernet-based TCP/IP host and the GPIB. Through GPIB-ENET multiple hosts can share a set of GPIB instruments or a single host can control several GPIB systems. The GPIB-ENET converts a computer equipped with an NI-488.2 driver and an Ethernet port into a GPIB Talker/Listener/controller. The Maximum GPIB transfer rate is 100 kbytes/s. The limited rate is due to the fact that we are using a general LAN of the institute, and not a reserved one. In this case GPIB-ENET is used to control the KS 3988 Crate Camac Controller from a PC.

The control software has been developed using LabView, by National Instruments [6]. LabView is a graphical programming environment developed for data acquisition, control data analysis, data presentation. The graphical programming method used is completely flexible, becoming very close to a powerful programming language, without reaching the same difficulty and complexity. A basic library of graphical instruments, which permits to manage GPIB communication quite easily, is also provided by LabView

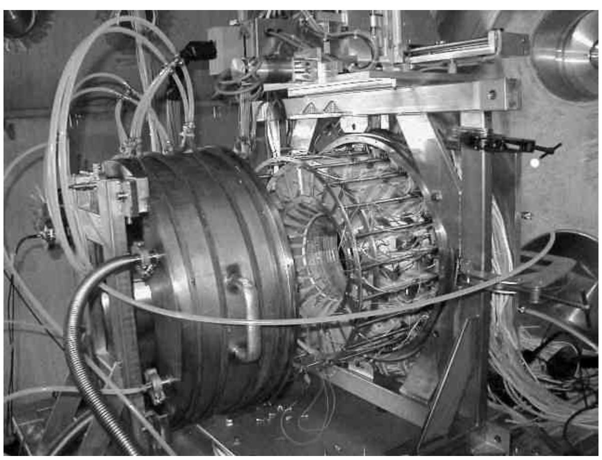


Figure 1: Detector

2.3 Start Up

A general set-up of all the parameters is required at the beginning of each experiment: this is a quite slow action because it concerns all the channels and almost all the settable parameters. This action is an initialization necessary to be sure some standard parameters will be automatically set (for example the very last parameters used at the end of a previous run).

2.4 The data

The data stored in a data base file are sent to the module front-end and a comparison checking the conformity between the parameter values in the data base and the setting in the module is performed. The status column gives out an error if differences are found. Problems can derive due to network bad transmission or malfunctioning of individual channels.

2.5 The GUI

The control system developed on this basis allows the handling and check of the systems parameters through a Graphical User Interface (GUI) (see Fig. 2).

The GUI can then provide these control functions:

- *Set-up of parameters*, chosen by the user, for individual channels or for groups of channels. Different channels can be grouped from the users, depending on the experimental set-up (grouping by kind of detectors, sectors or rings of the apparatus etc.).
- *storage* of the set parameters in an Excel spreadsheet used like a data-base.
- *check of the current status* of the apparatus, reading of the parameter values in the modules via TCP/IP and comparison between read data and data stored in the database.

The control system provides some Internet based tools to monitor the module status and to perform the alarm notification.

The status monitor service allows the display of the updated database spreadsheet in a Web Browser, using an HTTP server. It converts the database file into a HTML file, exploiting the LabView Internet toolkit and it upgrades the file at each significant event. This file can be watched by means of a Web Browser, which making use of a java applet, reloads the file periodically. If the monitoring is performed using Internet Explorer while loading the file, an Informative Microsoft Agent appears.

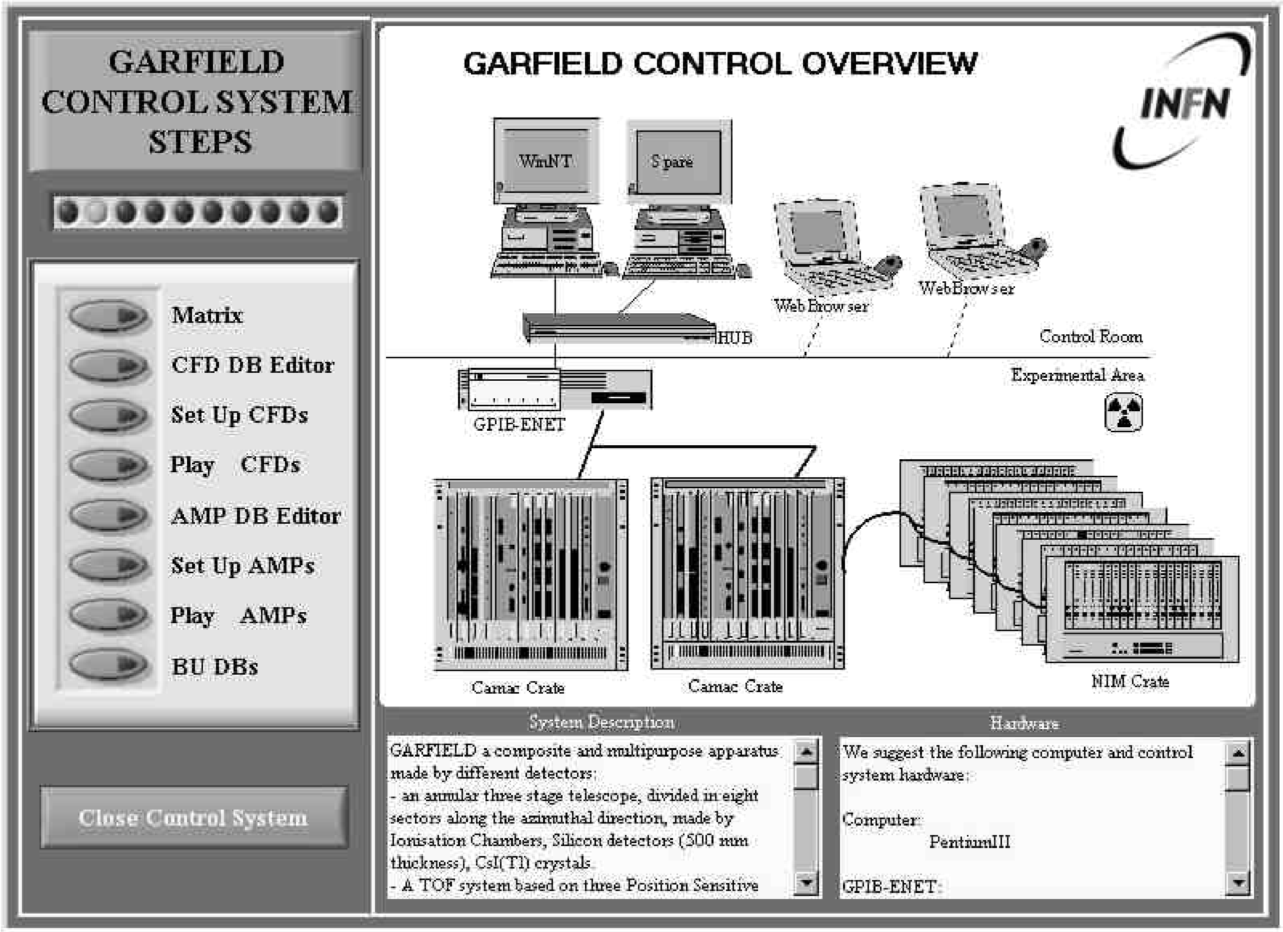


Figure 1: The GUI

This Agent can announce alarms by mean of a speech output. Alarms can also be announced by an e-mail notification of failure performed by using the LabView Internet toolkit.

2.6 Conclusions

A slow control system for the Garfield electronics has been developed to set up and monitor Amplifiers and Constant Fraction Discriminators parameters. The architecture of the control system was designed to be easily applied to different experiments. Some future developments are in progress with regards to High Voltages systems. The Lab View platform is a powerful graphical programming environment, which permits a wide field of applications.

REFERENCES

[1] F. Gramegna et al. A389(1997)474, R.T. De Souza et al. NIM A295(1990)109,
I. Iori et al. NIM A325(1993)458, J. Pouthas et al. NIM A357(1995)418

[2] R.Bassini et al NIM A305 (1991) 449, R. Bassini et al IEEE Symposium S. Fe (1991) , A. Ordine IEEE Trans. On Nucl Scien. 45(3)1998

[3] CAEN Catalogue

[4] KinetikSystemCorporation Catalogue

[5] GARFIELD:a General ARray for Fragment Identification and for Emitted Light particles in Dissipative collisions – to be published

[6] http:/www.ni.com

[7] ‘Costant fraction discriminators control for Garfield’ - LNL-INFN(REP)- 160/00

TUAP015
physics/0111043

THE ADVANCED PHOTON SOURCE INJECTOR TEST STAND CONTROL SYSTEM

J.F. Maclean and N.D. Arnold, Advanced Photon Source, Argonne National Laboratory, Argonne, IL USA

Abstract

The Advanced Photon Source (APS) primary and backup injectors consist of two thermionic-cathode rf guns. These guns are being upgraded to provide improved performance, to improve ease of maintenance, and to reduce downtime required for repair or replacement of a failed injector. As part of the process, an injector test stand is being prepared. This stand is effectively independent of the APS linac and will allow for complete characterization and validation of an injector prior to its installation into the APS linac.

A modular control system for the test stand has been developed using standard APS control solutions with EPICS to deliver a flexible and comprehensive control system. The modularity of the system will allow both future expansion of test stand functionality and evaluation of new control techniques and solutions.

1 INTRODUCTION

The Advanced Photon Source (APS) is a third generation light source that provides high brightness x-ray beams to a user community. The main purpose of the APS injector test stand (ITS) is to test and characterize injectors destined for use in the APS. It is anticipated that the injectors tested will be of varying designs. These requirements dictate that the control system must be easily adaptable for reconfiguration of the test stand.

The first task of the test stand is to test and characterize replacements for the APS main injectors. Since the provision of replacement injectors had some urgency it was necessary to find a way to implement control quickly whilst allowing for expansion and reconfiguration at a later date. It was therefore decided to use standard APS control components where possible but not to preclude new solutions if they were warranted on technical, cost, or other grounds. Once a functioning system was completed, it was anticipated that the test stand would provide a good opportunity to test control solutions that are new to the APS. It would allow testing in an operational environment with the benefit of not impacting the operation of the APS.

2 TEST STAND

The test stand is located in its own shielded room, which is approximately 5.5 m by 3 m in size. The beamline is mounted on an optical bench of approximately 3 m by 1.2 m. Items requiring control include many of the elements found in the main APS linac:

- Injector – the gun under test
- Magnets – correctors, quadrupoles, and dipoles
- Vacuum equipment – pumps and valves
- Beam scraper
- Current monitors – transformers and Faraday cups
- Video cameras
- Galvanometers

The beamline is described in [1], and the main user control screen reproduced in Figure 1 gives an indication of the layout of the line. Figure 2 shows the main components of the control system.

3 CONTROL SYSTEM

3.1 General

The injector test stand control system uses EPICS [2], which is used for the control system of the APS. One input/output controller (IOC) is used to control the ITS. This is a Motorola MVME162 CPU mounted in a 6U VME chassis. The chassis is physically located in a room adjacent to the test stand. The IOC is connected to the APS control local area network (LAN) via a dual redundant Ethernet link. Mounted in a 19-inch rack with the IOC is other ITS control hardware. This includes:

- Video camera driver chassis
- Stepper motor drive chassis
- Allen-Bradley 1771 chassis
- Tektronix TDS7404 oscilloscope

3.2 Links to the APS Control System

Although it is to a large extent independent from the APS control system, the ITS does depend upon it for a number of facilities.

- Control LAN and infrastructure. The ITS IOC is connected to the main APS control LAN and servers. This allows control, monitoring, and development from any workstation on the subnet.
- Timing. A signal is sent to the ITS IOC from the APS timing system ten microseconds prior to an rf pulse.
- Video. The video signal from the ITS is sent to the central APS video multiplexor for distribution.

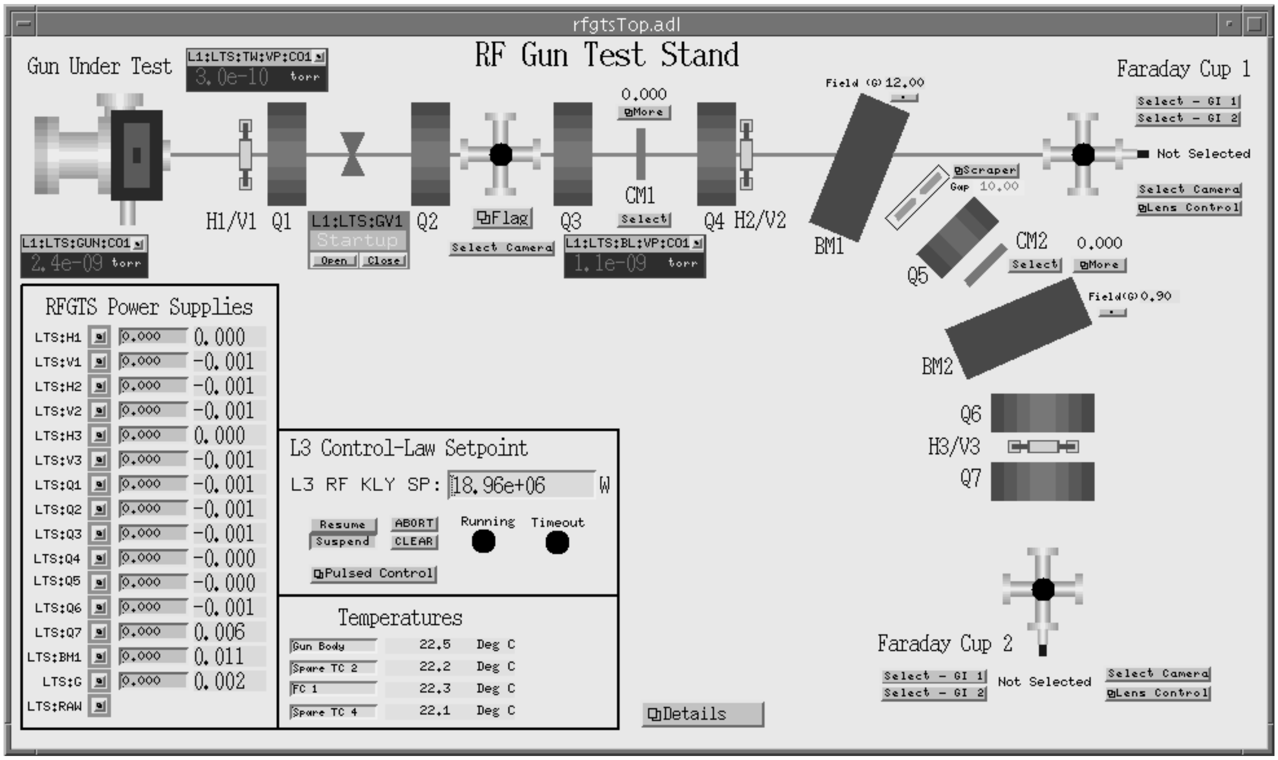


Figure 1. Main ITS control screen, showing main components.

- Interlock. Vacuum or gun cooling faults will cause the rf to trip off.

3.3 Input and Output

Input and output (IO) are done via VME cards in the IOC chassis. EPICS support already existed for all IO cards used, and all but one card had previously been used at the APS.

The one VME device new to the APS, a Berkley Nucleonics BN950 digital delay generator, provides a good example of the ITS being used as a test bed for introducing new equipment. In this case, a message to the EPICS "tech-talk" mailing list quickly revealed that EPICS device support had already been written at the Berlin Electron Synchrotron (BESSY). This is an example of a benefit from the EPICS collaboration.

The main individual elements of the ITS and the interface methods used to control them are:

- Magnet power supplies. These are controlled by a standard APS power supply control unit (PSCU) that communicates with the IOC over a Bitbus fiber.
- Vacuum pumps. These are controlled by a standard APS interface unit connected to the IOC by Bitbus fiber.
- Beam scraper. Two scraper blades are driven by stepper motors controlled by an Oregon Micro Systems OMS58 VME card. Positional feedback is provided by linear potentiometers read by a 16-bit analog-to-digital converter industry pack module.
- Current monitors. Four current monitors are multiplexed into two APS gated integrator VME cards. Timing signals are provided via the BN950 delay generator. Spare multiplexor channels are available for future expansion.
- Galvanometers. These are GPIB devices interfaced to the IOC by APS GPIB/Bitbus converters.

3.4 Databases

Much of an EPICS-based control system is typically built with databases. The ITS is no exception. Each controlled hardware component has a database associated with it. Since most of the hardware components were of a type that had previously been used in the APS linac, databases already existed for them. We were thus able to reuse software in the form of these databases. Many required little more than the renaming of record instances via macro substitution and the changing of some physical parameters.

This brought the usual benefits associated with software reuse in terms of reduced development time, reduced debugging time, and improved reliability of the finished product.

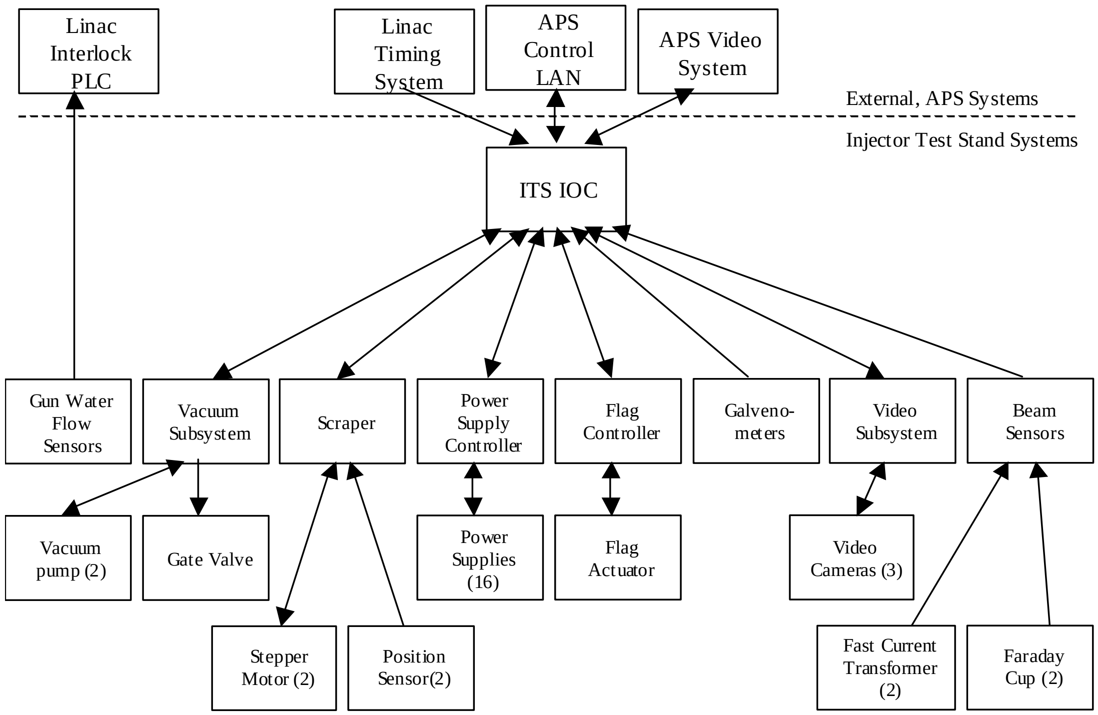


Figure 2: Block diagram showing major ITS control system components.

3.5 Displays

The engineering control screens were produced using MEDM, which is the APS standard user interface tool. This was another area that permitted some software reuse. Control screens for some elements, such as the scraper, power supplies, and galvanometers, were reused from screens originally created for use in other areas of the APS. Configuration of process variable names is achieved by passing parameters when launching the screens.

4 RESULTS AND FUTURE WORK

An effort was made to integrate the control system to the hardware as soon as the hardware was available. The success of this effort was shown by the fact that beam was obtained from the first injector tested on the test stand less than two hours after rf energy was applied to it. This included the time required for “rf conditioning” of the injector. In subsequent testing of a second injector, beam was obtained in less than one hour after applying rf power.

APS physicists were very pleased with the ease with which the control system was brought into use. This can largely be attributed to the reuse of software modules and the reliability of EPICS.

Besides tests of new injectors we intend to use the ITS for tests of new control equipment, including a replacement for the standard APS power supply control unit.

5 ACKNOWLDGEMENTS

Many people contributed to the ITS control system. The authors would particularly like to acknowledge the hard work and assistance of the following people: William Berg, Richard Koldenhoven, John Lewellen, Josh Stein, and Jim Stevens.

This work is supported by the U.S. Department of Energy, Office of Basic Energy Science, under Contract No. W-31-109-ENG-38.

6 REFERENCES

[1] J. Lewellen et al. “The Advanced Photon Source Injector Test Stand” PAC2001, Chicago, June 2001, to be published.

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

TUAP016
physics/0111073

DESIGN OF THE MPRI CONTROL SYSTEM

J.C. Collins, M. Ball, B. Broderick, J. Katuin, Wm. Manwaring, N. Schreuder
IUCF, Bloomington, IN 47408, USA

Abstract

The Indiana University Cyclotron Facility (IUCF) is in the process of building the Midwest Proton Radiation Institute (MPRI). This involves refurbishing the 200MeV cyclotron and building new facilities for the purpose of providing clinical treatment of human cancer tumors. First patients are expected in the Spring of 2003. This paper presents the design and implementation to date of the controls, hardware and software, for both accelerator and treatment areas. Particular attention is placed on issues of personnel safety and control system security, development of inexpensive VMEbus boards with emphasis on the use of MicroChip PIC processors, beam diagnostics and monitoring and the use of commercial robots and vision systems for patient positioning.

1 INTRODUCTION

Faced with the end of support for physics research on its 200MeV cyclotron, IUCF has chosen to pursue further use of the accelerator for the treatment of various cancer tumors. To this end, the cyclotron system is being modified to produce protons at the single energy of 205MeV, the cyclotron experimental hall has been cleared and a trunk beam line with separate treatment beam lines are being installed.[1]

Controls for both the cyclotron and all beam lines will use an architecture very similar to that used in the most recent IUCF projects [2]. Two minicomputers, one for the cyclotron, one for all treatment beam lines, provide the user interface and all related processing. Each uses a PCI-VME interface to communicate with a master VMEbus crate, which is connected to geographically distributed crates via fiber optic bus extenders. The minicomputers run OpenVMS and Vsystem. On/Off controls and interlocking are done with PLCs, using centralized processors and distributed I/O modules.

The full plan calls for three treatment rooms, two of which are to hold gantries. Each treatment room will have its own, independent room control and dose monitoring computers. In all rooms, accurate and reproducible patient positioning will be achieved using commercial robots. But because robot motion has more degrees of freedom than needed, an independent computer vision system will be used to track and interlock robot motion. Finally, all patient treatment parameters will be supplied to the various control computers from a medical computer system.

2 HARDWARE

2.1 Beam Line Control

The cyclotron and beam line control computers are Compaq DS10 AlphaStations, each with an SBS Bit3 PCI-VME interface to a 6U VME crate. In turn, this crate contains VMIC 5531L modules which connect to remote 6U and 3U VME crates via fiber optics.

For DACs and ADCs, the standard debate between buying commercial or designing in-house was decided in favor of the latter option. Using FPGAs and out-sourcing construction allows even a small facility to do such a project. This option is not more expensive than the commercial one when requirements for good galvanic isolation and a high degree of modularity (for quick repair) are included.

In fact, we have developed four single width VME boards:

- DAB (Data Acquisition Board) - 3U board contains one DAC and six ADC channels.
- SDPM (Serial Dual Port Memory) - 6U board connects to up to eight PIC processors, each connected to a serial (e.g., RS-232) device.
- ADPM (Analog Dual Port Memory) - 6U board connects to up to eight PIC processors, each having four channels of 12-bit ADC.
- DIO (Digital Input and Output) - 3U board with two 16-bit channels; supports VMEbus interrupts.

Interlocked functions using digital I/O are performed with an Allen-Bradley PLC and Flex-IO modules.

2.2 DAB

All DAB channels are 2’s complement, 16 bits and ± 10V. The analog section of the board is optically isolated from the VME power, ground and signal bus. The six ADC channels are paired internally, each pair sharing an external connection. All channels have low-pass filters, but with the response of one (slow) channel in each pair having greater attenuation with frequency than the other. The slow channel will be quiet and used for normal operator displays, while the other is available to diagnostic software looking for high frequency components signaling problems.

2.3 SDPM, ADPM

These boards are variants of the same basic design. Originally conceived to be an inexpensive interface to devices with slow, serial interfaces, the boards present two banks of 1KB to the host computer. While the PIC processors gather data and store it in one memory bank, the host computer can read data from the other. Bank switching is under host control. Adding 12-bit, 2's complement ADCs to the remote PICs created an inexpensive method of gathering slow analog data close to the beam line. The RS232 or analog section of each board is optically isolated from the VME power, ground and signal bus. Figure 1 shows how beam line elements are connected to the control system using DAB and ADPM.

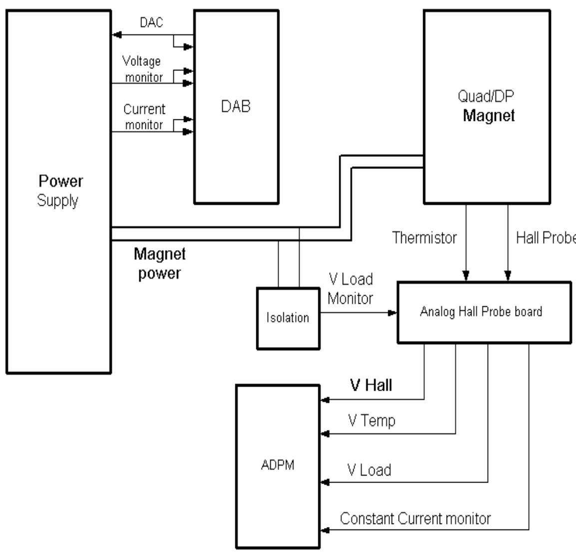


Figure1. Beam Line Element Connection.

2.3 Timing

The few timing signals required will be supplied by a single sequencer module[3]. Depending on how it is configured, the sequencer can provide up to 32 signal outputs with 1•sec resolution or DAC output synchronized with timing signals. The latter setup may control beam intensity when using multiple treatment rooms.

3 SOFTWARE

3.1 Overview

The control computers will run OpenVMS and Vsystem, a combination IUCF has used for many years. Each treatment room control computers will run a version of Windows because of the need to run commercial software. The dose monitoring computer will run a small, realtime OS which has not been selected yet. All these computers will have Ethernet connections and be separated from the rest of IUCF by managed switches and from the rest of the world by firewalls.

3.2 Cyclotron Control

In December, 2000, the PDP-11/44 which had run the cyclotron since 1985 was replaced with a DS10. An interface between VME and the original DIO control bus was installed and about 20% of the control hardware was replaced before the cyclotrons were brought back into operation using Vsystem in March, 2001. We will continue to replace portions of the old cyclotron controls as time and access permit.

3.3.Beam Line Control

Basic beam line control will be based on programs previously created and tested at IUCF. Major changes or additions to that software include:

- 2-of-3 Checking – As shown in Figure 1, each beam line element will have power supply current, load voltage and a temperature corrected Hall probe magnetic field readout. All three values must be within acceptable ranges before operation is allowed. There will be a procedure whereby the operator will be able to specify that operation is allowed with one out-of-range value for a short, fixed time (until the next maintenance period).
- Machine state restore – There will be one "golden" set of DAC and ADC values defining proper accelerator and beam line operation. That will be the only set supported for computer restoration.
- Energy setup – A degrader in each treatment beam line will be used to change the fixed cyclotron output proton beam energy to that desired for a specific treatment. We will set each treatment line element by interpolating in a table of empirical values and check that beam transmits through the line as expected before allowing a real treatment to start.
- Security – No remote logins will be permitted for the treatment room controllers. Restrictions will be placed on those network nodes from which users may log into the control system. All controls, except a beam stop and closed loops, will be disabled while a treatment is underway.

4 DIAGNOSTICS

4.1 Beam Position Monitor (BPM)

BPMs[4], non-intercepting transducers that provide both position and intensity information, are the primary beam tune-up and monitoring tool. The BPM system has four major sections, the pickup, the radio frequency (rf) amplifiers and mixers, the analog and conversion section, the analog-to-digital converter (ADC), and the software. They provide ±0.2mm position resolution over a ±12mm area and 10% intensity accuracy for beams from 10nA to 2μA. Operator displays are typically a set of bar graphs showing all readouts for a single beam line.

4.2 Harp

A harp is a multi-wire device which can measure the beam profile in one transverse plane using either secondary emission electrons or ionization (gas harp). Our harps will use a wire spacing of 0.5mm and operate between 10nA and 2μA beam current. Harps are usually mounted in X,Y pairs and operators are shown data from both as wave forms.

4.3 Wire Scanner

This is a wire helix inserted in the beamline at 45º and rotated so as to cross the beam once in each transverse plane on each rotation. Wire scanner enable and selection will be done with an in-house built controller and multiplexer which enables an operator to select any two wire scanners. Operators are shown X,Y results for the selected wire scanner(s).

4.4 Multi-Layer Faraday Cup (MLFC)

Used to characterize the energy and energy distribution of the beam, the MLFC consists of a solid block followed by insulated plates[5]. The set of currents read from the plates can be used to determine if the beam energy properties are correct. The MLFC will be able to resolve energy drifts >100KeV and energy spread changes >500KeV using currents between 10 and 500nA.

5 PATIENT POSITIONING

For treatment, the patient (actually, the tumor) must be accurately and reproducibly positioned with respect to the proton therapy beam line. This will be accomplished using a Motoman UP200 robot and XRC controller. This robot provides six degrees of freedom, a rated payload of 200kg and positioning accuracy of ±0.2mm throughout its 2446mm working envelop. A precalculated and tested (simulated) robot job file defining all patient-specific robot motions will be part of the patient treatment prescription obtained from medical records when the patient is identified. Local, manual control of the robot will be provided through a wireless, handheld PC.

A robot is almost perfect for the patient positioning job except that its motion has six degrees of freedom, i.e., it is capable of turning the patient upside down. To stop such undesired movements, the patient support device (bed or chair) will be fitted with an acceleration and tilt monitoring system independent of the robot controller. The patient support device will also have special markers attached which will be watched by a Vicon 460 Vision System. The Vision System uses infrared to track each marker and report its position many times per second. Knowing the expected robot path and the fixed geometry of the room, we can determine if the robot is doing something unexpected or about to hit an object and turn the robot off.

REFERENCES

[1] D. Friesel, et al., “The Indiana University Midwest Proton Radiation Institute”, PAC2001, Chicago, IL, June 2001.

[2] J.C. Collins, et al., “The CIS Control System at IUCF”, PAC1999, New York, NY, March 1999.

[3] W. Hunt, “Cooler Injector Synchrotron Control Hardware at IUCF”, PAC1999, New York, NY, March 1999.

[4] M. Ball, et al., “Beam Diagnostics in the IU Cooler Injector Synchrotron”, 7th Beam Instru. Workshop, Argonne, IL, 1996.

[5] N. Schreuder, “The Development of a Multi-Layer Faraday Cup to Characterize the NAC Proton Therapy Beam”, PTCOG-XXXII, Upsala, 1999.

TUAP017

CONTROL SYSTEM FOR THE DIAGNOSTIC NEUTRAL BEAM INJECTOR FOR THE TCV TOKAMAK

A.A. Ivanov, M.V. Kollegov, V.V. Kolmogorov, E.A. Kuper, A.S. Medvedko, A.N. Shukaev, BINP, Novosibirsk, 630090, Russia

Abstract

The diagnostic Neutral Beam Injector (DNBI) for the TCV tokamak, Plasma Physics Institute, Lausanne, was developed and commissioned by the BINP team in 1999. The DNBI is capable of providing a beam of hydrogen atoms of 50 kV maximal energy, with an equivalent beam current of up to 1 A. The injector is equipped with all the power supply units needed for operation and for control, including a 50 kV, 2.5 A modulator and 10 kW, 4.6 MHz RF amplifier. The output beam can be extracted continuously over a 2 second period or as an ON/OFF modulated sequence with arbitrary pulse repetition frequency and minimal pulse/pause duration of 2 ms. An increased noise level looks like a normal condition for all subsystem operation. The control system of the DNBI is developed under the pressure of these specific conditions with increased noise. The system includes 8 DAC, up to 64 ADC channels and 64 channels of In/Out digital status control. It operates as a self-sufficient system with minimal data exchange with the global control system of the tokamak, so it becomes flexible to adopt arbitrary external control system.

1 INTRODUCTION

The TCV is a tokamak device built at the Plasma Physics Institute, EPFL, Lausanne, Switzerland. It operates with two-second "shots" with a two-minute repetition rate. A variety of plasma fusion diagnostics, such as Charge Exchange Recombination Spectroscopy, Motional Stark Effect and Beam Emission Spectroscopy are widely used in tokamaks [1]. The typical diagnostics requirement is to have the injected beam of neutral particles (Hydrogen) with energy of 20 kV to 55 kV and equivalent current ~ 1-2A. At times, the spectroscopy methods need to have a continuous beam, at other times a modulation of the diagnostic beam intensity, which enables one to improve the signal to noise ratio because of the synchronous detecting of signals.

2 DNBI MAIN SUBSYSTEMS

The Dignostic Neutral Beam Injector (DNBI) is a device required for these kinds of diagnostics. It comprises the injector ion source, the neutralizer cell, ion bending magnet, residual ion dump, aiming device, two cryo-pumps and thermocouple arrays housed inside a cylindrical vacuum tank. The equipment located in the injector tank provides both regular and repair procedures for injector handling. These procedures include vacuum pumping and pressure monitoring, cryo-pumps refill, ion source aiming, and calorimetric measurements for control of the beam parameters. The injector is equipped with all the power supply and measuring units needed for operation and control.

The major element of DNBI is an ***injector ion source***. It consists of a special assembly of grids with high potentials distributed between them, so it operates like a multi-aperture electrostatic accelerator. A plasma emitter in the ion source is produced by an inductively excited RF discharge in a cylindrical ceramic tube. A special gas valve to puff the hydrogen is installed close to the ion source. A resonant multi-turns RF antenna operates at about 4.6 MHz frequency with up to 5 kW RF-power absorbed in the plasma during the "shot". A non-homogeneous magnetic field is applied into the plasma box to obtain the required homogeneous profile of the ion current density at the plasma grid area. The grids are mounted on the water-cooled flanges enabling the full heat removal from pulse to pulse as well as partial heat removal during the injection pulse. The thickness of the molybdenum electrodes is chosen to be 2mm and 4mm for different grids. As a result, the ion source is very sensitive to the energy dissipated by grids, especially due to breakdowns. Dissipation during one breakdown should be in the level of a few joules. For the DNBI it was specified to allow less than 16 arc breakdowns during one "shot". It is controlled by High Voltage Power Supply (Modulator) that switches OFF the output power within 200 μsec under the local interlock command.

The Modulator is based on the power converter system with switch mode PWM technology [2]. It can provide an output voltage of up to 55 kV with 180 kW of power during of the 2 sec. "shot". The High Voltage (HV) part of the Modulator is placed close to Injector, directly inside the TCV bunker. The Low Voltage (LV) part is placed at the electrical equipment zone, up to 50 m from the Injector. Both HV and LV parts of Modulator consists of six identical cells. Each cell includes an IGBT inverter operating at a frequency of 5 kHz, a high-voltage transformer, a rectifying diode bridge and a capacitor filter. Outputs of the cells are set up in series. The design has small stored energy in transformers and cables that allows reliable protection for the grid system of the Injector. A modulator can perform an ON/OFF amplitude modulation of the output voltage with a rise/fall time of less than 200 μsec. The minimal time intervals of the «OFF» state should be longer than 2 ms, maximal time interval of the «ON» state can be up to 2 sec. To avoid the influence of this modulation on the mains a special dummy load is used inside the Modulator. It works like a 180 kW DAC, accepting the "switched OFF" current of real load during the "OFF" stage of modulation. Of course, the RF power in the antenna is modulated too, but the "low level" of the RF in the antenna is independently regulated.

Beam profile monitor. At the exit of the injector tank there is a retractable target for calorimetric measurement of the beam profile (movable beam dump). The monitor consists of a stack of nested copper rings, which can be installed co-axially with the beam, and a water-cooled tube to which the stack is welded from inside. There is also a thermocouple array, which is used to measure the temperature rise of the rings impinged by the beam particles. When not in use, the beam dump is normally stored in the radial cavity adjacent to the beam duct. A pivoting suspension driven by electric motor is used to traverse it between the storage location and the duct.

Other subsystems of the DNBI: Vacuum system with two cryogenic pumps; RF-system; some power supplies for grids, for bending magnet, driving electronics for Gas Valves and for Ignition. The systems are distributed at the DNBI Tank, at the High Voltage Tank and at six racks with dimensions close to Euromechanics standard.

3 CONTROL SYSTEM

3.1 Specific requirements

A number of specific requirements have been made for the control system of the DNBI during the developing stage. Some of them are:
1) The Control System should be highly resistant to the noise and interference, especially to arc breakdowns at the injector ion source. It should accept without errors and breaks the failures of the power subsystems. ***2) The Control system should have*** a number of protections and interlocks at the hardware level with obligatory computer control of their status. In addition, all the safety functions necessary for safe DNBI system operation should be included in the TCV control system to prevent system operation in the event of possible danger. ***3) The Control system should be*** as autonomous as possible. DNBI can be operated in Test mode and TCV mode with the ability of its control system to be able to safely distinguish these two modes [3]. ***4) No galvanic connections*** between the DNBI system and "external world" (TCV) is allowed. "Hot" cables and signals must be separated from "cold" cables and signals.

All of these specific requirements were satisfied during the development of the control system and DNBI as a whole.

3.2 Control System Hardware Specification and Configuration

The Control System provides control and measurement of Injector Equipment, Distribution unit, High Voltage Cabinet, Low Voltage Cabinet, Vacuum system, Water Cooling system and Gas supplying system. As a result of analysis of configuration of the DNBI subsystems and of the necessity to check their status or to look for their signals before, during and after "shot" we had found that the control system should have the following computer controlled functions (Table 1):

Table 1: The set of computer-controlled channels.

Function	Number of channels, up to
ON/OFF setting	32
ON/OFF checking	32
DAC	6
Slow ADC	35
Fast ADC	12
Timing	14

Of course, these binary status-channels, analog channels for measurement and for control, belongs to the different subsystems of the DNBI. They are placed in different racks, different rooms and halls and, as a result, they have different potentials relative to the "ground", especially due to high voltage breakdowns. From the other side, it is desirable to use multi-channel DAC, ADC, IN/OUT digital Registers and Timer for these kinds of systems. In the case of TCV we applied the CAMAC standard for the control Modules. Connections of all required analogous input/output and digital (status) signals between CAMAC and DNBI subsystems are implemented at the Low Voltage cabinet through a special Cross-Panel Module and system of low-pass filter Modules: Thermocouple Filters Module, Injector Equipment Filters Module and Modulator Interface Module.

The Cross-Panel Module has the function to connect DNBI subsystems via Filters directly with CAMAC Modules. This solution allows us to have minimal noise influence. List of CAMAC Modules of the DNBI Control system includes the IN/OUT registers, 16-channel DAC (0.01%), "Slow" ADC with Analog multiplexers for 64 channel, three "Fast" ADC with 4 channels each and 16-channel Computer Controlled Timer.

All the IN/OUT Digital Signals are organized with TTL compatible current sources. The precise "Slow" ADC is applied for the system because it is a double-integrating device, practically insensitive to the oscillating short-term noise and interference. The "Fast" ADC with programmable sampling period is applied to organize the oscilloscopic mode of DNBI subsystems observing. We can check up to 12 channels with this ADC and simultaneously view up to four of them at the PC monitor.

4 EXPERIMENTAL RESULTS

The DNBI has been designed, tested and successfully installed at the TCV tokamak at the Plasma Physics Institute two years ago [4]. The DNBI control system is operated from an IBM PC compatible computer. It works under Windows 95 using the Java 1.1x environment. A PPI adapter card connects the Crate Controller to the PC. The system accepts all shot-to-shot operation commands from the TCV Control System and follows the clear and simple specification of the "Start of Day" and "End of Day" procedures for safe and reliable DNBI operation. A typical screenshot of DNBI test results displaying status and oscilloscopic information from the HVM is presented in Fig.1.

Fig.1 Typical operation of DNBI subsystem during test "shot"

REFERENCES

[1] E.Hintz and B.Schweer, Plasma Phys. Control. Fusion 37 (1995) A87-A101.

[2] V. Kolmogorov, A. Bulatov, Yu. Evtushenko, A. Medvedko, "High Voltage Modulator with small energy accumulation in the output filter". Proc. of the 7th European Particle Accelerator Conference (EPAC 2000), June, 2000, Vienna, Austria.

[3] J. Mlynar, A.N. Shukaev, P. Bosshard et al., "Diagnostic Neutral Beam Injector at the TCV Tokamak", LRP 710/01, CRPP EPFL, October 2001

[4] J. Mlynar, B. Duval, et al., EPS Plasma Phys. And Controlled Fusion Conference, Madeira, Portugal, 2001 (to be published).

TUAP018
cs.DB/0111018

DATA ACQUISITION AND DATABASE MANAGEMENT SYSTEM FOR SAMSUNG SUPERCONDUCTOR TEST FACILITY

Y. Chu, S. Baek, H. Yonekawa, A. Chertovskikh, M. Kim, J. S. Kim, K. Park, S. Baang, Y. Chang, J. H. Kim, S. Lee, B. Lim, W. Chung, H. Park, K. Kim
Samsung Advanced Institute of Technology, Taejon, Korea

Abstract

In order to fulfill the test requirement of KSTAR (Korea Superconducting Tokamak Advanced Research) superconducting magnet system, a large scale superconducting magnet and conductor test facility, SSTF (Samsung Superconductor Test Facility), has been constructed at Samsung Advanced Institute of Technology. The computer system for SSTF DAC (Data Acquisition and Control) is based on UNIX system and VxWorks is used for the real-time OS of the VME system. EPICS (Experimental Physics and Industrial Control System) is used for the communication between IOC server and client. A database program has been developed for the efficient management of measured data and a Linux workstation with PENTIUM-4 CPU is used for the database server. In this paper, the current status of SSTF DAC system, the database management system and recent test results are presented.

1 INTRODUCTION

The KSTAR device is a tokamak with a fully superconducting magnet system, which enables an advanced quasi-steady-state operation. The major radius of the tokamak is 1.8 m and the minor radius is 0.5 m with the elongation of 2. The superconducting magnet system consists of 16 TF (Toroidal Field) coils and 14 PF (Poloidal Field) coils. The arrangement of the KSTAR coil system is shown in Fig.1.

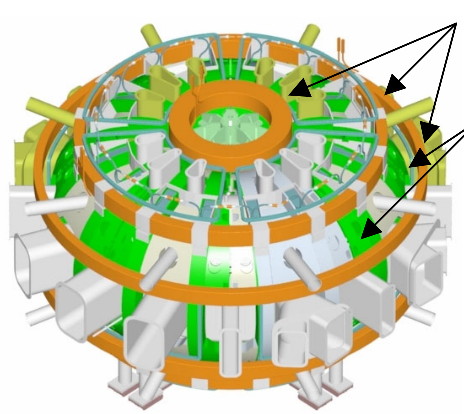


Fig.1 Arrangement of the KSTAR Magnet System

SSTF is constructed in order to test superconducting strands, conductors, and magnets. It consists of a vacuum cryostat system, a cryogenic cooling system, a background magnetic field generation system, a power supply system and a data acquisition and control system. Large Vacuum Cryostat (LVC) with diameter 6 m and height 7.9 m is located in a pit of 9 m x 9 m x 6 m. It is connected to three cold boxes, Helium path control valve box (CB#1), Helium flow rate control box (CB#2), and current lead box (CB#3), respectively. The cryogenic cooling system has two 200-watt Helium refrigerators and one 1000-watt Helium refrigerator, which is under construction. The power supply system also has a superconducting transformer type power supply, which is able to provide 50 kA to a superconducting CICC (Cable-In-Conduit Conductor) sample. The operation scenario of KSTAR device requires a steady state operation of TF coils and a fast ramping of PF coils. Thus, the test of KSTAR superconducting magnet and CICC also need to be performed under the condition of the fast varying magnetic field. The background magnetic field generation system is designed to provide a changing magnetic field of a 3 Tesla/second for 5 seconds and a 20 Tesla/second for 0.05 seconds. Therefore the data acquisition system has two modes, a high-speed data logging and a low-speed monitoring. The low-speed monitoring system is used to control slow-varying parameters such as Helium flow rate, temperature, and pressure, etc. However, a quench protection system, which is linked with the power supply system, and a magnetic measurement system require a high-speed data acquisition.

The computer system for SSTF DAC is based on UNIX and Linux system and VxWorks and RTEMS are used for the real-time OS of the VME system. The EPICS is the basic communication software for the monitoring of slow-varying parameters and RT-Linux system with a PCI-VME interface is also used for the high-speed data acquisition system. A Linux workstation with PENTIUM 4 CPU is the database server for the in-house developed database system. In this paper, the current status of SSTF DAC system, the database management system and recent test results are presented.

2 SSTF DAC SYSTEM

2.1 System Overview

Figure 2 shows a schematic diagram of SSTF DAC system. The basic configuration consists of operator interfaces, I/O controllers and Ethernet link. The operator interfaces are both UNIX workstations and Linux-based PCs. VME controller boards are based on Motorola 68040 CPU. VxWorks and RTEMS are used for the real-time OS of VME controllers. RT-Linux is the OS for the host computer of the high-speed data acquisition system, where SBS Model 620 is used for the PCI-VME interface and PENTEK 4275 is the main A/D converter. The data from 64 channels can be handled with 100 kHz sampling rate. Some independent devices such as strain gauge controllers, GPIB devices, vacuum furnaces and Helium liquefiers do not have an Ethernet interface. The data from such devices are collected to PCs and stored to an in-house developed database system using a NFS (network file system). Then, the data can be monitored using an in-house developed quasi-real-time monitoring system.

PENTEK 4270 DSP boards and PENTEK 4275 A/D converters with 4202 MIX baseboards are used for the quench detection and protection of superconducting magnet. Slow-varying parameters such as temperature, pressure, and Helium flow rate are monitored using VMIC VMIVME3122 A/D scanners. Various I/O modules such as VMIVME 4100 D/A Converter, PENTEK 1420 clock, and NIGPIB 1014 board are also installed in VME crates. Sets of PLC systems are used for the control of pneumatic valves, Helium refrigerator, vacuum furnace and etc. The National Instrument GPIB-ENET is also used for the control of GPIB devices such as a signal analyzer, a digital oscilloscope, voltage source, and etc.

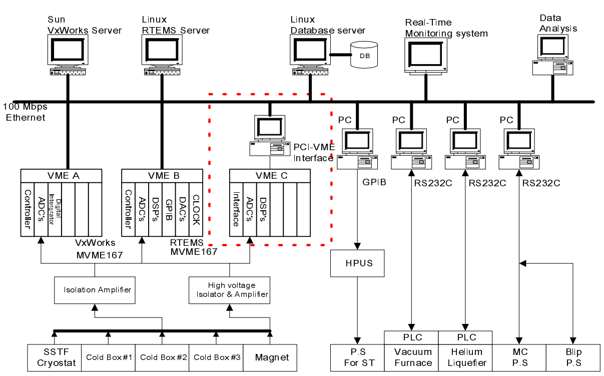


Fig.2 Current status of SSTF DAC system

Most of signals from various sensors are conditioned using isolation amplifiers. Isolation amplifiers have low pass filters and selectable gain controls by resistor change. A voltage tap signal could have a high voltage and high voltage isolation amplifiers are used to protect the data acquisition system with the maximum isolation voltage of 20 kV.

2.2 EPICS core Software

The main purpose of EPICS is to provide a fast, easy interface to data acquisition and control, and to provide an operator interface to all control system parameters.

At present, EPICS is being used as the data acquisition and control software only in VMIVME 3122 and VMIVME 4100 modules. The EPICS IOC database with respect to these I/O modules has been made using Capfast, which is a commercial schematic editor and compiled to be loaded into IOC via VxWorks startup script file. Each I/O controller provides a channel access server and both operator interfaces and I/O controllers are available as channel access clients. Using the client/server model and TCP/IP, channel access provides network-transparent access to the IOC database.

Most of parameters in the SSTF cryogenic system change slowly and the duration of the operation is the order of month. The EPICS is a convenient software tool for such an application and is used to control slow-varying parameters such as Helium flow rate, temperature, and pressure, etc.

2.3 Quench Detection and Protection

Quench is the phenomenon that a superconducting material is changing from a superconducting state to a normal conducting state. During the quench of a superconducting magnet, the magnet could be damaged both electrically and mechanically. In order to protect a magnet during quench, the data that is required for the quench detection is sampled with high speed during the operation of the magnet.

For example, the sampling rate of voltage tap signals is 100 kHz. The data is then transmitted via local bus to the DSP board. According to the quench detection algorithm provided, DSP board analyzes the status of the superconducting magnet. In case of quench, DSP board generates a trigger signal for the quench protection system and the magnetic energy stored in the magnet is dissipated to the energy dump circuit.

3 DATABASE MANAGEMENT SYSTEM

The Linux OS with PENTIUM 4 CPU with 800 MHz Rambus DRAM is adopted as a database server because of its faster memory access time. The data from various sensors, GPIB instruments, vacuum gauge and power supply is stored with a given database format. The total number of data points is estimated to reach 5000 points in full-scale experiments.

The data has its own device name and data name. The data is stored in a separate file with binary format. Each file has three columns of double type data, which generally represent time index, raw measured values, and calibrated values, respectively. A certain data such as the voltage tap signal has the same raw values with calibrated values. The data from a device such as signal analyzer have the different data style and (time index, frequency, amplitude) and (time index, frequency, phase shift) will be stored in two separate file. Eventually, all the data during experiment can be easily accessed without modifying the application software for data retrieval and archival. Users merely have to know the data name and recorded date/time to retrieve the favorite data from the database. Also, users can download their data both in text and binary file format using another software. The text format is useful to perform analysis with the combination of graph applications running on PC platform. Besides, in order for users to monitor their equipment data more easily, two GUI programs are developed through x-window libraries. The application program, XY_MON, is a real-time data monitoring software using EPICS CA library and another is XYP which shows a graph after obtaining data through the database server access. XYP can also used for the quasi-real-time monitoring of the data through the continuous access of the database.

4 TESTS AND RESULT

Figure 3 shows the SSTF thermal shield cool-down history. XYP and XY_MON also generate the graphics output in postscript format, which is shown in Figure 3. The raw voltage data from the thermal shield of SSTF LVC is converted to the physical temperature data by built-in calibration routines in the database software, which also accesses the calibration data of various sensors.

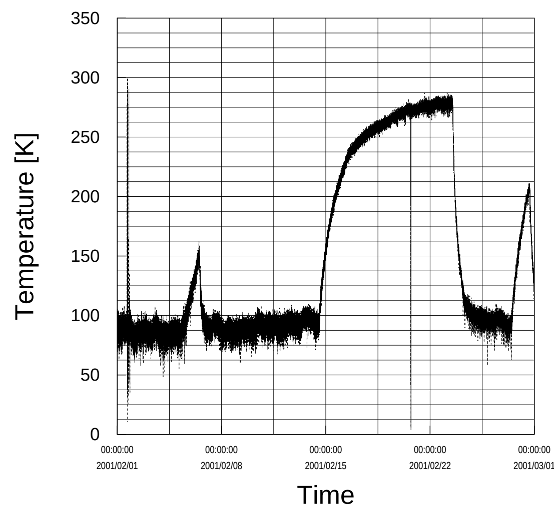


Fig.3 Thermal Shield Temperature of LVC

5 CONCLUSIONS

The database system with EPICS, which plays an essential role in the control system of SSTF, proved its reliability and scalability. For the full test of superconducting magnets under various conditions, the quench detection and protection system using DSP is still under development for more reliable quench detection algorithm. The data from the high-speed data logging system, which also include the data for the quench detection, is also stored to the database server with the same file format through DMA access between PCI adapter and VME adapter. RT Linux is currently adopted as basic OS for a real-time tasking in this system.

REFERENCES

[1] Y. Chung et al, ‘Development of GUS for Control Application at the Advanced Photon Source’, Proc. EPAC94 (London, 27 June – 1 July 1994) pp.1794-1796

[2] H. Choi et al, ‘Data Acquisition and Database Management System for Samsung Superconducting Test Facility’, Proc. ICALEPCS (Trieste, 4-8 Oct. 1999)

[3] Joel Schultz, ‘KSTAR Design Description Document’, KSTAR-USMIT-JHSchultz-82599-01, August 23, 1999

[4] S. Baang et al, ‘KSTAR Magnet System Review WBS135’, Samsung Advanced Institute of Technology, August 25, 1999

TUAP019
physics/0111208

INSERTION DEVICE CONTROLS AT THE SWISS LIGHT SOURCE

T.Korhonen, B.Kalantari
Paul Scherrer Institute, Villigen, Switzerland

Abstract

In the initial phase of the Swiss Light Source (SLS), four insertion devices (ID) will be installed for the first four beamlines. The control system for all the ID's follows, where possible, a uniform concept to allow a rapid installation schedule, while at the same time allowing for a variety in operational requirements. The components of the control system include the gap drive system with encoders, temperature monitoring, beam position monitoring, power supply controllers for corrector magnets [1] and a timing module [2]. The gap drive system requirements range from micron-level accuracy to driving double undulators in synchronism with each other and also with the other beamline elements. The local synchronism is achieved with a tightly coupled system having an intelligent motor controller and the global synchronism by extending the machine timing distribution to the insertion devices and the beamlines with the capability to add beamline-specific timing events.

1 INTRODUCTION

At the SLS, the first insertion devices are a wiggler for the material science beamline, an in-vacuum minigap undulator for protein crystallography, an elliptically polarizing undulator (two identical devices) and a long electromagnetic undulator (two devices.) The core of the control system is specific for each device, but they also have a number of components in common.

The first device installed was an in-vacuum undulator (U24) developed in collaboration with Spring-8 in Japan. Its gap drive has a single stepper motor and an absolute rotary encoder. These are controlled through a local controller that has a GPIB bus interface. To integrate it with our control system, we wrote EPICS device support code for that controller in order to poll the status and send commands to drive the gap to a desired position. The common part of the control system and the user interface, however, was developed for this device first and adapted for the others.

For the two other permanent magnet devices, namely the wiggler W61 and a double undulator UE56, the gap controller had to be developed in-house. The aim was to design a system that would serve as a basis for the control system of future insertion device development.

The wiggler W61 has two motors to control the gap. It was decided to have two motors to be able to control the parallelism (taper) of the magnet arrays.

The UE56 is an undulator capable of producing linearly and elliptically polarized light. The switching of the polarization mode is achieved by splitting the upper and lower magnet arrays into two halves and having a mechanism move them relative to each other. There are a total of four motors: two for the gap and two for controlling the shift arrays. Two identical IDs will be installed for the beamline and these which have to be controlled synchronously, giving us 8 axes to control.

The fourth device is an electromagnetic double undulator UE212. This ID has no moving parts, but the fields are generated by electromagnets. For power supplies for this device we use the standard power supplies of the SLS machine [1]. Very little additional development was necessary, except for the operation of the device in several modes defined for it. These could be implemented as additions on top of the generic SLS power supply control software.

2 GAP CONTROL

Gap control is the centerpiece of an insertion device control system. For the W61 and UE56, the gap control system had to be developed in-house. Although the devices are quite different, we strived to find a common design that could also be used as a basis for future developments.

The gap control consists of a drive system to move the magnet arrays and a position measurement system to monitor the distance (gap) between them. The positions are typically measured by encoders. The philosophy was to have a gap measuring device as close as possible to the real mechanical gap. To have a high precision over a quite large range, we decided to use linear incremental encoders manufactured by Heidenhain [4]. The selected encoders (ULS300) have high precision absolute reference marks. This allows high reproducibility of the gap setting.

For the motor controller, we selected the Oregon Microsystems OMS58 motor controller card [7]. This card has software support from the EPICS community and is already in wide use at SLS beamline controls. One big advantage is also that this card has support for both servo and stepper motors, with the same software interface, so a large part of the software for both the servo and stepper motor-based systems could be reused.

The devices had a number of different requirements: the ability to drive to a fixed setting with open loop control only and real time position hold feedback. In all cases, several motors have to be moved synchronously.

For the development of the gap drive we decided to build an “ID test stand”, a mechanical model of an ID with which we could test and debug the control system before having the actual devices. The test system included only the gap drive and the linear encoders; it turned out to be an extremely valuable development tool, especially because the time between ID delivery to PSI and installation was short and the time for control system development very limited. With the test stand the control could be developed without having the actual device. Without the test stand it would have been almost impossible to meet the construction schedule.

For the W61, the gap accuracy requirement is not that strict and the movement is only in the gap direction, so it was sufficient to have stepper motors and an open loop control with capability to monitor the linear encoders and, if necessary, do a position correction. The system has two motors that have to run synchronously to keep the taper, i.e., the inclination angle of the magnet arrays, close to zero. To have a protection in the low level we built a controller with a PLC to act as an interface between the power drive system and the control system. The PLC monitors that both motors are running and that the taper does not get too large. It stops the drives if there is a problem. The foreseen operation mode for the wiggler is to be driven to a gap value and left there for a long period of data taking during the experiment. For this reason, there were no requirements of closed loop control; an open-loop control with stepper motors is adequate. The system can be controlled with high resolution by correcting the position with the help of the linear encoders.

The UE56 has more demanding requirements. The device has the ability to change the light polarization from linear to elliptical by shifting the upper and lower magnet arrays relative to each other. The large magnetic forces push mechanically the gap open when the shift arrays are moved. The difference is measured to be about 300 microns, which is not acceptable for operation. The system required closed loop position control, with four axes. The drive system uses AC servo motors. The conventional way would be to use rotary encoders on the motors for the position feedback, but to ascertain the required high precision, we decided to do the feedback directly from the linear encoders. This is more difficult to tune, because the whole system is included in the feedback loop and the components cannot be individually tuned. However, this simplifies the additional software because the regulation is directly based on the gap and no corrections for the effects of the mechanical elements in between (like backlash, bending, creep) are required.

Like for W61, a PLC system for low-level local control was developed. This time an external company implemented our specifications.

Two identical devices will be installed for the beamline. The reasons for having two devices are first to have a larger photon flux, and to have a fast switchable polarisation selection available. The switching is achieved by steering the beam horizontally a little apart through the undulators, putting them into different polarization modes and then having a mechanical beam chopper to select one of the two photon beams. In this mode, the two undulators have to be controlled as a single device.

We have achieved 1 micron precision and repeatability for the UE56 gap drive. The polarization mode can be changed dynamically and the feedback keeps the gap value constant regardless of the highly nonlinear magnetic forces from the magnet array shifts.

3 ADDITIONAL SYSTEMS

For the operation of the insertion devices, additional monitoring systems are necessary to guarantee the safety of the operation and to optimize the performance.

3.1 Beam Position Monitors (BPM)

For monitoring the beam position close to the insertion device, we use a separate beam position monitor system, consisting of pickups, a BPM processing module (Bergoz) [3] and a (SLS standard) ADC to record the positions. The main purpose of these BPMs is to provide signals to an external interlock system that could dump the beam if the beam orbit had a large offset or a large angle at the insertion device. These BPMs have proven very valuable to monitor the effect of the ID on the beam because they are situated close to the ID and give direct information about the beam orbit at that point. Further study is necessary to fully understand the behaviour and calibration of these BPMs.

3.2 Beam Loss Monitors

Beam loss monitors have been installed near the insertion devices. Especially critical is the U24 in-vacuum undulator. For that there are large area scintillators. The scintillators are very sensitive and have a fast response. They have been proven to be very useful for commissioning. The output from the beam loss monitors is pulses with the frequency giving the loss rate. The pulses are read out with a (SIS [6]) multichannel scaler into EPICS channels.

3.3 Temperature measurement

Monitoring of the vacuum chamber temperature is a safety measure against heating when the orbit is bad, or, as in the case of the in-vacuum undulator also serves as an indicator of possible problems in the cooling system. In this device, the control program constantly monitors the temperature and if it rises over a specified threshold, the gap is automatically opened. The temperatures are measured with thermocouples and Greenspring [5] thermocouple IP (Industry Pack) cards. For U24 we can also monitor the cooling system through a serial port interface.

3.4 Correctors

Each insertion device has horizontal and vertical corrector magnets placed close to the device upstream and downstream. These correctors are used to minimize the effect of the residual kicks from the ID at different settings. The corrector values for each gap setting follow breakpoint tables that have been measured during commissioning of each device. At the time of writing, the detailed implementation of these lookup tables for the UE56 is still partially open because there are two parameters: the gap and the polarization mode (magnet array shift.) Both of these have an effect of the orbit. We need two- (or perhaps even three-) dimensional lookup tables to fully implement the automatic correction.

3.5 Synchronization

In the future the insertion devices should be operated in synchronism with the other beamline components. In most cases this can be achieved simply with EPICS Channel Access. A method for tighter synchronization is foreseen with an extension of the machine event distribution system to the beamlines. Although the main interest of tight synchronization is in the experimental stations, the extension is done in the ID system by putting an event generator in the ID crate. In this way, additional events that are specific to each beamline can be generated by accessing the event generator. An example would be a sequence of movements that could be pre-loaded to the beamline control IOCs and synchronously triggered by sending the events from the ID control system crate.

4 THE SOFTWARE

The control system is implemented in EPICS, using mostly readily available driver and device support modules plus standard records. So far, all the functionality could be achieved without having to modify any of the low-level components (record types etc.), except, of course, writing device support for cards that do not have it already.

4.1 User Interface

Although the devices are very different, much effort has been taken to make them look similar to the operator and to hide the different levels of complexity behind a common operating interface. Basically, only a minimum of information is normally presented. The operator normally can only switch on or off the drives and to set the gap to the desired value. For instance, before turning on the power the devices go internally through a series of checks and settings (the gap setpoint is synchronised with the actual gap value, for instance) before the motor power is turned on.

At the time of writing the devices are still considered to be in commissioning and the user interface is rather rudimentary.

4.2 Integration to the beamline

The devices have been operated as independent units without any direct connection to the corresponding beamline. In the future, however, it will become increasingly important that the beamline components are aware of each other and can be controlled as one system. For this, it is required to agree on the method and rules of communication between the component control systems. The fact that all the beamline components are controlled with EPICS eases the integration.

The regular operation when a device is set to a certain gap/energy value is straightforward. The system will get more complicated when we want to do synchronised scans, i.e., operate the beamline monochromator in sync with the insertion device. Since the monochromator and the gap control are fairly slow devices, the synchronization is easily achieved with EPICS Channel Access over the network.

5 CONCLUSIONS

We have initially commissioned all the four types of planned insertion devices at SLS. Although work still remains to be done to finalize the operator interface and debug and calibrate the components, the concept has proven to work quite well and fulfills all the requirements. Two big factors that contributed to its successful development were the strong support and availability of software from the EPICS community, and the fact that we were able to standardize our control hardware to a large extent and could reuse the elements and we also had an extensive in-house expertise with the components. One big factor was having built a “test stand”, a mechanical model of an insertion device, with the ability to simulate (nonlinear) magnetic forces. Using the test stand we could develop the gap drive system before the devices were available for testing; this gave us almost one year of lead time before the delivery of the device. When the real device arrived, testing and commissioning could proceed smoothly without big surprises.

6 REFERENCES

[1] A.Luedeke et.al.,’Digital Power supplies for the Swiss Light Source’, ICALEPCS’01, San Jose, California, USA.

[2] T. Korhonen, M.Heiniger, ’Timing System of the Swiss Light Source’, ICALEPCS’01, San Jose, California, USA.

[3] Bergoz SA., “BPM system user’s Manual”

[4] Heidenhain GmBH. www.heidenhain.com

[5] SBS Greenspring, www.greenspring.com

[6] SIS GmBH. www.sis.de

[7] Oregon Microsystems, Co.. www.omsmotion.com

TUAP023
physics/0111039

THE SNS RUN PERMIT SYSTEM*

C. Sibley, SNS, Oak Ridge, TN 37831, USA
E. Bjorklund, LANSCE, Los Alamos, NM 87545, USA

Abstract

The Spallation Neutron Source (SNS) is an accelerator-based neutron source being built in Oak Ridge, Tennessee, by the U.S. Department of Energy. The SNS will provide the most intense pulsed neutron beams in the world for scientific research and industrial development. The facility is scheduled for completion in 2006. The project is a collaboration between Lawrence Berkeley National Lab, Argonne National Lab, Los Alamos National Lab, Oak Ridge National Lab, Brookhaven National Lab, and Jefferson Lab. The Run Permit System (RPS) is a software based system which has several key functions critical for safe operation of the machine. It coordinates machine mode and beam parameter changes with the timing system, verifies the Machine Protection System (MPS) hardware configuration, and provides an interface for masking MPS hardware inputs when necessary. This paper will describe the primary functionality of the Run Permit System and the interface between the Run Permit System, the Machine Protection System, and the timing system.

1 INTRODUCTION

The SNS Run Permit system is one level in the overall SNS Machine Protection System (MPS). The purpose of the MPS is to protect *equipment* from being damaged by the beam. It contains one software level (Run Permit), and three hardware levels (Fast Protect Auto-Reset, Fast Protect Latched, and High-QA MPS). More information on the SNS Machine Protection System can be found in [1,2].

The function of the Run Permit portion of the MPS is to 1) set up and verify the machine operating mode, 2) verify that all the relevant equipment is operating and within the desired setpoint ranges for the selected machine mode, 3) verify the equipment masks in the MPS hardware for each machine mode, 4) verify the beam parameters requested are within tolerance for the machine mode, 5) schedule user defined beam pulse parameters, and 6) provide an operator interface to display the status of the MPS and set software masks as appropriate.

2 RUN PERMIT MODES

2.1 Machine, Beam, and Operating Modes

The SNS "Machine Operating Mode" consists of two parts, a "Machine Mode" and a "Beam Mode." The Machine Mode determines where the beam stops (Linac Dump, Extraction Dump, Target, etc.). The Beam Mode determines the duration and power of the beam. Establishing a Machine Mode requires turning on equipment, verifying that beam dumps are ready, etc. before the desired Machine Mode can be made up. Consequently, the Machine Mode will not change very often under normal commissioning, tune-up, and operating conditions. The Beam Mode, on the other hand, can change on a pulse-to-pulse basis. Table 1 lists the Machine and Beam modes for SNS. The Machine Mode and maximum beam power or pulse width are selected in the control room via a key switch. The Beam Mode can be changed to smaller pulse widths from software applications, such as an emittance scan or profile measurement application.

Table 1: SNS Machine and Beam Modes

Machine Modes	Beam Modes
Ion Source	Diagnostics (10 µsec)
Diagnostic-Plate	Diagnostics (50 µsec)
Linac Dump	Diagnostics (100 µsec)
Injection Dump	Full Width (1 msec)
Ring	Low Power (7.5 KW)
Extraction Dump	Medium Power (200 KW)
Target	Full Power (2 MW)

The Operating Mode for the machine is the combination of Machine and Beam modes. Note that not every Beam Mode can be applied to a given Machine Mode. For example, when the machine is in "Linac Dump" mode, the Beam Mode may not exceed 7.5 KW, and only the "Target" Machine Mode can accommodate the "Full Power" Beam Mode. Table 2 lists the allowable Machine and Beam Mode combinations.

* Work supported by U.S. Department of Energy

Table 2: Valid Machine and Beam Mode Combinations

	10 µsec	50 µsec	100 µsec	1 msec	7.5 KW	200 KW	2 MW
Source	X	X	X	X			
D-Plate	X	X	X	X			
L-Dmp	X	X	X	X	X		
I-Dmp	X	X	X	X	X	X	
Ring	X	X	X		X		
E-Dmp	X	X	X		X		
Target	X	X	X	X	X	X	X

2.2 Diagnostic Modes

The "Diagnostic" Beam Modes exist to limit the integrated current absorbed by intercepting diagnostic devices such as wire-scanners or emittance harps. The pulse width limits are energy dependent. At lower beam energies damage to copper and diagnostic wires is greater so the beam pulse widths are limited to shorter values. This is accomplished by limiting the width of the beam gate.

Before an application program can insert a wire scanner (for example) into the beamline, it must first make a request to the RPS for an operating mode change. The RPS determines the required operating mode based on the device making the request, and the current machine mode. It then interfaces with the timing system to set new limits on the width of the beam gate. Once conditions have been met for the new operating mode, the RPS sets the new mode (this is also done through the timing system) and signals the application program that it may proceed with the wire scan. The operating mode is locked by the RPS until the application signals the wire scans are complete. Wire scanner "Home" signals are inputs into the MPS system. If a wire scan begins before the appropriate diagnostic mode is made up, the MPS will drop the beam. Once the wire scan is complete, the RPS returns the operating mode to its previous value.

3 RPS USER PULSE SCHEDULING

Various beam pulse profiles can be requested through applications or an EPICS user interface. Each active profile is given a "User_ID" for use in synchronous data acquisition and for feed forward lookup tables in the LLRF systems [3]. The User_ID is broadcast over a Real Time Data Link [4]. There is a limit of 7 active profiles due to memory constraints in the LLRF system. The integrated beam current in the requested profile is checked against pulse width limits for diagnostic machine modes. The integrated power is calculated for the requested repetition rate of the pulse (and other pulses in the request queue) and placed in the queue if the power limitations are within tolerance for the dump in use. Figure 1 shows the operating envelopes for various rep rates and diagnostic modes.

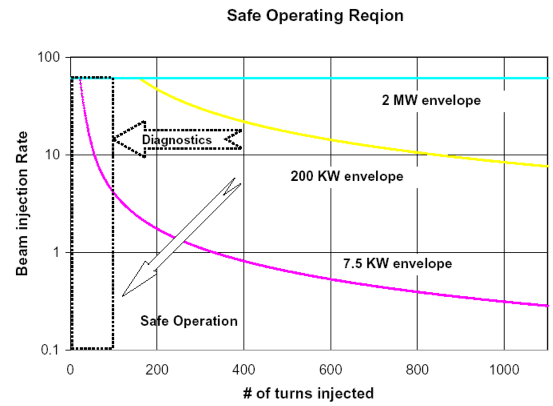


Figure 1. Operating envelopes

3.1 Sequencer

The Run Permit System builds the sequences of various pulse types and hands them to the timing system. Each sequence consists of a "Supercycle" for the machine. This is the lowest continuous repetition rate of interest to operations. Single shot pulses are inserted as requested after pulse width verification. There is a separate sequence for each power level and diagnostic machine mode. The RPS only populates sequences that are allowed in a particular mode and where the mode can change on a pulse-to-pulse basis. For instance, the mode can always change down in power to a diagnostics mode, but can never change to a higher power level than set by operator key settings in the control room. During the operational phase of SNS, diagnostic pulses will be scheduled to calibrate and verify target system diagnostics and the injection phase space painting hardware. A part of the timing system, the Time Line Verification Board monitors the event link and records and verifies the events sent out. Discrepancies result in a temporary beam fault.

4 EQUIPMENT MASKING

4.1 MPS Mode Masking, Software Masks

The Operating Mode (Machine plus Beam Mode) determines what equipment must be operational and at the correct set points before beam is allowed. The MPS hardware contains a provision for "Mode Masking," which selectively enables or disables inputs into the MPS system based on the Operating Mode. The mode masks are originally derived from the SNS technical database that describes every piece of equipment and signal in the SNS [5]. The database records whether or not a piece of equipment has an input into the MPS system, and if so, under what Operating Modes the input is relevant. Two files of these masks are created, one for downloading to the IOCs during initialization and a second used by the Run Permit system for verification during running.

Some of the inputs to the MPS can be masked out in software as the need arises. Beam loss monitors for instance, are automatically masked out using software masks when an upstream wire is inserted in the beam. Faulty equipment not required for a particular running scenario can be masked out as long as the MPS hardware configuration allows it. This allows the Mode Masks to remain relatively stable and easily kept in configuration control.

4.2 Set Point Monitoring

During the commissioning stages of SNS, the set points of power supplies can be changed as required for various accelerator physics applications, such as beam-based alignment. This should not matter from the machine protection point of view as the average beam power will be small and the accelerator components can generally take two or more full beam pulses in the event of a failure.

In the operational stages the ability to change power supply set points will be restricted. Channel access security will be used to prevent inadvertent set point changes or accidentally powering down a supply. There are other failures that can occur: power supply regulation loops failing, zero flux current transducers failing, or DAC failures in the power supply controller. An EPICS State Notation Language program running on the IOCs will be monitoring the equipment to detect these types of failures. Although there are other systems designed to detect these failures (primarily the Beam Loss Monitor System), this adds to the defense in depth character of the Machine Protection System.

5 OPERATOR INTERFACE

The final function of the RPS is to provide an operator interface to the SNS MPS system. Using EPICS RPS screens, the operator can view the state of the MPS system, see which sections are ready to take beam, which sections have MPS faults, and which MPS inputs have been bypassed. A fault timeline is provided to determine where a fault originated. The fault timers use an "Event Link" clock (approximately 16 MHz) to record fault times. These counters are converted to a time stamp in EPICS using revolution frequencies distributed in the Real Time Data Link. In some cases (where allowed by the hardware configuration) an MPS input may be bypassed in software using the RPS operator interface screens. Equipment masking is logged in the EPICS Archiver and acknowledged during shift changes, or as required by operations.

6 CONCLUSIONS – CURRENT STATUS

The MPS is designed for Defense in Depth and the RPS is the software layer of that defense. Hardware is used wherever a system failure will cause a loss of beam. Software is used for equipment status verification, including the configuration of the MPS.

The RPS will be installed in phases following the installation of the beamline equipment. The present installation schedule is shown in Table 3.

Table 3. Global Controls (GC) Readiness dates.

Activity	Finish
GC Ready for FE commissioning	9/20/02
GC Ready for DTL commissioning	12/4/02
GC Ready for CCL commissioning	4/27/04
GC Ready for SCL commissioning	8/19/04
GC Ready for HEBT-Ring comm..	11/8/04
GC Ready for RTBT-TGT comm.	11/15/05
GC Subtask Complete	11/30/05

Although the commissioning and installation schedule shows completion in 2005, the RPS needs basic functionality, pulse scheduling, operator interface, and user defined pulse shapes to be operational for the front end commissioning in 2002.

REFERENCES

[1] C.Sibley, "SNS Machine Protection System VME/PMC Module Design", Internal documentation:

[2] C.Sibley, "SNS System Requirements Document for Machine Protection System", Internal documentation:

[3] A. Regan, "System Description of the RF Control System for the Spallation Neutron Source", Internal documentation

[4] B. Oerter, "Timing System Requirements Document", Internal documentation:

[5] K. Kasemir, "EPICS DB and Oracle", Internal documentation:

TUAP025
physics/0111038

FEL INJECTOR CONTROL SYSTEM ON THE BASE OF EPICS

T.V. Salikova, A.A. Kondakov, G.Ya. Kurkin, A.D. Oreshkov, M.A. Scheglov, A.G. Tribendis
Budker Institute of Nuclear Physics, 630090, Novosibirsk, RUSSIA

Abstract

The control system of the 1.5 MeV FEL injector is built on the base of ported EPICS. It uses low-cost hardware: personal computers with the processor Intel x86 and CAMAC equipment produced by our institute. At present time, the distributed control system includes one Pentium at OPerator Interface (OPI) level and two IOC (Input Output Controllers) under supervision of the real time operating system LynxOS/x86 at the low-level. Each IOC is used for monitoring of autonomous parts of the injector. The first IOC operates the Radio Frequency (RF) system. The second IOC operates the injector equipment.

STATUS OF FEL

The first stage of the FEL (Free Electron Laser) complex [1] consists of a 1.5 MeV injector, one-track microtron-recuperator with accelerating RF system, and submillimeter FEL (Fig.1). The particle energy is 14 MeV. The bunch repetition rate is 0.022 to 22.5 MHz, and the average current is 10 to 50 mA. The FEL produces radiation of 1 to 10 kW average power and wavelength of 100 to 200 µm. The pulse duration is 20 to 100 psec. The first stage of the FEL is to test recuperation of the beam power and to obtain high power terahertz radiation.

At the present time, the FEL injector and RF system are installed, and installation of the microtron equipment is in progress.

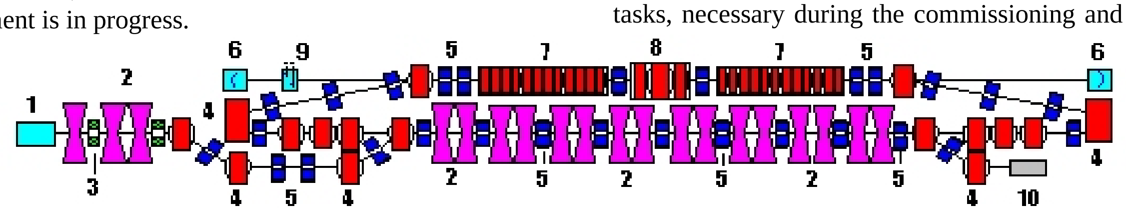


Fig.1. First stage of FEL.
1–electron gun; 2–RF cavities; 3–solenoids; 4–bending magnets; 5–quadropole lenses; 6–FEL optical resonator mirrors; 7–undulators; 8–buncher; 9–outcoupler; 10–beam dump

The RF system contains one bunching cavity, two accelerating cavities fed by three generators of the injector, and 16 accelerating cavities fed by two generators of the microtron. The cavity has two contactless tuners of the fundamental-mode frequency, which is 180.4 MHz. The fundamental mode tuning range is 320 kHz. Two Higher-Order Modes (HOM) tuners detune the HOM frequencies while having almost no influence on the fundamental mode. The IOC of the RF system controls the operation of all cavities and RF generators. It also can run OPI functions for adjustment of the equipment. The RF system database includes 481 records.

The injector IOC operates the electrostatic gun, which emits electron bunches of 0.1 to 1.5 *n*sec duration with 0.022 to 22.5 MHz repetition rate. Power of a bunch reaches 300kV at the exit from the gun. Further, the electron bunch is accelerated in the RF cavities up to 1.5MeV. The injector IOC controls the injector magnetic system, which consists of solenoid lenses, correctors, pick-ups and beam transducers. The equipment for beam diagnostics is connected to the injector IOC. The injector system database includes upward of 150 records.

FEL INJECTOR AUTOMATION

By now we have updated the software of the injector control system, which had been developed on the base of Windows95 with the LabWindows/CVI tool kit. Two variants of software for the RF control system have been developed: the first variant uses LabWindows/CVI, and the second variant is built on the base of EPICS. The RF system control program under Windows95 executes a range of technological tasks, necessary during the commissioning and service of the equipment, and all functions of the control system. The EPICS variant of software executes only functions of the control system and is designed for the operation of the RF system as a part of the FEL complex.

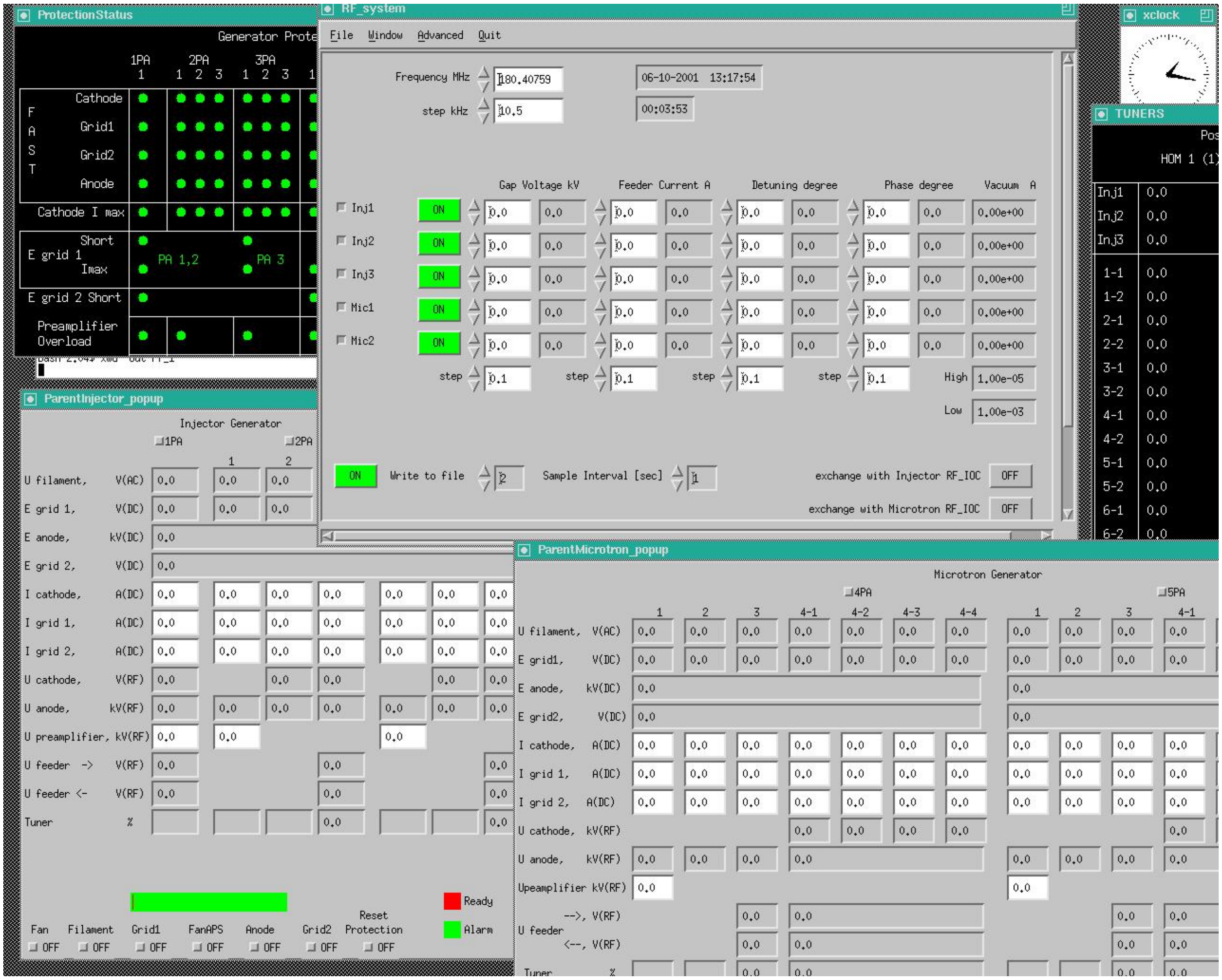


Fig.2 Main control panel of RF system.

The FEL control system is built on personal computers with Intel x86 processors under supervision of the real time operating system LynxOS, and CAMAC devices produced by our institute, which reduces the automation cost. The FEL control system uses 14 record types only and 20 device support routines, which were written for optimization of data exchange with the equipment. All exchanges with the devices are executed by a standard mechanism, a driver supporting synchronous and asynchronous requests.

The CAMAC bus has low performance: the time of full execution of NAF reaches 16 μsec. Thus, control of emergency conditions is executed by the hardware. That is the history-formed approach of our institute to solution of hard real time problems via creation of special hardware. A set of such devices was created for the FEL control system: devices for control of beam propagation in the injector, a gun driver, devices for control of locks of doors of the accelerator hall, a driver for monitoring of temperatures, etc.

At the OPI level, the RF control program (Fig.2) has seven panels for monitoring of the generator parameters of the injector RF system and microtron RF system as well as control panels for monitoring of three injector cavities and sixteen microtron cavities

The OPI program includes a panel with automation symbols of the injector, where each element is linked to its own control panel. A click to a mnemonic element generates a subwindow containing all control information.

The user has a set of windows made as a set of manipulation panels with a panel with automation symbols. A color palette indicates the alarm status. If a device fails, then the corresponding element in the panel with automation symbols is painted red. If a device works without errors, then its color is green.

The OPI programs of the RF system, gun and magnetic system log all updates of physical parameters to a cyclical buffer, which is periodically written to a file, which helps in analysis of failures of equipment.

CONCLUSIONS

The FEL control system uses low-cost hardware made by our institute for specific tasks of control systems of particle accelerators. EPICS guarantees a scalable system, which is important for development of the FEL control system. Performance of the EPICS tool kit meets the requirements of hard real time control.

REFERENCES

[1] N.A. Vinokurov, "Status of Novosibirsk high power free electron laser", Proceedings of AFEL-99, June 8-10 1999, Korea Atomic Energy Research Institute, Taejon, Korea pp 7-12.

TUAP026
physics/0111097

STATUS OF THE CONTROL SYSTEM FOR THE FRONT-END OF THE SPALLATION NEUTRON SOURCE {*}

S. A. Lewis, C. A. Lionberger,
P. T. Cull, LBNL, Berkeley, CA 94720, USA

Abstract

The Spallation Neutron Source (SNS) is a partnership between six laboratories. To ensure a truly integrated control system, many standards have been agreed upon, including the use of EPICS as the basic toolkit. However, unique within the partnership is the requirement for Lawrence Berkeley National Lab, responsible for constructing the Front End, to operate it locally before shipping it to the Oak Ridge National Lab (ORNL) site. Thus, its control system must be finished in 2001, well before the SNS completion date of 2006. Consequently many decisions regarding interface hardware, operator screen layout, equipment types, and so forth had to be made before the other five partners had completed their designs. In some cases the Front-End has defined a standard by default; in others an upgrade to a new standard is anticipated by ORNL later. Nearly all Front-End devices have been commissioned with the EPICS control system. Of the approximately 1500 signals required, about 60% are now under daily operational use. The control system is based on "standard architecture"; however, it has a field-bus dominated layout. This paper will discuss some unique interface requirements that led to adding new device families into the EPICS repertoire. It will also describe the choices and trade-offs made for all major areas.

1 OVERVIEW

The Spallation Neutron Source (SNS) is a 1 MW, pulsed neutron source being built by six Department of Energy Labs and scheduled for completion at Oak Ridge National Lab (ORNL) in 2006. At completion it will cost about $1.4B and extend over nearly 1 km. Lawrence Berkeley National Laboratory (LBNL) is tasked with constructing the Front End (FE) at Berkeley by June, 2002 and then shipping it to ORNL for re-commissioning. The FE itself is comprised of three sections: the Source and Low Energy Beam Transport (LEBT) which produces 50 mA of 65 keV H^- suitably chopped and focussed (using electrostatic devices); the Radio Frequency Quadrupole (RFQ) which bunches the beam with 800 kW of 402 MHz RF and further accelerates it to 2.5 MeV; and the Medium Energy Beam Transport, which matches the beam to the Linac (using a magnetic lattice) and includes diagnostic devices. All three sections are served by suitable vacuum pumping, employing turbo-molecular, cryogenic, and getter-ion types.

Thus it can be seen that although the FE represents roughly 3% of the full SNS measured by beam energy (2.5 MeV), length (10 m), or cost ($20M), it has nearly the same complexity. For the FE control system, a similar ratio applies: it will have about 1550 signals (of about 60,000 for SNS), cost $1.5M (of about $60M); have one operator console (of 12), and three I/O controllers (of 150).

When complete, the 1550 signals in the FE control system will come from 300 devices, assigned by system: Source/LEBT, 250; RFQ, 100; MEBT, 700; Vacuum, 500.

To support the extensive R&D program and operation of the whole FE, the control system has been in continuous operation since 1999, with incremental additions to track the FE itself. As of 1 November, two of the three I/O controllers (IOCs) are in service, with about 700 signals (for 200 devices) interfaced, essentially completing the Source, LEBT, and RFQ systems. In addition to the required operator console (OPI), three temporary OPIs have been provided to allow concurrent but independent operation of the 402 MHz systems, vacuum commissioning, Source/LEBT testing, and a software development area. An EPICS "gateway" allows controls and management staff at SNS partner labs to continuously and efficiently monitor active devices, while critical engineering and technical staff routinely monitor and control devices in their jurisdiction from their offices or even homes.

2 FRONT-END AND SNS STANDARDS

The SNS has chosen the well-known EPICS toolkit [1] as the basis for its control system. While assuring easy integration among the multiple collaborators, this decision still leaves open many options. Because of the early start and immediate operational needs of the FE,

* Work Supported by the U.S. Department of Energy under Contract #DE-AC05-00OR22725

it did not adhere to all of the final SNS standards as developed by the collaborating partners [2].

2.1 Hardware

IOC. The EPICS Input/Output Controller (IOC) uses 21-slot Wiener™ or 7-slot Dawn™ VME64x or VXI crates with Motorola™ MVME-2100 series PowerPC CPUs. FE will conform to the SNS standard (although it has operated well to date with older MVME-167 68K series). SNS uses the VxWorks™ kernel.

PLC. For robust vacuum service, a layer of Programmable Logic Controller (PLC) is placed between the EPICS IOCs and the hardware interface. FE has chosen the Allen-Bradley™ (A-B) PLC/5 family, with IOC-to-PLC and PLC-to-I/O both using the proprietary A-B Remote-I/O (RIO; RS-232 based) and the 6008 VME Scanner; and the A-B 1794 (Flex-I/O™) interface modules. FE will not conform to the SNS standard [3], which stays with the A-B family, but uses the ControlLogix/5000™ family; Ethernet/IP for IOC-to-PLC and ControlNet™ for PLC-to-I/O, as well as CLX/5000 family interface modules.

FE exclusively uses self-contained vacuum gauge units. FE does not conform to SNS standards, which require rack-mounted electronics away from high radiation areas.

Power Supply I/O. For general electrostatic, magnetic, RF and other devices, FE has two alternate specifications (see [4]): (1) Group-3, a proprietary system utilizing a serial fiber ring; (2) A-B Flex-I/O directly driven from A-B 6008 Scanner with RIO. FE will not conform with the SNS standard [4], which uses a BNL-designed Power Supply Controller/Interface technology.

Motion Control. FE will use OMS-58 VME boards. FE will conform to SNS standards.

High-Power RF. The temporary interface now in use will be replaced by a LANL-supplied, VXI-based unit including full EPICS support, which is the SNS standard.

2.2 Software

Application Development Environment (ADE). The ADE is an enhancement of the EPICS application *makefiles* with a strong notion of release management (based heavily on experience at TJNAF with the CEBAF EPICS system) that recognizes the common usage of shared drivers across applications, and of shared applications among many IOCs. All textual file entities are version-controlled using CVS at a single repository at the SNS ORNL site. FE conforms to the SNS standard.

Operator Screens (HMI). EPICS offers several “display manager” client applications. FE conforms to the initial SNS standard, *dm2k*. However, a recent change to *edm* will require an upgrade. FE conforms to use of all other EPICS HMI-level clients such as *Knob Manager*, *StripTool*, *Channel Archiver*, *Backup Restore Tool*, *Probe*, *etc*. FE initiated a suite of colors, fonts, navigation strategy, layout and other HMI visual factors which have become the SNS standard.

Sequencing. The EPICS sequencer (v 1.9.5) is used for all applications which are well modelled as a Finite State Machine. In particular, it is used for high-powered RF conditioning; for RFQ temperature-frequency controls; for remote oscilloscope controls; and is under development for vacuum sequencing. FE conforms to the SNS standard.

Naming. A naming system was adopted by a working group for controls well before FE construction started. FE conforms to the SNS standard for systems, device, and signals (some late revisions will be addressed as an upgrade).

Relational Database. FE uses structured text files with extensive macro-substitution scripts to expand to the full EPICS configuration from primary files. FE does not conform to the SNS standard Oracle relational database to generate the configuration nor to hold the primary data in tables. Until an upgrade is performed, no mechanism exists to ensure compliance with the Naming standard.

3 FIELD-BUS DOMINATED LAYOUT

The FE control system at the level of block diagram is a conventional, “standard architecture” as used in both general and EPICS contexts; however, as the choice of either Group-3 or Flex-I/O for all device interfaces became pervasive, it became clear that overall this would be a field-bus dominated layout. That is, no analog or digital signals are directly wired to or from the VME modules (except OMS-58 motor controller cables); rather, all device signals are distributed over robust, serial, “daisy-chain” media via a protocol.

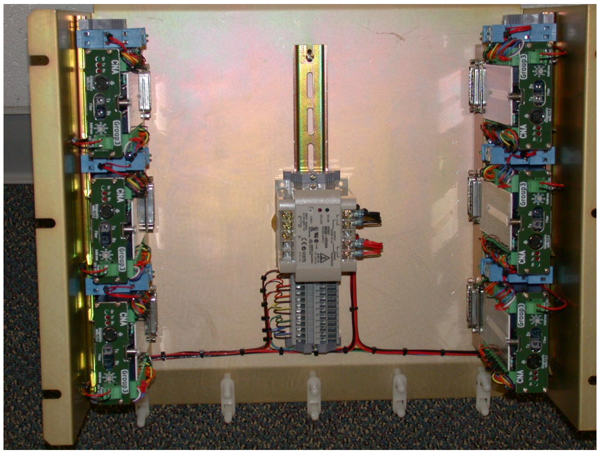


Fig 1: Rear rack U-chassis for Group-3.

Thus, there are no cross-connects, no requirement for low-level cable isolation in wire-ways, and no congestion as 700-800 signals converge in the racks housing the VME crates. Instead, outboard units were fabricated, either as rack-drawers immediately adjacent to rack-mounted equipment; as U-shaped chassis mounted on the rear rack rails (Fig 1); or as enclosed beam-line mounted boxes (Fig 2). All three types were heavily re-used, requiring only different wire-lists; and they all are fully connectorized to allow easy servicing.

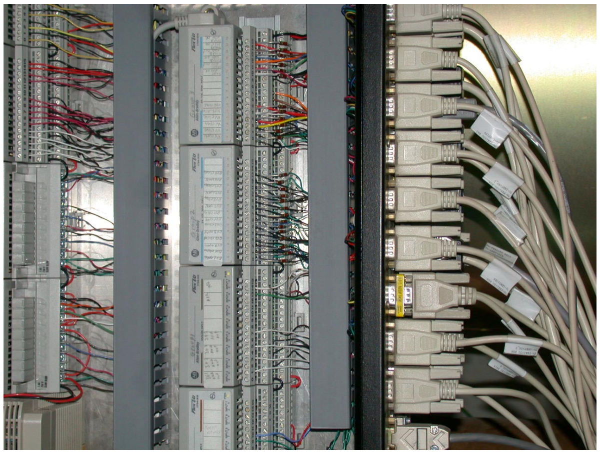


Fig 2: Enclosure for Flex-I/O with connectors.

4 NEW DEVICE TYPES FOR EPICS

4.1 Group- 3

The Group-3 technology was chosen for its easy isolation of high potentials found in the Source and LEBT areas as well as for noise immunity from the severe transients expected due to sparking. The isolation is a function of the fiber which connects the individual units on the proprietary serial field bus; the noise immunity is improved by the liberal inclusion of typical bypass circuitry. As advised by the vendor, all due care is still required in external circuitry, grounding, and shielding practices.

The Group-3 technology is also used where the specific interface modules are particularly well matched to the device. The CNA unit has a combination of high-resolution analog with filtering combined with digital signal handling which gives a one-to-one match with most LEBT and MEBT DC power supplies. The more configurable CN3 unit is used for motor/encoder requirements or cases with unusual input/output combinations. Driver software for this new EPICS device family was added by FE during the R&D phase, and includes very detailed, online, concurrent device diagnostics (Fig 3).

4.2 Allen-Bradley Flex-I/O

Flex-I/O is a moderate-cost, lower-resolution option for industrial type signals (RFQ and MEBT cooling) that can share the A-B 6008 scanner and RIO cabling with the PLC. Although EPICS has long had a rich repertoire of drivers for it, more detailed concurrent online diagnostics were added based on the favorable experience with Group-3, and drivers for several newer 1794 family items were added.

4.3 Tektronix™ Scopes

Both low- and high-level support was provided for the Tektronix TDS200 and TDS3000 2- and 4-trace scope families, using RS-232 and a Janz™ 8-port serial VME card for the former, and Ethernet for the latter. A near replica of the scope front panels is available to the operator. The serial line units update at about 1 trace/sec, whereas the Ethernet models can achieve about 25 traces/sec (500-points/trace).

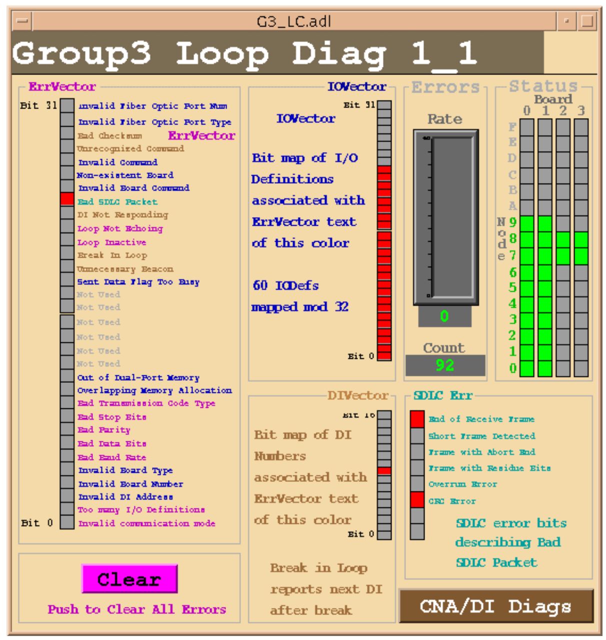


Fig 3: Diagnostics screen for Group-3 devices.

REFERENCES

[1] http://www.aps.anl.gov/epics; also http://lansce.lanl.gov/lansce8/Epics

[2] D.P. Gurd, W.R. DeVan, J. Hammonds, S.A. Lewis, J.D. Smith, "Plans for a Collaboratively Developed Distributed Control System for the Spallation Neutron Source", ICALEPCS'99, Trieste, October, 1999

[3] J.Y. Tang, L.A. Smart, H.C. Hseuh, W. Schneider, P.S. Marroquin, K. Kasemir, L.R. Dalesio, K. Kishiyama, D. Gurd, M. Hechler, "SNS Vacuum Instrumentation and Control", this conference.

[4] S. Peng, J. Smith, R. Lambiase, B. Oerter, "SNS Standard Power Supply Software", this conference.

TUAP027
physics/0111150

DIAMOND CONTROL SYSTEM OUTLINE DESIGN

M. T. Heron, A. J. Duggan, B. G. Martlew, A. Oates, P. H. Owens, V. M. Smithers, P. D. Woods
CLRC Daresbury Laboratory, Warrington, Cheshire, WA4 4AD, United Kingdom

Abstract

DIAMOND is a new synchrotron light source currently being designed in the UK. The control system for DIAMOND will be a site wide monitoring and control system for the accelerators, beamlines and conventional facilities. Initial work on the control system has selected EPICS as the basis for the control system design.

The requirements of the control system are presented. The technical solutions being considered to interface EPICS to the equipment being controlled are discussed together with the requirements for configuration and management of a large EPICS installation. Options being considered for the procurement, installation and commissioning of the control system are also presented.

1 INTRODUCTION

DIAMOND is a 3rd generation, 3GeV synchrotron light source currently being designed in the UK. The storage ring is based on a 24-cell double bend achromatic lattice of about 560m circumference. It uses a full energy booster synchrotron and linac for injection. The spectral output is optimised for high brightness up to 20keV from undulators and high flux up to 100keV from multipole wigglers. Initial construction includes seven photon beamlines.

The project is currently in a design and specification phase, with build and procurement starting in April 2002 and operation scheduled from Sept 2006. Details of the project status are described in [1].

2 CHOICE OF CONTROL SYSTEM TOOL KIT

Four options for the control system tool kit were reviewed during 1999: TACO/TANGO from the ESRF [2,3], commercial control systems, a development of the SRS control system [4] and EPICS [5]. The conclusion of this was that any one could deliver a working control system [6]. Following on from this a detailed analysis of control system tool kit requirements showed that EPICS offered advantages; in terms of its application to other accelerator projects, a larger base of developed drivers and performance/functionality.

3 CONTROL SYSTEM SCOPE

The defined scope for the control system is: that it will be a site wide monitoring and control system for the accelerators, beamlines and conventional facilities. The control system will extend from the interface of the equipment being controlled through to the operator. It will include all hardware and software between these bounds: computer systems, networking, hardware interfaces and programmable logic controllers.

The control system will not include any control or data acquisition for the experimental stations. It further will not include the personnel protection system, which is a separate system that will be monitored by the control system.

3.1 Operational Requirements

The control system will have to support a facility operating in excess of 5000 hours per year but with high control system availability required at all times. The facility is expected to operate for around twenty years.

The operational functions to be provided by the control system must include the routine operation of the facility by the operator, support for the technical groups to diagnose equipment and for the physics group to carry out experiments in characterising the facility.

3.2 Interface Requirements

Considering typical systems across each of the accelerators and the 7 beamlines there will be around 4000 physical devices that will require interfacing into the control system. The estimated numbers of interface types, not necessarily channels, for these devices is shown in Table 1.

Table 1: Numbers of interface types

Interface type	Quantity
Analogue	1677
Digital	1111
VME back plane	73
Video	8
GPIB	2
Serial	2116
System Integration	8

4 DIAMOND CONTROL SYSTEM

The DIAMOND control system will adopt the standard two-layer architecture with workstations as the client and VME crates as the server. There will be no field bus to a third layer, but extensive use will be made of a field bus or serial interfaces from layer two, to the equipment.

4.1 EPICS Tool Kit

The current development work is using EPICS 3.13.4 and VxWorks 5.4, which is providing a stable environment. The versions of EPICS and VxWorks to be used for the installed system will be fixed later in the design phase.

4.2 Equipment Interface

The interface from the control system to the equipment will be through VME64X Input Output Controllers (IOCs). IOCs will be installed for each technical system e.g. Vacuum and for each geographical area i.e. per cell on the storage ring, giving a total of around 212 IOCs, as shown in table 2. These will use PPC processor boards, Industry Pack carriers and modules as the principal interfaces. The preferred interfaces to equipment will be analogue, digital and serial (RS232, RS422 etc). Support will also be provided for at least one field bus to interface PLCs. The choice of field bus is currently undecided but preference will be for an open standard field bus. Each IOC will also contain an event receiver for synchronous operation and accurate time stamping of data. This combination allows for a high density of IO from a compact VME system, with most requirements being satisfied by four VME modules housed in seven slot crates.

Table 2: Numbers of IOCs

IOCs	Linac	Tx Paths	Booster	SR	BLs (7)
Main Magnets		1	1	25	
Steering Magnets		1	1	24	
RF	1		2	5	
Vacuum	1	2	4	48	7
Diagnostics	1	2	4	25	
Pulsed PSUs			1	1	
Personnel Safety	1			2	2
Vessel protection				24	
Rad. Monitors	1			2	2
IDs or Motors				7	7
Misc				5	2
TOTALS	5	6	13	168	20

4.3 Serial Interfaces

The decision has been taken to support the interfacing of equipment through serial interfaces. This offers benefits in being able to use commercial off the self equipment, reduced wiring, better correlation with local equipment values, and reduced installation and commissioning times. It is recognised that serial data processing will impose an IOC CPU overhead and supporting multiple vendor specific protocols will impose a development overhead.

4.4 Equipment protection

Equipment protection is included within the control system where required, with the control system issuing hardwired "permit to operate" signals to equipment. Three levels of protection, determined by assessment of damage/cost caused by failure, are: High Integrity, provided in hardware; Routine Interlocking, provided by PLCs; Prudent Operational Limits, provided by IOCs.

Global interlocks are required to protect the storage ring vessel and dipole magnets. These will be generated at each control and instrumentation area, converted and sent via the control system computer room over the fibre optic cables to the RF plant and dipole PSUs respectively.

4.5 Consoles

The consoles will be workstations and for cost reasons these are likely to be PCs running Linux. There is also a requirement to provide a control interface to Win32 based applications.

4.6 Central Servers

Central servers will be provided for development, applications, relational database, intranet and internet, archiver, alarm management and machine model and IOC booting. At each control and instrumentation area there will be local a boot server for the IOCs. These will be the primary IOC boot servers to minimise network load from IOCs rebooting. Consideration is being given to standardising all servers on Linux.

4.7 Applications

The principal application requirements can be met through the standard EPICS tools, control panels through MEDM, alarm management through Alarm Handler, archiving through Channel Archiver together with scripting languages for rapid application development.

A standard application for viewing and controlling channels in a tabular form is planned and an application for applying settings to a large number of the channels, to sequence the accelerator from one mode to another will be developed.

Applications will interface to the control system through the CDEV [7] abstraction layer. This will facilitate the integration of dynamic control data from Channel Access and persistent data stored in a relational database (RDB).

Soft records are currently being used to simulate the channels available in the final control system and so facilitate early application development.

4.8 Configuration and management

Capfast [8] is currently being used for IOC DB configuration but consideration is being given as to whether Visual DCT [9] offers a greater long-term future.

The RDB is planned to manage conventional control system information together with the control system configuration. All IOC DBs will be generated out of the RDB so ensuring that there is a central repository for all control parameters. Work required to link the EPICS database build process together with RDB is current being defined.

4.9 Physical structure

The control system will interface to the other technical systems at 48 control and instrumentation areas covering the linac, booster, transfer lines, plant rooms, storage ring and beamlines. For the storage ring there will be one control and instrumentation area per cell, thereby ensuring each cell is self-contained.

4.10 Network

A network infrastructure will connect the control system computer room to each of the 48 control and instrumentation areas with single and multi mode fibres. The fibre will provide two computer networks: one for control system and one for other computer systems to enable effective management of traffic on the control system network. Each network will use a central switch in the control system computer room and the control network will have secondary switches at each control and instrumentation areas. The fibres will further be used for event distribution, global interlocking and beam position feedback system

5 CONSTRUCTION

The detailed programme of work for construction phase of the project is currently being planned with latest options presented.

5.1 Procurement

The sub system below the IOC, principally PLC based, will be procured as design and build contracts from conventional systems houses. The procurement of complete technical sub systems, e.g. Linac with EPICS IOCs fully integrated, is being discussed with likely vendors.

The majority of IOCs will be procured on a component basis and integrated in house. The procurement of VME modules and instrumentation with EPICS device support is being investigated, with one VME manufacturer already developing EPICS support to our requirements.

5.2 Commissioning.

Commissioning is planned to progress through the linac, booster and storage ring. The storage ring cells are based on three girders, which support the vessel and magnets. These will be constructed offline and at this stage tested in conjunction with the control system. By ensuring each cell of the storage ring is self-contained, as the technical systems are installed the control system for that cell can be set to work and commissioned.

6 CONCLUSION

The structure for the DIAMOND control system is now being defined. With the control system well specified early on in the project the detail design can be resolved to ensure a high level of functionality available for day one commissioning.

7 REFERENCES

[1] M.W.Poole et al, "The DIAMOND Project: An Advanced Light Source",PAC2001 Chicago, 2001.

[2] http://www.esrf.fr/computing/cs/taco/taco.html

[3] http://www.esrf.fr/computing/cs/tango/tango_basic/index.html

[4] M. T. Heron, B. G. Martlew, A. Oates and W. R. Rawlinson, "An analysis of the SRS Control System upgrade", PCaPAC2000, DESY Hamburg, 2000.

[5] http://www.aps.anl.gov/epics/php.php

[6] M.T. Heron, B.G. Martlew, "A Review of Options for the DIAMOND Control System", PAC'99, New York April 1999 .

[7] http://www.jlab.org/cdev/

[8] http://lansce.lanl.gov/lansce8/epics/dbase/capfast/caprefv2-1.html

[9] http://www.sls.psi.ch/controls/software/VisualDCT/visual_dct_demo.html

TUAP028
physics/0111095

STATUS UPDATE ON THE ISAC CONTROL SYSTEM

R. Keitel, D.Bishop, D.Dale, H.Hui, S.Kadantsev, M.Leross, R.Nussbaumer, J.Richards, E.Tikhomolov, G.Waters, TRIUMF, Vancouver, B.C., Canada

Abstract

Implementation and commissioning of the EPICS based control system for the ISAC radioactive beam facility was completed. The target ion source, mass separator, and low-energy beam-lines started beam production for experiments. In parallel, controls for the accelerator system, a radio-frequency quadrupole followed by a drift-tube linac, and several high-energy beam-lines were implemented and commissioned. The number of controlled devices more than doubled since the last report. An overview of relational database usage, software engineering and quality control methods used will be given.

1 ISAC

The ISAC radioactive beam facility at TRIUMF was recently completed and achieved the design specifications. Fully accelerated radioactive beams (1.5 MeV/u) were produced for the first time in October of this year and routine delivery of accelerated beams to experiments has started.

Since the last report [1], the size of the facility increased considerably. In the low energy area, a second electrostatic beam line was added which provides laser-polarized beams. In the high-energy area, the stripper section, a five-tank drift tube linac and three experimental beam lines were added. It should be noted that the ISAC control system also provides controls for vacuum, optics, and diagnostics systems of large experiments, such as the DRAGON spectrometer.

In addition, temporary controls had to be provided for a high current target test with 100 μA of 500 MeV protons.

2 CONTROL SYSTEM

The ISAC control system uses the EPICS control system toolkit. Within this framework, a "standard model" architecture is implemented on an Ethernet backbone, using Sun workstation servers, VME based input/output computers (IOCs), and console X-terminals.

2.1 I/O Architecture Recap

All ISAC device control is executed through the IOCs. The I/O hardware consists of several subsystems:

- Beam diagnostics devices are controlled via VME modules in order to maintain tight coupling with the IOC CPU [2].
- Beam optics devices have intelligent local controllers, which are supervised by the IOCs via CAN-bus networks [3].
- Vacuum devices are controlled by Modicon Quantum series PLCs, which are peer nodes on the controls Ethernet and are supervised from the IOCs via TCP/IP.
- "Special" devices for laser control use Ethernet based GPIB interfaces [4]
- RF control systems are VXI based. They are peer nodes on the controls Ethernet and are supervised from the IOCs. These systems are maintained by the TRIUMF RF controls group and are outside the scope of this paper [5]

2.2 Scope

The recent additions to the ISAC machine more than doubled the number of controlled devices to approximately 2000, and the number of supervised RF systems increased from 1 to 15. The control system now contains 12 IOCs, which interface to 4200 digital, 2500 analog hardware channels, and 40 motors. The EPICS runtime databases grew to a total of 52000 records.

2.3 Hardware Additions

Most of the support for the new ISAC sections involved doing "more of the same". In addition, the requirements for laser control and fast faraday cup read-out introduced GPIB-bus devices into the system. They were interfaced with Ethernet based GPIB controllers from National Instruments (GPIP-ENET).

The Ethernet infrastructure was improved by introducing 100BaseT switches in a topology, which allows step-wise isolation of the control system for trouble-shooting and diagnostics. The ISAC controls section of the Ethernet was separated from the rest of the site with a firewall implemented on a Linux system.

2.4 Software Additions

Again, a lot of work during the report period involved doing “more of the same”, i.e.

- Creating EPICS function block data bases to implement IOC software functionality
- Programming ladder logic for the new vacuum subsystems
- Providing operator interface pages for all new vacuum and optics sub-systems as well the supporting device control panel pages, scripts, save-restore and alarm configurations.

Considerable effort went into automating some of these software activities, in order to improve the quality of the control system and the productivity of the small control system software group. This will be touched upon in the “Quality control” section below.

Laser stabilization: The production of Laser-polarized ion beams required integration of GPIB devices and feedback control to stabilize the laser system. This was implemented combining EPICS controls with a LABVIEW system. Although workable, this solution is not robust enough, partially due to the difficulty of integrating LABVIEW into the overall system and some problems with the Ethernet-based GPIB interface. More details about this sub-project can be found in [4].

Data Archiver: In trying to archive ISAC data, unhappy experiences both with the EPICS *ar* tool and with a beta version of the high performance binary channel archiver led to the development of a simple archiving tool *trar*, which is tailored to the ISAC requirements:

- ASCII data files, default: one date-stamped file per day
- Archive groups with different disk write intervals
- Conditional archiving of groups
- Possibility of archiving any group to an arbitrary file
- Retaining IOC sampling information by archiving data minimum and maximum between disk write intervals

An accompanying retrieval utility was written, which is wrapped with a Perl-Tk GUI and uses GNUplot for display of the data.

Macro tool: A macro tool was developed for capturing operator actions and replaying them. It is implemented as a Perl-Tk script, which uses the output files of the EPICS activity logging facility. The operator presses a “Macro start” button, which inserts a “start recording” message into the activity-logging file. A “Macro stop” button inserts a “stop recording” message. If a “Macro save” button is pressed, the tool extracts all channel access commands between these two messages, retains the ones issued by the macro operator, and formats them into a procedure file, which can be executed by the host based sequencer *cppe* [1].

2.5 Operations Support

The ISAC operations console started out with two multi-monitor PC systems under Windows98, which are mainly used as X-terminals using the Exceed X-server from Hummingbird, Inc. Windows98 was chosen because of its multi-monitor capabilities but proved to be unsuitable for routine operations. At least one reboot of the operating system was required per operations shift. Windows98 was replaced with Linux, which increased both performance and reliability dramatically.

The save/restore system was completely re-worked to allow conflict free restoring of beam tunes depending on the selected beam modes for each ion source.

EPICS access control was implemented. Only operations consoles have write access to process variables, but the controls group, beam physicists, and specialists are able to give themselves “self service” write access for a limited time. Experimenters are allowed write access to their own sub-systems and selected devices, dependent on the selected beam modes.

The ISAC controls web site is constantly upgraded with system and trouble-shooting information.

2.6 Quality Control

Relational Data Base: In the initial stages of the ISAC project, only the interactive EPICS tools were used to generate operator displays (edd display editor) and IOC function-block databases (CAPFAST schematic editor) In order to improve productivity and reduce error frequency, the control system was back-ended with a relational device data base. This database is implemented using Paradox and is supported by several tools implemented as Perl scripts. This system contains all device parameters, interlock specifications and all required information to

- Generate Interlock specification documents for approval by the system specialists
- Automatically generate CAPFAST schematics with device instantiation for ISAC sub-systems [6]
- Automatically generate all device control panels with visualization of device interlocks [7]
- Check the interlock implementation in the PLC programs against the interlock specification [7]
- Generate test documents for sub-system commissioning
- Generate web pages for documentation of module and channel usage in VME crates.

Bypass/Force summaries: The control system generates summary information about bypassed device interlocks and forced device states, both overall and by sub-system. The corresponding IOC databases are now automatically generated by Perl scripts, which collect all necessary EPICS record information from the IOC sub-system databases.

Wiring documentation: ISAC device wiring diagrams are semi-automatically maintained on the ISAC controls web site. A Perl tool was developed which generates HTML pages using screen dumps of EPICS OPI pages. The OPI call-up buttons on these pages, which would start device control panels on the controls console, are automatically turned into hyperlinks for calling up device wiring diagrams in Adobe PDF format. Buttons, which navigate between OPI pages, maintain this functionality on the web pages.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the help of C. Payne from the ISAC operations group in installing and maintaining the group’s Linux workstations and firewalls. K. Langton, K. Pelzer, and D. Boehne helped with the hardware installation. Thanks also to the ISAC operations crew and machine physicists for their input and patience while the system was and is being debugged.

REFERENCES

[1] R. Keitel, et al., 'Design and Commissioning of the ISAC Control System at TRIUMF', ICALEPCS99, Trieste, October 1999, p. 674

[2] D. Bishop et al., ‘Custom VME Modules for TRIUMF/ISAC Beam diagnostics’, ICALEPCS99, Trieste, October 1999, p.226

[3] D. Bishop et al., ‘Distributed Power Supply Control Using CAN-bus’, ICALEPCS97, Beijing, November 1997, p. 315

[4] R. Nussbaumer, ‘The Laser Stabilization Controls for the ISAC Beam Polarizer’, this conference

[5] K. Fong et al., ‘RF Control Systems for the TRIUMF ISAC RF’, APAC01, to be published

[6] R. Keitel, ‘Generating EPICS IOC Data Bases from a Relational Data Base – A Different Approach’, this conference

[7] R. Keitel, R. Nussbaumer, ‘Automated Checking and Visualization of Interlocks in the ISAC Control System’, this conference

DEVELOPMENT STATUS OF EPICS APPLICATION FOR PLS COMPUTER CONTROL SYSTEM

J. H. Kim, K. M. Ha, J. M. Kim, K. H. Kim, E. H. Lee, J. C. Yoon,
J. W. Lee, J. Choi, S. H. Nam, J. Y. Huang and I. S. Ko
Pohang Accelerator Laboratory, POSTECH, Pohang 790-784, Korea

Abstract

The control system for Pohang Light Source (PLS) was designed about twelve years ago and has been working without serious problem so far. However, as the demand for the application programs for accelerator physics and data analysis increases, the present control system shows the lack of flexibility and control speed. To have faster and more flexible performance, we have decided to convert the current control system into an EPICS-based one on the two-layered hierarchical Ethernet network. EPICS Toolkit will be applied to our system while preserving our investment in most hardware. All EPICS system will be replaced to ensure homogeneous layout for PLS linac and storage ring. This paper presents the status of the application development with EPICS on a test bed system and the future plans.

1 INTRODUCTION

A basic function of the control system for the accelerator is to operate various subsystems remotely in real time manner and to archive necessary data for the operators and users. In this sense, the current control system for the PLS satisfies the basic function in a daily operation [1-3]. However, this custom-built, twelve-years-old system is facing difficulties to catch up the growing number of beam lines last decade from two to almost twenty and many insertion devices. Moreover, fast changing technologies in this field make the routine maintenance difficult because of short lifetime of hardware and software. For example, some VME boards are no more in the market since long time, so the replacement is simply impossible. We are now facing to adopt rapidly growing information, telecommunication, networking, and computer technologies into our new accelerator control system. This system should provide not only fast and reliable performances, but also various advanced services such as modeling and simulation for the operators, accelerator physicists, and beam line users. A short list below shows the reason why we need to upgrade the current PLS control system (or replace the current system to new one in a sense):

- Replace obsolete subsystems to worth new ones
- Adopt current technology development in IT/Control system
- Getting better reliability in system
- Easy and quick maintenance capability
- More flexibility to adopt ever growing demands from users
- Easy use of operation data by accelerator physicists
- Provide good data handling tools for beam line users
- Maintain the system configuration within industrial, international, and de facto standards

2 COMPARISON BETWEEN THE CURRENT SYSTEM TO NEW ONE

The design of the current control system was finished in 1990 and completed by in-house members of PLS in 1994. It has three layers of hierarchical structure. The VME data acquisition system was very similar to Elettra (Trieste, Italy) control system, and the upper layer in Unix was adopted from the concept of SPEAR (SLAC, USA) control system. This system, especially hardware platform, has been used until now with good reliability and acceptable performance. However, the rapidly expanding beamlines and the new insertion devices, increasing the control points, produce serious problems to this system. This difficulty is enhanced by the poor structure of database. Because of this, developments of various application programs for accelerator operators, physicists, and users are limited.

In the current control system [1-3], the data or the control/monitoring commands are delivered through three layers of hierarchical (and physical) structure that means three different computers. Due to this, the response time takes longer and it has higher chance of failure in the data delivery. Also, there are two databases: one in Unix layer (or operator interface layer) and another in VME layer (or data process layer) which functions as a supervisory control computer. Even though the original aim was to organize handling of data in each layer, this actually gives data handling more complex, longer data handling time, and waste of resources. For the operators and accelerator physicists, it is not easy to make their own application program because of complex database and lack of utility tools. From a hardware point of view, MIL-STD-1553B channel and 10 Mbps Ethernet were the top of the line products when they were chosen, but not any more. This network is too slow to handle large image data.

By counting various aspects, we have decided to build new control system for PLS with replacing necessary hardware and network to new ones, and intensive use of EPICS in the software development [4-6]. The new control system has two layers of hierarchical structure and a centralized database. This gives the fast response in the system and the effective use of resources. Accelerator operators and physicists can build their own application, such as alarm handler (AR) easily by using simplified database and tools that EPICS provides. Since this upgrade or replacement requires a lot of efforts and experienced manpower, we decide to build a test bed that is a scaled version of one full system [7].

3 TESTBED FOR PLS CONTROL SYSTEM UPGRADE

Figure 1 shows the schematic structure of the test bed for PLS control system upgrade, and Table 1 lists the hardware used in the test bed. The current status of the test bed is following:

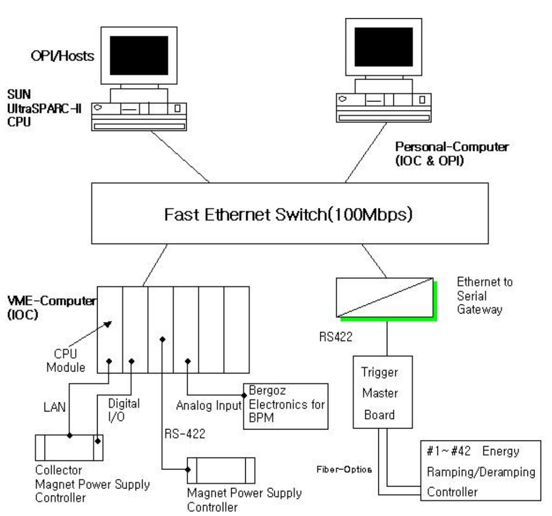


Figure 1: Schematic structure of test bed control system.

3.1 Host

Most of EPICS extension tools are successfully installed. As an example of the PLS main control screen, the control/monitoring window for the magnet power supplies (MPS) is completed as shown in Fig. 2 with MEDM tool. A test of channel access with IOC has been completed. There is also a panel for energy ramping control in Fig. 3 by LookOut (NI product) under Windows 2000 environment. This LookOut is connected to IOC via channel access client.

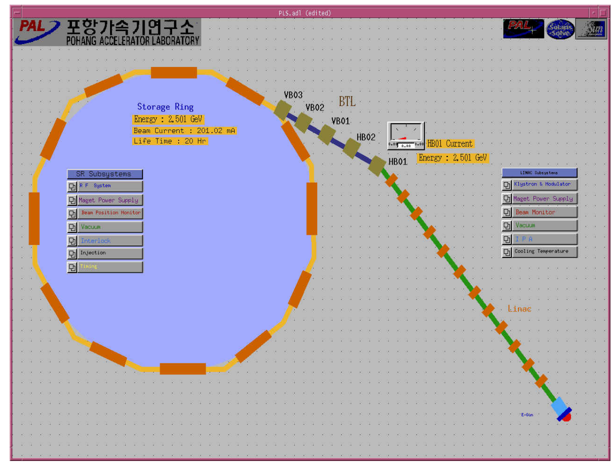


Figure 2: PLS main control screen

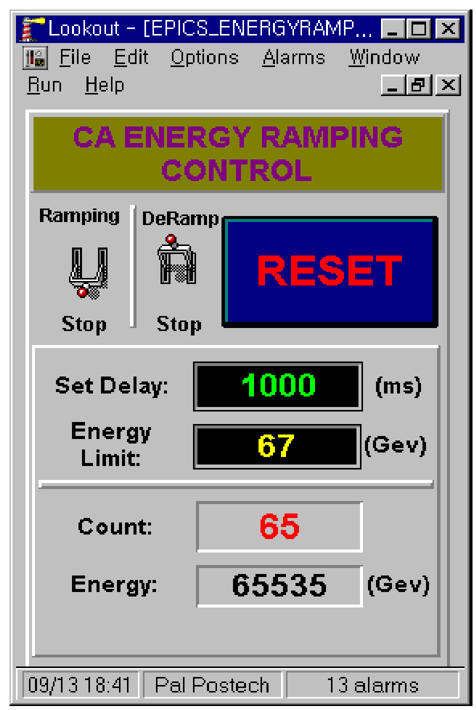


Figure 3: Ramping /Deramping Control panel

3.2 IOC

There are two IOC hardware platforms (VME, PC) to satisfy the current PLS control system installed in the field. Therefore, the test bed has independent development environment for VME and PC IOC. In a VME crate under vxWorks, PowerPC SBC (MVME 5100) and M68K SBC (MVME147) are used along with I/O boards connected to MPS controllers, BPM electronics, and ramping controller as listed in Table 1. For this system, EPICS Base R3.13.5 is used. At present, device access tests are being carried our by using records supported by EPICS. We are now testing the several I/O boards to select the optimum interface so that our machine controller can work in the lower level. Also, under Windows NT and 2000 environments, EPICS R13.14.alpha2 version is used to connect for PC IOC. Record supports are obtained by producing threads from EPICS standard aiRecord and aoRecord. The Capfast and VDCT are used as a data configuration tool.

3.3 Local Controller

The new controllers are being developed for the corrector magnet power supply and the linac modulator control with EPICS architecture concept. To support the existing and EPICS control system hardware interface, these controllers have two independent data channels to connect the IOC within controllers. Especially, the MPS corrector controller is designed to have Ethernet controller chip with TCP/MODBUS communication protocol [8]. The prototype modulator controller based on the industrial PC is changed from having simple functions for Linac modulator and klystron to adopting the EPICS IOC function.

Table 1: Hardware list used test bed control system.

Hardware	Specification	Purpose
SUN ULTRA 80	Two UltraSPARC-II 450MHz CPU	SUN WS Host
MVME5100	PowerPC SBC	VME IOC CPU B'D
MVME147	M68K SBC	VME IOC CPU B'D
Pentium_III PC	Pentium_III MotherBoard	PC IOC/Host CPU B'D
TSVME500	4 Ch. Serial RS422	MPS controller-1
TPMC866	8 Ch. Serial RS422	MPS controller-2
VMIVME-3122-331	64Ch. ADC Board	BPM Electronic intf.
ADAM-4572	Ethernet to Serial Controller	Ramping Magnet
CES VMDIS 8003	VME Bus monitor	VME bus debugging
Home made Corrector MPS Controller	One chip intelligent controller board with Open Modbus/Tcp intf.	Corrector MPS Control/monitor
Home made MPS Controller	One_chip intelligent controller board with serial interface	Linac MPS Control/monitor

4 FUTURE PLANS

Even though this test bed project is limited in size, this is a good opportunity to educate EPICS to software engineers, operators, and others who have interests in learning how to use EPICS in PLS [9]. We will continue to work on this test bed by increasing the number of IOCs by early 2002. After we have satisfying results, we will expand the test bed to one full cell out of 12 cells in the Winter 2002 maintenance period. A complete upgrade of PLS control system is expected by 2003.

ACKNOWLEDGMENT

This work is supported by Pohang Iron & Steel Co. and Ministry of Science and Technology, Korea.

REFERENCES

[1] I. S. Ko, J. H. Kim, J. Choi, and W. Namkung, "Control System of PLS 2-GeV Electron Linac," IEEE Trans. Nulc. Sci., 43, 25 (1996).

[2] I. S. Ko, Jin W. Lee, Jong C. Yun, Eun H. Lee and Byum R. Park, "Computer Control System of PLS 2-GeV Storage ring," IEEE Trans. Nucl. Sci., 45, 2012 (1998).

[3] I. S. Ko, J. H. Kim, S. C. Kim, "Test Bed of Control System Using Multimedia Technology at PLS," IEEE Trans. Nulc. Sci., 45, 2016 (1998).

[4] http://www.aps.anl.gov/epics

[5] http://mesa53.lanl.gov/lansce8/Epics/epics.htm

[6] http://csg.lbl.gov/EPICS_collaboration.html

[7] J. H. Kim, K. M. Ha, J. M. Kim, S. H. Nam, J. Y. Huang, I. S. Ko, "EPICS based upgrade plan for PLS control system," Korean Physical Society 2001 Fall Meeting, Kwang-ju, Korea (2001).

[8] OPEN MODBUS/TCP SPECIFICATION Release 1.0, 29 March 1999

[9] K. H. Kim, J. Choi, T. Y. Lee, G. N. Kim, M. H. Cho, W. Namkung, and I. S. Ko, "Development of the Simulator for the Global Orbit Feedback System using EPICS," in these proceedings

TUAP030
physics/0111151

USE OF EPICS FOR HIGH-LEVEL CONTROL OF SNS CONVENTIONAL FACILITIES [1]

J.K.Munro, Jr., J.E.Cleaves, E.L.Williams, Jr., D.J.Nypaver, ORNL, Oak Ridge, TN 37831, USA
K.-U.Kasemir, LANL, Los Alamos, NM 87545, USA
R.D.Meyer, Sverdrup Technologies, Tullahoma, TN 37388, USA

Abstract

The SNS Project intends to integrate Conventional Facility Controls with its EPICS-based Accelerator and Target Control Systems. EPICS will therefore be used to provide distributed high-level access to all subsystems of the SNS conventional facilities, including cooling water towers, chilled water, de-ionized water, HVAC, waste processing, and power monitoring. EPICS will provide operator displays and high-level control for more than 1000 process variables. EPICS support will be provided by four IOCs using PowerPC-based VMEbus controllers. Low-level control will be provided by Allen-Bradley ControlLogix PLCs that will communicate among themselves using ControlNet and with EPICS using EtherNet/IP [2]. Both the PLC layer and the EPICS layer will be implemented by an industrial supplier. File server support will be Linux-based and Concurrent Versions System (CVS) will be used to manage version control, both for EPICS and for PLC program and configuration files. All system process variable names, hardware, software, and database configuration properties will be maintained in a master Oracle database which will be used to generate and maintain EPICS and PLC databases for the entire project.

1 INTRODUCTION

This paper describes the business and technical arrangements made by the SNS Project with Sverdrup Technologies (SvT) to develop the control systems for the SNS conventional facilities (CF). EPICS will be used to provide high-level control and display of information about CF subsystems to the operators; commercial PLC technology will provide low-level control of all CF hardware. Primary CF subsystems include HVAC, chilled water, de-ionized water, air handling, waste processing, and power monitoring.

New elements for this project exist in both the technical and business areas. In the technical area the Linux operating system is being used to support all EPICS functions and development tools; the Motorola MVME2101 PowerPC processor board is being used in all I/O controllers (IOCs). In the business area this development effort probably is the first time all CF control systems are being integrated “up front” into the EPICS control system for a large accelerator facility. This infrastructure must be built early to provide support for the major accelerator subsystems as they are added; hence, is one of the first major SNS subsystems to be completed and put into operation. To guide development, standards and guidelines for control system software and operator displays have been prepared, approved, and distributed to the development contractor(s). As a consequence, the CF control system development schedule has driven the development of standards and guidelines and selection of hardware that will be used for development of control systems for other SNS subsystems.

The basic plan is for the SvT developers to identify all the I/O points required for the CF. I/O point names (following a standard naming convention) and descriptions are entered into spreadsheets from which they can be read by scripts to generate tag name lists used in the PLCs and process variable (PV) names for EPICS databases which are loaded into IOCs. Tag names are grouped by EPICS record type and packed into data transfer arrays which are used by a communications driver (EtherNet/IP) [2] to move data for the operator displays between the PLCs and IOCs. The SvT developers will write the ladder logic programs used by the PLCs for low-level control of the hardware, will develop the operator screens needed for high-level control, and will generate the EPICS databases to be loaded into the IOCs.

2 BUSINESS ARRANGEMENTS

All SNS CF control systems are being developed through fixed price contracts with SvT, located in Tullahoma, TN, a few hours’ drive from the SNS Project office in Oak Ridge. SvT engineers have broad experience in the integration of highly technical systems. Contracts have been placed for system (Title I) design, detailed (Title II) design, cabinet fabrication design, electronics procurement and cabinet fabrication, software and database development (both EPICS and PLC logic), loop testing after equipment installation, and procurement and storage of sensors and control elements. Cabinet, sensor, control element, cabling/conduit installation, and field cabling termination will be done via fixed price contract with an installation contractor with expertise in this area.

Having a single integrating contractor perform all this work eliminates many interfaces and the need to pass information from one contractor to another. It also allows loading developed software into cabinet equipment and testing the entire system before delivery for installation.

3 PLC APPLICATION DEVELOPMENT ENVIRONMENT

All PLC application development is being done using RS Logix 5000 on a Windows 2000 system to configure the hardware and write the ladder logic control programs. All I/O point information is being maintained in Excel spreadsheets and MS Access databases.

Considerable effort went into the development of a standard naming convention for control PVs. The results of this effort have been published in a project reference document [3]. Scripts have been written to use the common signal naming syntax to read names from the I/O points database maintained in Excel spreadsheets and write files for the PLC and EPICS databases with PV names constructed according to the different syntax rules used for tag names (PLCs) and the corresponding EPICS database records.

Performance measurements made for data transfers between PLCs and IOCs using the EtherNet/IP driver [2] showed in dramatic fashion the importance of using arrays to transfer data quickly and efficiently. For this reason a separate data transfer array is used for each different EPICS record type, e.g., ai, ao, bi, bo, mbbi, and mbbo.

4 EPICS APPLICATION DEVELOPMENT ENVIRONMENT

The computers and software used by SvT were first set up in Oak Ridge the way they would be configured in Tullahoma for the development work. All software tools were installed and configured. An example application was set up and used to check out every step of the development sequence, from I/O points listed in an Excel spreadsheet to an operating application on a PLC with data transferred to an EPICS database loaded into an IOC using the EtherNet/IP driver, and PVs displayed on screens built using the Extensible Display Manager (EDM). This example application was later used in training exercises.

A minimal development environment sufficient to duplicate the environment set up at SvT was set up in Oak Ridge for use in providing technical support to the SvT team. This system is also available for giving demonstrations at SNS program review meetings. Any upgrades to operating systems, application software, or development procedures are first implemented and tested on the Oak Ridge environment before being installed on the Tullahoma system. This minimizes impacts on the developers and supports strong configuration control.

2.1 Hardware Configuration

Seven workstation class (Pentium III Xeon) computers were purchased for programming the PLCs (4 Windows 2000 systems) and developing the EPICS screens and databases (3 Linux systems). Two Linux file servers with 5 disk RAID systems were purchased. Identical EPICS application development environments (ADEs were set up on them. A second Ethernet card was installed in the SvT server to permit setting up the development environment at Tullahoma on a private LAN. These two servers communicate via a T1 line.

2.2 Software Configuration

An SNS CF ADE was set up to minimize the amount of EPICS system administration work needed by SvT. The directory structure is consistent with conventions used by the scripts, rules, and makefiles provided with the official EPICS software distribution and described in documents available from the Advanced Photon Source Web site at Argonne National Laboratory.

2.2 Application Development Process

The SvT development team has already identified the CF I/O points from process and instrument drawings (P&IDs) and has used spreadsheets and an MS Access database to archive and maintain this data. Scripts were written to apply the tag and EPICS PV naming conventions to this I/O point data to generate a set of spreadsheets which are used as input to RS Logix 5000 to generate tag names and to scripts used to build the EPICS database files needed for the operator displays. The tag names are used in the ladder logic programs which are downloaded into the PLCs. Data for selected tag names are packed into transfer arrays which an EtherNet/IP driver reads from PLC memory and writes into the EPICS database records accessed by the displays.

All these sets of I/O points data will eventually be loaded into the SNS master Oracle database and maintained by tools written for this purpose. A long-term goal is to automate the process of generating and maintaining the data that must be downloaded to the PLCs and IOCs from the master Oracle database so that all control system data can be centrally maintained and administratively controlled. This should also allow IOC databases to be modified and downloaded more reliably and IOCs to be rebooted automatically.

Training of the development team was built around the development sequence and processes that would be used to generate all parts of the CF control system. It consisted of an abbreviated version of the standard EPICS course, with emphasis on the more limited tool set needed to build the CF applications.

5 CONFIGURATION CONTROL

During development the SvT team will arrange for configuration control of the I/O points database, scripts, configuration files, PLC programs, and operator display screens. All primary data, scripts, programs, and display screen files from SvT will eventually be archived and maintained in SNS project CVS repositories.

Configuration control of all EPICS related software is done using the SNS CVS repository in Oak Ridge. New versions of the screen editor and display manager EDM are used and tested in Oak Ridge before being installed on the file server in Tullahoma. The same procedure will be followed for any changes to EPICS base or any of the EPICS extensions obtained from the primary repository at Argonne National Laboratory.

6 CONCLUSIONS

A number of lessons have been learned in the areas of developing requirements, functional system design documents, standards, training, software development environment, technical support, configuration management, and contract management from experience to date. This effort is leading the way for a significant amount of control system application development to follow for SNS. A few highlights will be indicated here.

Document tag names, signal names, setpoints, EPICS record parameters, and logic functions in a manner that can be used to test completed software (e.g., in spreadsheet or database applications). Generate as much of this information as possible from P&ID diagrams: this permits converting design information in P&ID language to data in formats that can be used directly and tested by software applications. The contractor who will develop the CF control systems should generate all the design information before any control system development activities begin: this should help to insure that the contractor software developers understand what must be built.

Standards should encompass all aspects of the work, should be documented in a design criteria or similar document, then applied to some representative prototypical examples and tested to verify their features accomplish what is desired/expected.

Choose examples for use in training that can be used later in what must be developed. These examples must use the standards and tools in the same way as the product to be delivered. Set the examples up in the same ADE (in a “training” directory) that will be used for development.

In preparing the ADE, make reasonable efforts to use database application and other tools with which the contractor is already familiar. This should speed development time and support development of a more reliable product.

REFERENCES

[1] Work supported by the US Department of Energy under contract DE-AC05-00OR22725.

[2] Kay-Uwe Kasemir and L.R. Dalesio, “Interfacing the ControlLogix PLC Over EtherNet/IP” ICALEPCS 2001, San Jose, CA 27-30 November 2001.

[3] “Systems Requirement Document for Equipment, Device, and Signal Naming,” SNS 102000000-SR00001-R00, August 2001.

TUAP031
physics/0111051

STATUS AND CONTROLS REQUIREMENTS OF THE PLANNED HEAVY ION TUMOR THERAPY ACCELERATOR FACILITY HICAT

R. Bär, H. Eickhoff, T. Haberer; GSI Darmstadt, Germany

Abstract

The HICAT project is a Heavy Ion accelerator for light ion Cancer Treatment to be built for the clinics in Heidelberg, Germany. It consists of a 7 MeV/u linac, a compact synchrotron and three treatment places, one of them located after a 360 degree gantry beam-line. The facility will implement the intensity controlled raster-scanning technique that was developed and successfully demonstrated at GSI. In order to produce the beams with the characteristics requested by the treatment sequencer, the accelerator must operate on a pulse-to-pulse basis with different settings. This concept imposes strict and challenging demands on the operation of the accelerators and hence the control system of the facility. The control system should be developed, installed and maintained by and under the complete responsibility of an industrial system provider, using a state-of-the-art system and wide-spread industrial components wherever possible. This presentation covers the status of the project and the requirements on the control system.

1 HICAT MACHINE LAYOUT

The HICAT accelerator facility was designed to meet the medical requirements used for the successful GSI cancer treatment program. HICAT consists of two ECR ion sources with independent spectrometer lines to produce both low LET ions (p, He) and high LET ions (C, O) to cover the medical requirements and allow for fast switching. The beams will be accelerated by a compact-designed 7 MeV/u linac (RFQ and IH-DTL) and a 6.5 Tm synchrotron for beam energies of up to 430 MeV/u. In order to meet the intensity requirements, multi-turn injection will be used for beam accumulation. For extraction, the rf knock-out method in combination with a variable extraction time will be implemented. The beams can be delivered to three medical treatment areas, located after two fixed horizontal beam lines and one 360° rotating beam transport system (isocentric gantry), as well as one quality assurance/experimental place.

No passive elements are foreseen to match the dose distribution to the individual tumor geometry. Instead, all beam lines will be equipped with horizontal and vertical scanning magnets and beam diagnostic devices for the intensity controlled-raster scanning technique that was developed and successfully demonstrated within the GSI experimental cancer treatment program with over 100 patients to date.

Fig. 1 shows the cross section of the first underground floor of the building which houses the accelerator sections, the patient treatment areas, local control rooms and various laboratories and offices. For a general description of the accelerator complex, see [1].

2 CONTROL SYSTEM

For the realization of the HICAT project, a close cooperation with industry is necessary. GSI has done the machine layout and worked out crucial technical specifications. The development and implementation of the control system will be performed by a commercial firm.

Therefore, the HICAT control system (cs) should be delivered under a supply contract established with an industrial system provider. The full development and configuration of the system will be the responsibility of the contractor, who will also perform commissioning and all tests on the site. GSI will advise and supervise the commercial contractor during the project.

The cs should be based on an industrial control system (e.g. commercially available SCADA system) that fulfills the specific requirements for controlling the facility. At a life cycle of 20 or more years, quite reasonable for such a system, the cs must be laid out for long service. Consequently, it is proposed to use wide-spread industrial standards and commercial solutions wherever possible.

The cs must meet the following general requirements:

- high reliability
- easy and efficient operation by a small crew
- development and maintenance by industry

Since a reliable control system is an indispensable element to supply beams for medical practice the system has to respect stringent requirements in terms of reliability, safety, maintainability and legal aspects such as the ones required by regulatory authorities. It must meet the following general requirements:

The HICAT cs runs the accelerator in a hospital environment. It must be easy to operate because daily operation of the accelerator is carried out by operators who are not necessarily accelerator specialists. The control system's functionality is determined by these facts, together with the controls demands of the accelerator facility in itself.

In particular, the following major tasks have to be covered by the accelerator cs:

- full control of the accelerator systems to deliver and assure a beam with characteristics requested by the patients' treatment plan on a pulse to pulse basis
- full control of the patient irradiation treatment with prescribed treatment plan and measure compliance of treatment
- patients' protection against wrong treatment
- control of all secondary and infrastructure systems
- minimal personal employment for controlling

The HICAT cs should be completely integrated, i.e. a single set of standards will be applied to all aspects of the facility. Although it is not the usual practice to include conventional facility controls in the accelerator cs, experience shows that signals from these systems are needed in the control room.

All systems – accelerators and conventional facilities – will be operated and monitored from a single control room. In addition there will be treatment control rooms for each treatment place from where the actual treatment is controlled and monitored.

3 TECHNICAL CONTROL REQUIREMENTS

Major aspects of the control system design are influenced by the experiences of the GSI cancer treatment program [2]; the requirements of this facility, however, exceeds those of the pilot project.

Logically, the HICAT cs will consist of two major integrated parts:

- the accelerator control system – controls the accelerator and all auxiliary systems
- control system for the medical treatment (one independent system for each treatment place) – controls the treatment sequencer and all treatment instrumentation

3.1 Process Data

The main characteristics of the HICAT facility are the application of the raster-scan method with active intensity-, energy- and beam size-variation in combination with the usage of an isocentric gantry to match the dose distribution to the individual tumor geometry. During the patient irradiation process (treatment mode) the treatment sequencer requests beams with certain characteristics on a pulse-to-pulse basis. The beam characteristics are fixed and pre-defined in a library of settings, e.g. 255 possible beam energy steps (Table 1). The task of the accelerator control system is to deliver these beams as requested.

Table 1: Process parameters

Steps	**Parameter**
4	ion species (p, He, C, O)
255	beam energy (50–430 MeV/u)
4	focus diameter (4–10 mm)
15	intensity level (variation 1 to 10^{-3})
~36	gantry angle (angle interpolation)
4	treatment area (beam lines)

In order to achieve high operational reliability the cs must meet the following requirements:

- use only proven and validated settings,
- protect settings from loss and accidental modification,
- inhibit operator's access/interference during irradiation,
- avoid networking for settings, provide all beams in stand-by on a pulse-to-pulse basis,

These requirements lead to a high number of settings (up to 100,000 per device) to hold at the device controller. Fortunately, the settings of the devices do not depend on all parameters in Table 1 at the same time. Because of the high number of data sets needed, manual tuning of the machine is prohibitive. To derive data automatically, a "theoretical model" of the accelerator will be used, allowing calculation of all device data from high-level machine parameters and applying once measured corrections.

3.2 Timing system

For the entire accelerator chain, including linac rf-structures, choppers, bunchers, and all synchrotron devices, a strict synchronization is required. Some time-critical functions (i.e. triggers, acceleration ramps) must by synchronized within $\Delta t<1–5$ µs at the worst. The cs must implement a timing controller that provides a system-wide synchronization for distributed devices with real-time performance. The system must be flexible enough to react on requests for beam extraction pauses and aborts.

3.3 Patient safety

Safety is a crucial issue for the cs, considering the special task of the facility. Due to the high number of devices acting on the beam, the accelerator control system cannot guarantee safe operation. Instead, it will implement a fast and fail-safe beam-abort system that reliably stops the beam within 200 •s. This is achieved by simultaneously triggering a chain of different actions using different devices which are all, by themselves, capable to prevent beam delivery to the treatment place.

The sophisticated on-line beam diagnostics of the treatment sequencer verifies the correctness of the beam parameters (position, focus, intensity) with respect to the treatment plan. In case of intolerable deviations to the treatment plan or failure of critical systems, the treatment control system triggers beam-abort channels to stop the irradiation treatment.

3.4 Scanner system

For the real-time treatment sequencer it is foreseen to use the existing state-of-the-art GSI-system [3] and adapt it to the new environment. The functions will be extended to feature gated beam extraction and end of extraction ahead of schedule.

4 STATUS AND OUTLOOK

Currently all essential technical specifications and requirements for the entire facility including the control system are being worked out.

Following the final approval by the board of the clinics by the end of 2001 an ambitious time schedule will be established. For the first quarter of 2002 a call for tender is scheduled. Contracts with industries are foreseen around middle of 2002.

According to the current project plan the first beam is scheduled in 2005 and the first patient treatments should take place in 2006 after an extensive commissioning phase.

REFERENCES

[1] H. Eickhoff *et. al.*, "The Proposed Dedicated Ion Beam Facility for Cancer Therapy at the Clinic in Heidelberg", EPAC2000, Vienna, June 2000.

[2] U. Krause, R. Steiner, "Adaption of a Synchrotron Control System for Heavy Ion Tumor Therapy", ICALEPCS '95, Chicago, USA, 1995.

[3] E. Badura *et. al.*, "Safety and Control System for the GSI Theray Project", ICALEPCS '97, Beijing, China, November 1997.

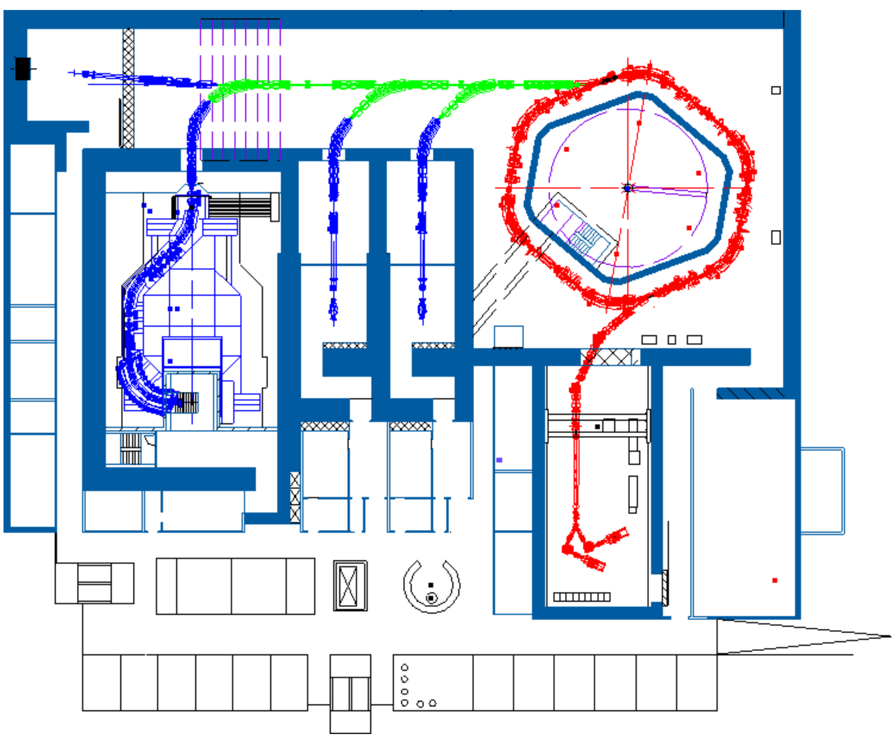


Figure 1: First underground level of the HICAT facility.

TUAP034
physics/0111190

REJUVENATION OF THE CONTROLS OF THE CERN ISOLDE FACILITY USING INDUSTRIAL COMPONENTS

J. Allard, O. Jonsson, F. Locci, G. Mornacchi, CERN, Geneva, Switzerland

Abstract

In the context of the general consolidation of the CERN ISOLDE facility, a project has been started to upgrade the ISOLDE control system. We describe the new ISOLDE control system, emphasizing the systematic use of industrial components such as PLCs and field buses, their integration with the existing, VME based, CERN PS control system and their potential applicability to both existing and new controls problems at the CERN PS complex. We also discuss how to extend a PLC-based solution to the case where real-time response is an issue.

1 INTRODUCTION

The On-Line mass-separator (ISOLDE) is a facility for the production of pure low energy radioactive ion beams at CERN [1]. A project for its consolidation has been started and the upgrade of the control system has been identified as one of the urgent jobs. To this end a sub-project has been defined with the mandate to consolidate the equipment layer of the ISOLDE control system and to integrate this latter with the PS controls infrastructure on the time scale of the ISOLDE 2002 start-up. The project builds on a previous study for the rejuvenation of the ISOLDE control system [2].

In this paper we describe the solution devised for the ISOLDE control system and report on the status of its implementation. Emphasis is put on the application of industrial components**.**

2 THE ISOLDE CONTROL SYSTEM

2.1 The existing ISOLDE control system

The existing ISOLDE control system [3] is based on commodity hardware (PCs), using Microsoft operating systems (DOS and WindowsNT) and tools (e.g. Access, Excel). The system is layered according to the accelerator controls “standard model”: a supervisory layer, consisting of PCs running console application software, and a front-end (or equipment control) layer, also based on PCs, which interfaces to the equipment hardware and runs the equipment specific control software. The integration of the different (about 30) PCs is realised with a in-house, high level communication protocol over TCP/IP and Ethernet. The interface to the equipment (e.g. power supplies) is not homogeneous but is a combination of old standards, such as CAMAC, special PC I/O cards and other ad hoc solutions. A detailed description of the ISOLDE control system is available in [4]. Without discussing its technical merits, the old ISOLDE control system is a unique implementation (no element in common with the PS control system). It is also technologically obsolete and its exploitation and maintainability are a very serious concern. Hence two key features of the design of the new ISOLDE controls: compatibility (with the PS control system) and use of industrial components (reliable, off-the-shelf technology).

2.2 The new front-end layer

A complete description of the new ISOLDE front-end controls is available in [5]. For some ISOLDE equipment, in particular the beam instrumentation, the equipment control layer will be implemented using the standard PS controls solution: the Device Stub Controller (DSC) [6]. This is a VME based system including a processor, running the LynxOS operating system, and various VME interface modules for the I/O towards the equipment. It is well adapted to buses such as GPIB or the MIL1553 standard.

For the rest of the equipment we have devised a solution based on the conclusions of a previous study [2], which systematically applies industrial components. The main ingredients are:

- The Device Stub Controller (DSC): this acts as a communication concentrator for the PLCs for equipment whose control is implemented with industrial components. The DSC connects a PLC based sub-system to the PS control system by translating the equipment control requests (from the console layer) into a simple control protocol (towards the PLC layer). The DSC glues together different environments: the supervisory layer, the service infrastructure (e.g. the databases) and the control process.
- The Programmable Logic Controller (PLC): the task of the control process is delegated to a PLC. One PLC is dedicated to a number of instances of equipment of the same family (such as a particular type of a power supply), and implements the local control process as well as maintenance functions. A dedicated PC or a simple display module may be installed to

operate the PLC locally, when offline from the control system supervisory layer.

- PROFIBUS [7]: this is the field bus for the standard industrial I/O modules (e.g. ADC, DAC, etc.) which connects to the equipment on either copper wires or an optical fibre (for example in the case of equipment running in a high voltage environment). While this is not the case at ISOLDE, a PLC controlling some complex equipment could delegate part of the control to a hierarchy of PLCs on PROFIBUS.
- Ethernet: it is used as the fieldnetwork to connect the DSC to the PLCs. This choice greatly simplifies the connection of our VME processor to the PLC. It allows us to benefit from the general CERN network infrastructure and support, and it opens the door to the use of e.g. Web-like applications. It also lifts the constraints on the frame size (i.e. the PROFIBUS limitation to 256 bytes/frame) and the communication paradigm (i.e. the PROFIBUS mailbox-like protocol).
- Protocols: given the selection of Siemens hardware, we run the RFC1006 [8] protocol on top of the Ethernet fieldnetwork. We also run the Siemens Fetch/Write protocol [9] to remotely execute load/store operations in the PLC memory.

In summary, the new control system of the ISOLDE facility includes two configurations for the equipment control layer: one is the standard, VME based DSC, while in the second, PLCs with industrial I/O modules on PROFIBUS are systematically used. In this latter the control process has been migrated from the DSC to the PLC; the former plays the role of link to the rest of the control system and its facilities. The main ingredients defined above are put together in figure 1, which depicts the layout for ISOLDE equipment controlled via industrial components.

We have designed and developed a simple client-server protocol, on top of RFC1006, to implement the generic communication interface between the DSC and the PLC.

The PLC software includes a small generic component that serves requests from the DSC and activates the relevant equipment specific threads of code. In the spirit of keeping the PLC programs as simple as possible, the state of the equipment is maintained by the DSC, so as to exploit all the facilities (such as dynamic data tables) which are available in the PS control system. A pictorial view of the organisation is shown in figure 2.

The sharp boundary between the generic (communication) part and the equipment specific control process greatly simplifies both the task of the software developer and the communication protocol. This latter is reduced to the exchange of a data structure.

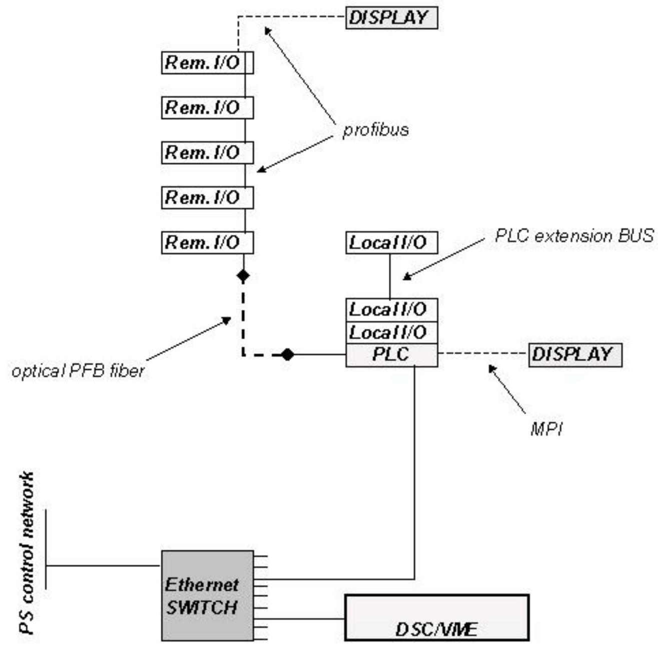


Figure 1: Generic PLC based control layout.

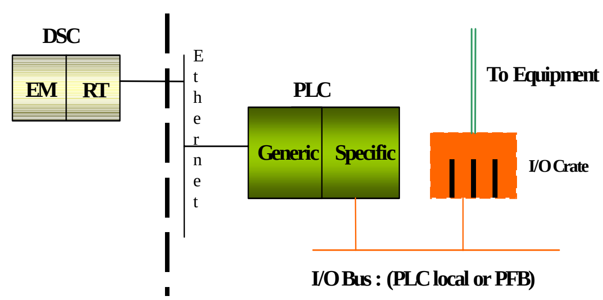


Figure 2: Organization of the equipment control layer.

CERN recommends a number of industrial components standards, products and manufacturers. We have chosen PROFIBUS as the I/O field bus and Siemens products for both PLCs and I/O modules. Depending on the required performance and I/O channel density, we have used either the SIMATIC S7-300 or the SIMATIC S7-400 series PLCs and I/O modules. The PLC software is developed using the Pascal-like SCL language. The choice of one brand does not prevent the use of other industrial components, however. Field buses and field networks allow different components to inter-operate, indeed we foresee to complement our architecture with other CERN recommended products.

As an example, take the system for the control of ~300 power supplies for the ISOLDE electrostatic lenses. They will be controlled by a SIMATIC S7-400 PLC based system. We currently estimate that a single S7 414-2 PLC will be enough to control the ~300 power supplies. Industrial I/O modules (ADC, DAC etc.) are accommodated in 3 segments of 4 crates each using the PLC extension bus. Equipment specialists will use a local display module for local configuration and diagnostics as well as to perform local control.

The status of the developments is well advanced: a few sub-systems will be installed at ISOLDE during fall, and if successful left in operation. The bulk of the new front-end layer will be installed during the 2001-2002 winter shutdown.

3 CONCLUSIONS

We have systematically used, where applicable, industrial components in the design of the new ISOLDE control system. The benefits of such an approach are extensively discussed in [2], and include flexibility in the design and development, easy assembly of off-the-shelf components, and tracking the evolution of industry. The approach is well adapted to a wide range of equipment: the power supplies, the stepping motors, the gas system and the vacuum system. For some equipment, in particular where the acquisition of a sizeable amount of data is concerned or the equipment is interfaced via the GPIB bus, we have retained the standard PS controls solution.

We have merged a generic solution based on industrial components with the existing PS control system: on the one hand we made use of the powerful tools already available, while on the other hand we have provided an additional building block to the PS control system. Indeed the solution we have devised is not limited to ISOLDE. The door is open for its use elsewhere at the PS complex, in particular for applications where the PS pulse-to-pulse modulation [6] is not an issue and to the controls of future CERN projects such as the LEIR and CTF3 machines.

The installation of Ethernet as a fieldnetwork was easy and it is very well adapted to a horizontal (device oriented) control system (as opposed to a vertical, channel oriented, system where the fieldbus is better suited). It is however more expensive, the cost of the Ethernet being of the same order as the cost of the PLC.

Finally there are two open questions, both related to the use of Ethernet as a fieldnetwork. The first concerns security: how to regulate the external access to the PLC sub-system, in particular if user interfaces based on an embedded Web server are used. The second is related to the use of a PLC sub-system in the PS environment; here the issue is deterministic (real-time) behavior, the order of magnitude being a few tens of milliseconds. While the PLC per se is deterministic[1], the same is not true for the delivery of e.g. timing signals over Ethernet. We propose to overcome the problem by using a structured deterministic (i.e. collision-less) Ethernet, an issue where industry is currently very active.

4 ACKNOWLEDGEMENTS

It is a pleasure to acknowledge the support of the controls group of the CERN PS division, in particular its leader B. Frammery, as well as colleagues from the ISOLDE equipment groups: PS/PO, PS/OP, PS/BD and LHC/VAC.

REFERENCES

[1] E. Kugler. The ISOLDE facility. Hyperfine Interaction 129 (2000) 23-42.
[2] O. Jonsson, F. Locci, G. Mornacchi. “ISOLDE Front-End Study, Final Report”. CERN int. PS/CO Note 2001-009 (Tech.).
[3] O.C. Jonsson et al. “The evolution of the ISOLDE control system”. NIM B, 126 (1997) 30-34.
[4] F. Locci. Le système de contrôle ISOLDE et REX”. CERN int. PS/CO Note 99-28.
[5] J. Allard, O. Jonsson, F. Locci, G. Mornacchi. “Consolidation of the ISOLDE controls”. CERN int. PS/CO Note, to be published.
[6] http://wwwpsco.cern.ch/.
[7] http://www.profibus.com.
[8] http://www.faqs.org/rfcs/rfc1006.html.
[9] SIMATIC CP 1430 TCP Coupler. Reference Manual. Siemens AG.

[1] Provided the PLC is capable of executing the cycle within the required time frame.

TUAP037
cs.OH/0111005

AUTOMATED REAL-TIME TESTING (ARTT) FOR EMBEDDED CONTROL SYSTEMS (ECS)

Jon Hawkins, Group Leader, Interlock System & Instrumentation Group

Haung V. Nguyen, Engineering Assistant, Interlock System & Instrumentation Group,

Reginald B. Howard, Test Engineer Consultant

Advanced Photon Source, Bldg 401, L1111, 9700 S. Cass Ave, Argonne, IL 60439

Key Words: Testing Embedded Systems, Real-Time Testing, Automated Testing, Personnel Safety Systems

Abstract

Developing real-time automated test systems for embedded control systems has been a real problem. Some engineers and scientists have used customized software and hardware as a solution, which can be very expensive and time consuming to develop. We have discovered how to integrate a suite of commercially available off-the-shelf software tools and hardware to develop a scalable test platform that is capable of performing complete black-box testing for a dual-channel real-time Embedded-PLC-based control system (www.aps.anl.gov). We will discuss how the Vali/Test Pro testing methodology was implemented to structure testing for a personnel safety system with large quantities of requirements and test cases.

This work was supported by the U.S. Department of Energy, Basic Energy Sciences, under Contract No. W-31-109-Eng-38.

1 INTRODUCTION

Many of today's automated real-time testing systems for embedded systems were developed using expensive custom hardware and software. In this article we describe how to use commercially available off-the-shelf hardware and software to design and develop an automated real-time test system for Embedded Programmable Logic Controller (PLC) Based Control Systems. Our system development began with the implementation of the VALI/TEST Pro testing methodology as a means for structuring the testing. Using this methodology, we were able to decompose system requirement documents for a Personnel Safety System (PSS) into its high, intermediate, and detail level requirements. Next, the validation procedures for the PSS system were decomposed into testing units called builds, test runs, and test cases. To measure the PSS system's test coverage, three levels of system requirements were mapped to their respective unit level of test using a specially constructed validation matrix that was designed to handle over 150 test cases and requirements. All of the above work led to the development of an Automated Real-Time Test System (ARTTS) that is capable of performing complete black box testing in real-time for Embedded PLC Based Control Systems. Also note, that the PSS system under test and mentioned in this paper is located at the Advanced Photon Source (APS) at Argonne National Laboratory Basic Energy Science Facility in Argonne, Illinois (www.aps.anl.gov).

2 PSS SYSTEM OPERATION

In this section, we explain the theory of operation of a Personnel Safety System (PSS) for which the prototype ARTTS System was designed to test.

2.1 PSS System Description

Consider figure 1. It depicts a typical configuration of "Station A", which is a major component of the overall PSS system. Note that the overall function of a PSS system is by definition, a highly reliable, fail-safe, redundant, stand-alone system that closely monitor and control personnel access into potentially hazardous Experimental Stations. It is also responsible for reducing hazards to mitigate harm to personnel against direct X-ray radiation from the Advanced Photon Source.

Fig. 1 Personnel Safety System (PSS)

2.2 PSS "Station A" User Panel Operation

Depicted in figure 2 is a typical layout of "Station A" user panels. Note that "Station A" consist of three panels 1) Station A user panel, 2) Station A door panel, and 3) System Controller panel.

Fig. 2 PSS User Panels

3 TESTING METHODOLOGY

In this section, we describe how the PSS system requirements and validation procedures were structured using the Vali/Test Pro methodology. Note that the Vali/Test Pro is a methodology developed by Interim Technology Company and is widely used in the testing industry as a means to provide a visual approach to validating requirement coverage of a system or software application. This methodology provided us with the means to decompose PSS system requirement documents (SAD, DOE, ES&H) into their high, intermediate and detail levels. In addition, we decomposed the validation procedures for one PSS beamline into testing units called builds, test runs, and test cases (see figure 3). Afterwards, we mapped each requirement level to its respective testing unit in a validation matrix in order to measure the test coverage.

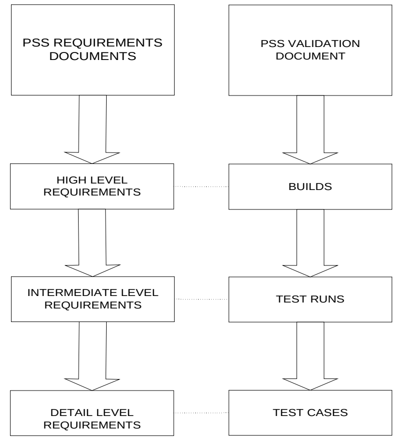


Fig. 3 Vali/Test Pro Testing Methodology

A validation matrix for each level of test requirements was constructed to map each level of requirements to its respective level of test. Below we define the different requirement levels and testing units used to defined each type of validation matrix.

- High level requirements: They represent the broadest categories of a system functions, such as business processes, including data entry and data capture, normal and exception processing, and database updates.
- Intermediate level requirements: These are components of the high-level requirements, often at the transaction or batch processing level. They may include sequences of actions to be taken, information to be displayed, error routines and associated messages.
- Detail Level Requirements: They are the specific steps in application processing, such as action steps field definitions, edit criteria, calculations, and error messages.
- Builds: A logically complete subset of an application that can be tested independently, and then integrated and tested with other builds.
- Test Runs: They consist of a set of related test cases that are used to validate the requirements at the intermediate level.
- Requirement Validation Matrix: A table that provides a cross-reference between a system's requirements and test case specifications.

4 AUTOMATED REAL-TIME TEST SYSTEM (ARTTS) DEVELOPMENT

In this section, we described how the ARTTS system was developed as a tool to automate the testing of the PSS system described above. The development of the

ARTTS system described here consists of three major areas of system integration: 1) Test Requirements, 2) Software Tools, and 3) System Hardware. Below we explain how the use of commercially available off-the-shelf software and hardware was integrated to perform all required testing for the PSS system.

4.1 Test Requirements

The Department of Energy regulations require the personnel safety system for each beamline to be validated every 6 months. In addition to the high level matrix described in section 3.0 there are an intermediate level and detail level matrices that were required to completely define the PSS system requirements.

4.2 Reasons to Automate PSS Testing

One reason to automate the PSS system testing is an increase in the need for additional validations over the past five years. Another reason to automate PSS testing is a steady increase in requests for software changes. Additional reasons to automate the PSS system are listed below.

- Manual testing of the PSS takes up to 3 days to validate one beamline depending on its configuration.
- An automated testing system will catch more software errors before final testing is performed on the lab floor.
- It will reduce the overall time required to validate a beamline.
- Since 98% of all software change requests (SCR's) are related to HMI and not safety, most of the testing performed on the lab floor can be moved to the ARTTS simulator. As a result, it will increase software reliability.
- It will reduce the cost of testing.

4.3 Software Tool Integration

Fully automated, the testing of a PSS system required the integration of four software tools onto a single computer platform along with a fifth tool that was set up to work on a separate notebook computer (see figure 8). Together the software tools provided the following functions: 1) Real time I/O simulation for programmable logic controllers, 2) Human Machine Interface (HMI) to user control panels, 3) Graphical User Interface (GUI) testing for Windows based applications, 4) Graphical test planning, batch testing and defect tracking, 5) PLC fault code verification for Allen Bradley PLC's, and 6) PLC fault code verification for GE PLC's.

The software tools selected to perform the above system functions were as follows:

- Programmable Industrial Control Simulation (PICS) software provided a real-time HMI and I/O simulator for testing PLC-based control system.
- WinRunner software provided a functional testing tool designed primarily to test graphical user interfaces of Windows base applications. It was used here in the ARTTS system to test and verify the state of the PSS system's I/O.
- Test Director software module provided graphical test planning, batch testing, defect tracking, and an interface to the WinRunner module.
- RSLogic is an A/B tool designed primarily to down load software to A/B's PLC's and to verify faults and task states in Chain B of the PSS system.
- State Logic is a GE software tool that provided the means to verify faults and task states in Chain A of the PSS system.

4.4 ARTTS Hardware

The ARTTS system's hardware is best described by the system hardware layout in figure 4. The following hardware makes up the ARTTS prototype system:

- Gateway GP7-500 – 500 MHZ CPU, 1GB RAM, 20 GB HD IDE, Vision Tek video card, A/B 1784KT card, AB-5236 SD, 10/100 MB/s Network Interface Cards
- Nokia 17 inch Monitor
- Allen Bradley PLC-5/30 Rack: A/B 120 VAC power supply, DH+A, DH+B, A/B PLC-5/30 CPU
- GE Fanuc Series 90-70 Rack: series 90-70 power supply, State Logic CPU, Ethernet interface
- Unicom Dyna-net/4 Null Hub

Fig. 4 ARTTS Hardware

4.5 ARTTS System Schematic

A schematic with all the ARTTS system components integrated together is shown in figure 5 below.

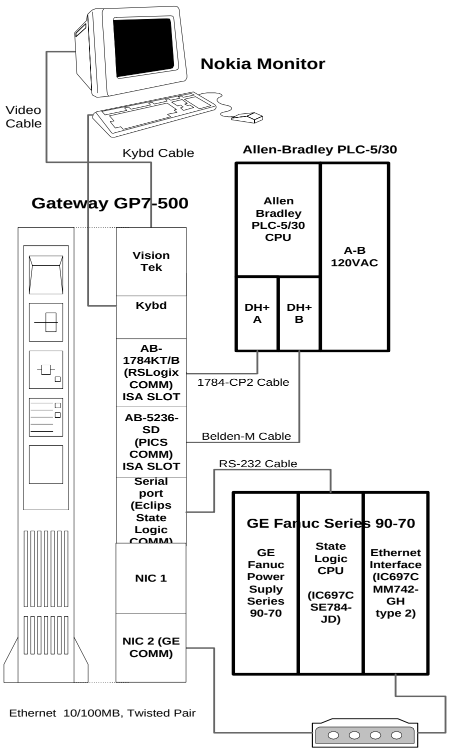


Fig. 5 ARTTS System Schematic

5 ARTTS OPERATION

5.1 Human Machine Interface (HMI)

A user interface to the ARTTS system is via a computer keyboard, mouse, and color screen computer monitor. Using PICS HMI graphical user interface, a user can control and monitor all functions of the PSS system by activating the proper switches on the user control panel and then monitoring the system function via LED outputs (see figure 6).

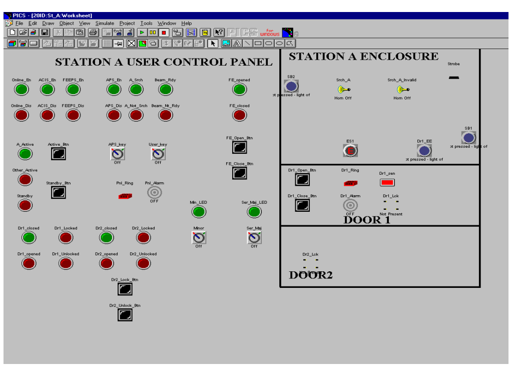


Fig. 6 ARTTS HMI Panel

5.2 Test Planning

A user can set up or activate batches of test to run by selecting test plan from TestDirector's menu. Once selected a batch of tests can be run by simply selecting Run Test from TestDirector's menu. This step automatically activates WinRunner to execute the selected test scripts (see figure 7).

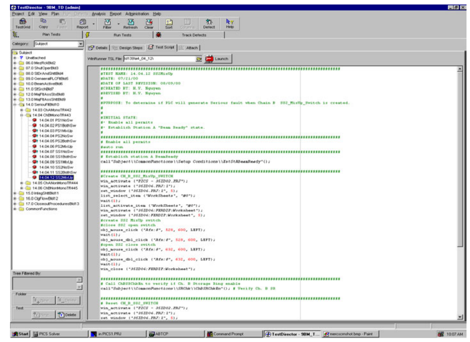


Fig. 7 ARTTS Test Planner

5.3 Test Results

Executed tests results are verified within Test Director by selecting the icon labeled "Details" from the menu.

The results of each test will be listed as pass or failed along with the date and time of execution of each test (see figure 8).

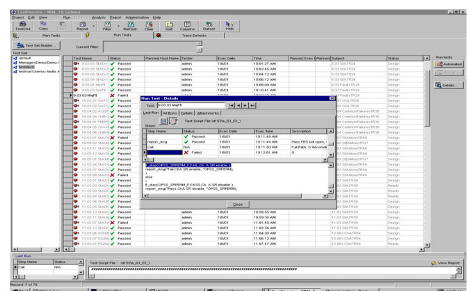


Fig. 8 ARTTS Test Results

6 ARTTS SYSTEM PERFORMANCE

The ARTTS system was very successful reducing the amount of time required to perform PSS testing. For example, the ARTTS system reduced the overall amount of time required to process a batch of 72 test cases from about 4 hours of manual testing to 1 hour and 36 minutes of automated testing. In general, the ARTTS system achieved an execution time of 1 minute and 20 seconds per test case. Furthermore, as a result of additional test coverage the ARTTS system greatly increased the reliability of the PSS software.

7 ARTTS SYSTEM ADVANTAGES VS DISADVANTAGES

7.1 Advantages

- Scalable.

- Major reduction in time required to validate a batch of test cases.
- Commercially available software.
- Commercially available hardware.
- Reduced development time.
- Easy to build.
- Better test coverage.

7.2 Disadvantages

- Uses 5 different software packages.
- Configuration management is more a challenge with multiple software packages

BIOGRAPHIES

Jon Hawkins, Group Leader
Argonne National Laboratory
Interlock Systems & Instrumentation Group
Bldg 401, Rm C1249
Argonne, IL 60439, USA
hawkins@aps.anl.gov

Jon K. Hawkins is a Group Leader at the Advanced Photon Source, Argonne National Laboratory. He is responsible for the design, testing, installation and validation of real-time embedded computer based mission critical systems. Mr. Hawkins is a senior engineer, whose experience ranges from high-speed analog and digital system design with full custom ASIC's to computer based system requirements/risk analysis. Mr. Hawkins holds a US patent, has been presented the Argonne National Laboratory Directors Award and has published numerous articles in technical journals.

Reginald B. Howard, P.E.
Argonne National Laboratory
Interlock Systems & Instrumentation Group
Bldg 401, Rm L1111
Argonne, IL 60439 USA
rhoward@aps.anl.gov

Reginald B. Howard is a test engineer consultant at Argonne National Laboratory while working for an IT consultant firm named Web Business Controls, Inc. He received his BS degree (1987) in electrical engineering from Old Dominion University. He has been working at Argonne National Laboratory for the last 2 years assisting with developing the automated real-time testing system for embedded control systems. In addition, he is responsible for developing and implementing a web based electronic documentation system for the Laboratory. Previously he worked as an electrical engineer with the Federal Aviation Administration (FAA) where he was responsible for developing the FAA's first touch screen control system for airport runway approach lights control systems at O'Hare International Airport Air Traffic Control Tower (ATCT). In addition, he has designed electrical distribution systems and lightning protection systems for ATCT facilities. Furthermore, he has worked in the manufacturing industry where he designed and tested electrical power conditioning products. Mr. Howard is a licensed professional engineer in the state of Wisconsin and is a member of the IEEE Computer Society and "The Instrumentation, Systems and Automation Society" (ISA).

Haung V. Nguyen
Argonne National Laboratory
Interlock Systems & Instrumentation Group
Bldg 401, Rm B1196
Argonne, IL 60439 USA
nguyen@aps.anl.gov

Van Nguyen is an Engineering Assistant at Argonne National Laboratory (ANL). He is also an undergrad computer science student who is planning to pursue a graduate degree in computer science at Illinois Institute of Technology (IIT). While working at ANL he has been responsible for testing Personnel Safety Systems (PSS) and responsible for maintaining Programmable Logic Controller (PLC) software for PSS systems. Furthermore he has been involved with developing and writing software tests scripts for the Automated Real-Time Test System (ARTTS).

REFERENCES

1 Interim Technology, "Vali/Test Pro-A Validation and Testing Process" Ver.1.1. pp. 6.1 –7.16

TUAP042
physics/0111161

THE NLC SOFTWARE REQUIREMENTS METHODOLOGY

G.R. White, H. Shoaee, SLAC, Stanford, CA 94025, USA

Abstract

We describe the software requirements and development methodology developed for the NLC control system. Given the longevity of that project, and the likely geographical distribution of the collaborating engineers, the planned requirements management process is somewhat more formal than the norm in high energy physics projects. The short term goals of the requirements process are to accurately estimate costs, to decompose the problem, and to determine likely technologies. The long term goal is to enable a smooth transition from high level functional requirements to specific subsystem and component requirements for individual programmers, and to support distributed development. The methodology covers both ends of that life cycle. It covers both the analytical and documentary tools for software engineering, and project management support. This paper introduces the methodology, which is fully described in [1].

1 BACKGROUND AND OUTLINE

The estimated budget for NLC controls is about $0.5bnUS. The software development effort and support infrastructure is about $75m, and represents ~475 person years of effort. However the funding calendar requires that our staffing is not constant over the development period.

Additionally, the software controls effort is likely to be distributed and collaborative, involving EPICS developers and other participating institutions. Therefore a formal methodology has been developed which we hope will help to: 1) communicate future intent and alternatives, 2) define the "treaty points" between sub-systems, and the interfaces between sub-systems, and 3) help designers delineate how sub-systems relate to the network architecture and data infrastructure (such as data acquisition, control, database and archiving).

The NLC controls project also has a very long timeline and the objectives of our methodology change significantly over time. In the short term our objective is to effectively estimate the software and infrastructure requirements for the purposes of cost estimation and project scheduling. In the long term our goal shifts to producing requirements suitable for producing accurate and clear designs. This methodology is designed to help on both fronts. It separately outlines development and management methods, and is based on a combination of systematic structured methodologies [2][5], in particular the Rational Unified Process, but incorporates elements from systemic methods [4][7]. "Use Cases" (part of the Unified Modeling Language) are used to delineate user level requirements [6], but also significant emphasis is given to the handling of non-functional requirements.

2 METHODOLOGY FOR LARGE SYSTEM DEVELOPMENT

Clearly defined documentation deliverables are itemized and described explicitly by the methodology. Table 1 summarizes those deliverables produced for each sub-system, application or component library; table 2 itemizes those for project management.

Table 1: Development Effort Deliverables

Artifact	**Description**
Vision Document	Overall summary of purpose and desired features
Discussion Docs	Textual discussion of Concepts
Domain Model	Use Case Model: Diagrammatical and **Textual** description of Use Cases (that is, scenarios of use from a functional perspective). Entity-Relationship diagrams. Algorithms
Developers Guides	Continuously updated programmers documentation as systems take shape.
User Interface Model	User interface mock-up
Use Case Package Report:	Diagrams of relationships of packages and subsystems
Non-functional requirements Document	Textual description of system constraints and references
Conflict Matrix	How conflicting data acq or control requirement will be handled.
Taxonomy	Summary of key functional attributes
Requirements Matrices	RequisitePro db of requirements
Cost Benefit Analysis	Description of Cost, Effort and functionality alternatives and estimates

The methodology distinguishes between functional requirements (which describe the desired behavior of

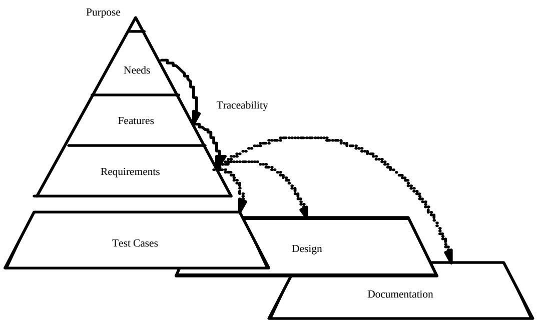


Figure 1: Hierarchy of Requirements

the system) and non-functional requirements (those concerned with the system's effective implementation). All requirements are viewed hierarchically; with very general requirements such as the purpose and goals at the top, followed by software features and then more specific, possibly testable items and "system requirements". As such, each requirement may be traced from its antecedents to its consequents (see Figure 1).

Each requirement is assigned an explicit description type, such as NEED, testable REQuirement, and so on (though absolutely correct classification is not necessary). For a very large distributed project, the role and specificity of *non-functional* requirements is particularly pronounced because these typically define the interfaces and shared resources of sub-systems. Table 3 lists some examples of these very specific non-functional requirements.

To each requirement are attached some "attributes" (different attributes depending on the requirement type) such as benefit, effort, cost, time, priority, risk of overrun etc, used for cost/benefit analysis.

One important attribute is obviously the level of "benefit" – captured as attribute values COULD, SHOULD and MUST. Additionally though we predict that in a large project there will be a large number of alternative requirements for alternative solutions proposed, each impacting the requirements of associated systems and changing their cost, time estimates, and level-of-benefit. Therefore the methodology proposes the documentation of "alternative requirement sets" to manage the treaty points between systems, their priority, cost alternative solutions, and to minimize friction and scope creep (see Figure 2).

To manage all this formal documentation we use a requirements management tool from Rational Software Corp. called RequisitePro, which is basically a database application. Each requirement in a document is captured using document parsing software. The resulting database of requirements can be queried, visualized, and analyzed with the usual drill-down and slice-and-dice paradigms [3].

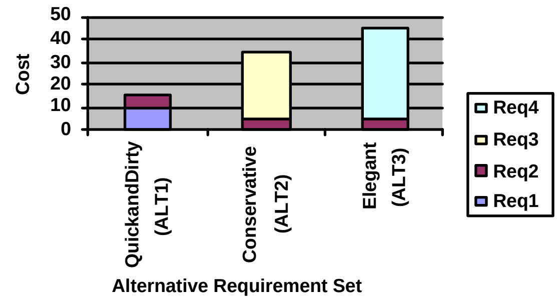


Figure 2: Comparison of Alternative Requirement Sets

Table 2: Global Deliverables

Artifact	Description
CDR summary	Summary of requirements work breakdown
WBS	Work Breakdown Structure
Basic Design Principles	Statement of intent about the core architecture, infrastructure, and design of systems.
Sizing and Performance Specification	Diagrams and text showing nodes, data flow, and timing requirement
System Boundary Analysis	Summary of System Boundary and Treaty Points
Domain Analysis	Document and Diagrams detailing domain objects and their relationship
Enterprise db Schema	Description of control system entities and their relations
Glossary	A dictionary of terms.
References	Pointers to literature, both accl. Physics and Engineering and programming

Table 3: Non-functional Requirement Types Traced

Abbrev.	Description
XREF	Requirement of another system.
UC	A description of a use of the system, described from an actors point of view.
ALT	An alternative NEED, or FEAT, and REQ set which can satisfy the purpose.
PERF	Performance Requirement.
DBNAME	A database element, being a collection of attributes. This may describe a device, or some other aggregate.
TESTC	A test suite, aimed at testing a Feature.

The global requirements analysis includes the construction of a "conflict matrix", which attempts to define the needs of each control subsystem (RF, feedback, beam monitor control etc) regarding their priority in contention situations with other systems. In the NLC this will be used to define contention resolution rules.

The methodology includes a taxonomy for the key control system characteristics of each major subsystem, see Table 4.

Table 4: Controls Subsystem Taxonomy

Property	**Description**
Initiation and Termination	How is the system started and terminated.
Duration	For how long is the system typically active and not idle
Frequency	How often is it initiated.
Read/write	Is the process mainly a reader or writer
Running State	What machine state is this system for?
Shareable input	Is the pulse related input data of this system shared by other systems.

Data handling and network performance requirements are driven by data throughput and processing requirements. In order to accurately predict what those will be we constructed a “data sizing” taxonomy, including volume, frequency, latency, resilience, structure etc, which helps to identify the different data handling needs of each subsystem which is involved with the same data. For instance, the data handling need of a monitor display are very different to those of the monitor data acquisition system, or the monitor data archive system. The result is a logical network and storage architecture.

We propose not to have monolithic design, followed by development phases, as in the traditional “waterfall” approach. Rather we suggest an iterative, feature driven approach, with fixed 6-week periods in which we decide what features will be added or refined in each period. More analysis and design of each feature will be done in early iterations of the overall project, more programming and testing in later iterations. This is a formalized version of “extreme programming” which has recently become popular.

Over a long development period important design ideas and other incidentals, which cannot be pursued in the immediate term, are documented and tracked specifically as “forward references”.

3. IMPLEMENTATION AND RESULTS

A formal systematic methodology was not used before in the SLAC controls community, though we had in the past applied the traditional approach very effectively, and design reviews and code standards were enforced. Although we have not started the NLC controls effort yet, we have adopted parts of this methodology for the existing B-factory, and our effort to migrate the existing control system to new technologies (i.e., off VMS). In particular the formal capture of requirements into a db, the use of distinct vision, requirements, and discussion documents, forward references, iterative fixed period development, and continuously developed programmers documentation have all been implemented successfully and have increased productivity. We have also started data sizing the NLC, and created the taxonomy of subsystems. Some lessons have been learnt, and refined our plans for the methodology, which have been reflected in this paper. In particular, we had proposed a very structured workflow, with identified roles for each job function. That will probably be removed from the methodology user guide, which is available in [1].

This paper has introduced the major concepts of the methodology we developed. We would very much like to hear feedback on the methodology presented, and experiences of formal methodologies in use at other accelerator software departments.

REFERENCES

[1] Software Requirements and Analysis Methodology Recommendations for NLC Controls. White G. R., Shoaee, H. SLAC ESD Software. Available from www-group.slac.stanford.edu/cdsoft/nlc_arch.html

[2] Rational Unified Process (RUP). Rational Software Corp. http://www.rational.com

[3] Rational RequisitePro, Rational Software Corp. http://www.rational.com

[4] General System Theory. Bertalanffy, Ludwig von, Brazziler, George. 1968

[5] Requirements Engineering and Rapid Development, Graham, Ian., Addison-Wesley 1998

[6] Object Oriented Analysis and Design. Liberty, Jesse. Wrox Press 1998

[7] Designing Systems for Business Success Mumford, Enid. Manchester Business School. 1986.

[8] Object Management Group (OMG) UML resource center, www.uml.org.

TUAP046
physics/0111102

TIMING SYSTEM OF SPRING-8 BOOSTER SYNCHROTRON

N. Hosoda, T. Ohshima, T. Aoki, T. Asaka, H. Ego, K. Fukami, Y. Kawashima, Y. Ohashi, T. Takashima, H. Yonehara, SPring-8, Hyogo 679-5198, Japan

Abstract

The timing system of SPring-8 booster synchrotron generates various timing signals concerning beam injection, acceleration from 1 GeV to 8 GeV and ejection. We have improved the timing system of the synchrotron giving it better stability and flexibility. This improvement results the other merits of advanced operations, for example, storing electron beam in the synchrotron, changing the injection cycle from 1 Hz to the slower frequency to increase the RF knock-out (RF-KO) operation period and ejecting the low energy beam during ramping up.

1 INTRODUCTION

The SPring-8 facility consists of a 1 GeV linac with a repetition rate of 60 Hz, a booster synchrotron and an 8 GeV storage ring. The timing system installed in the RF station named E-station in the storage ring generates a 508.58 MHz RF reference signal and a 1-cycle signal at the rate of 1 Hz. The 1-cycle signal includes the information of an RF bucket address in the storage ring. The timing system of the synchrotron receives these signals and regenerates many timing signals such as a gun trigger, pulse magnet triggers and ramping patterns. For a single-bunch beam operation of the storage ring, an RF-KO system installed in the synchrotron rejects the electrons in satellite RF buckets around the main bucket [1].

2 REQUIREMENTS

There are four requirements for the timing system in the synchrotron.

The first is to synchronize the 508.58 MHz RF phase between the synchrotron and the storage ring to control the beam filling of storage ring.

The second is to suppress the time jitters between the gun trigger signal of the linac and the RF signal of the synchrotron as small as possible less than 100 ps. The time width of a beam from the linac for a single-bunch beam operation is 1 ns, and the time width of an RF bucket of the synchrotron is about 2 ns. If the energy spread of a beam is $\Delta E/E \geq 1$ % and the gun trigger timing changes over 100 ps, the portion of the beam is not captured by an RF bucket of the synchrotron and the beam intensity of the synchrotron fluctuates. During the single-bunch beam operation, the synchrotron radiation users request the uniform beam intensity of each bunch of the storage ring. To realize it, the beam intensity from the synchrotron must be stable.

The third is to prepare the function to inject multi-pulse beam from the linac into the synchrotron and to inject into an RF bucket of the storage ring in 1 cycle to shorten beam-filling time in the storage ring. The harmonic number of the synchrotron is 672 and that of the storage ring is 2436. While the RF bucket of the synchrotron turns around 29 times, the RF bucket of the storage ring turns around 8 times. The RF bucket addresses of the synchrotron relative to that of the storage ring are summarized in Table 1.

The fourth is to add the flexibility of the advanced operations of the synchrotron such as storing the electron beam in the synchrotron, changing the injection cycle from 1 Hz to a slower frequency to increase the RF-KO operation period and ejecting the low energy beam during ramping up.

The timing system already satisfies the former three requirements. However, to satisfy the former two requirements with the higher precision, we improved the RF low power system last year. In addition, to improve the function to inject multi-pulse beam and to add the flexibility, we improved the timing system this year.

Table 1: RF bucket addresses of the synchrotron relative to that of the storage ring. The delay counts of M for two 508MHz SUC's are mentioned in section 3.2.

Synchrotron revolution	Synchrotron RF bucket address	Storage ring revolution	M for 508MHz SUC with N=672	M for 508MHz SUC with N=2436×8
0	0	0	0	1
3	420	1	420	2436
7	168	2	168	4872
10	588	3	588	7308
14	336	4	336	9744
18	84	5	84	12180
21	504	6	504	14616
25	252	7	252	17052

3 THE SYSTEM

3.1 Optic Fiber Link

All timing signals are linked in the timing system of the synchrotron located at the injector control room. We use two kinds of optic fibers. One is the single-mode phase-stabilized optic fiber supplied by Sumitomo Electric Industries Ltd. [2]. This is used for the distributions of 508.58 MHz RF reference signal and precise timing

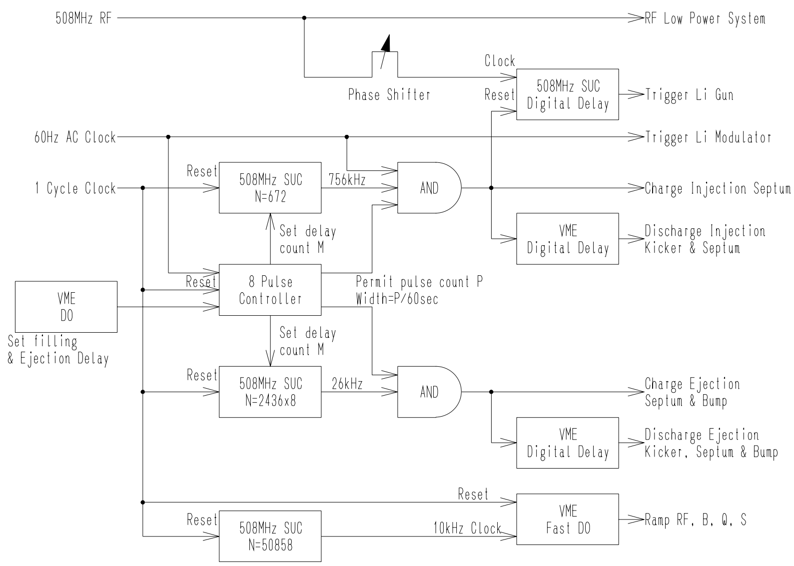


Fig. 1: Block diagram of the timing system of the synchrotron.

signals such as triggers for an electron gun and fast beam monitors. The RF reference signal from the storage ring is controlled under a phase-locked loop system (PLL) [3]. In addition, the RF reference signal to the RF low power system employs the PLL, too. The other one is the multi-mode graded-index optic fiber with core/cladding size of 50/125 μm. This is used for the distributions of triggers for magnets and slow monitors. The measured time jitters of this optic fiber system with a fiber length of 250 m is 440 ps as a standard deviation.

3.2 508MHz SUC

We can correctly count the RF bucket number using 508.58 MHz synchronous universal counters (508MHz SUC) [4]. The 508MHz SUC generates a clock with a frequency of 508.58/N MHz and a delay time of M/0.50858 ns from a counter reset signal. The N and M are positive integers.

Fig. 1 shows the block diagram of the timing system of the synchrotron. We use two 508MHz SUC's to control multi-pulse beam. One is to define the injection timing and is set N = 672. The other is to define the ejection timing and is set N = 2436×8 = 19488. The delay counts of M for two 508MHz SUC's are also summarized in Table 1. We set the beam filling of the synchrotron and the ejection timing to the 8-pulse controller as shown in Fig. 1 by using a VME digital out (DO) board. This controller controls the number of pulse and the delay count M's of two 508MHz SUC's for each pulse.

A 508MHz SUC is used for a gun trigger signal. If the trigger signal has some time jitters, the 508MHz SUC synchronizes the trigger signal with the RF reference again. Therefore, jitters smaller than 2 ns is completely cured. A phase shifter in front of this 508MHz SUC adjusts finely the timing of beam injection.

To generate the ramping patterns of the main magnets and the acceleration voltage (Vacc) using the VME fast DO [5], a 10 kHz clock is also generated by a 508MHz SUC.

3.3 VME Digital Delay

To discharge the pulse magnets such as kickers and septums for beam injection and ejection, we adopt VME 6 channel digital delay boards, Berkeley Nucleonics Corporation B951-2. The measured maximum time jitters at maximum delay of 167.8 ms is 0.9 ns. For multi-bunch beam operation, the time width of a beam from the linac is 40 ns and the kicker magnets have 60 ns flattop pulse width. The jitters is negligible.

3.4 Phase Control of RF Signal

The phase control part in the RF low power system of the synchrotron is shown in Fig. 2. We have two klystrons and each output power is provided into four 5-cell RF cavities. To make the beam acceptance larger, we control Vacc by changing the RF phase between two klystron outputs, keeping the RF voltage of the cavities constant.

The RF signals from the pick-up port of four cavities receiving the RF power from one klystron are combined. The combined signal is used as the input of the Cavity PLL. The combined RF signals are furthermore combined and total RF signal is used as the inputs of the Vacc PLL and the Vacc ALC (Auto Level Control feedback system). The Vacc PLL keeps the phase of total RF signal constant. The VME fast DO board generates the ramping pattern used as the reference signal of Vacc ALC. One of the Cavity PLL's controls a phase shifter using the Vacc ALC output as a reference and the other controls another phase shifter using the Vacc ALC output with reversed polarity. The relation of the phase between two klystron outputs and the Vacc during 1 cycle is shown in Fig. 3.

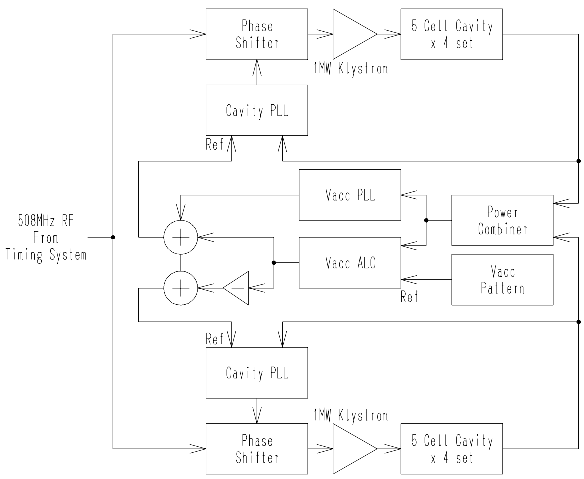


Fig. 2: Phase control part in the RF low power system of the synchrotron.

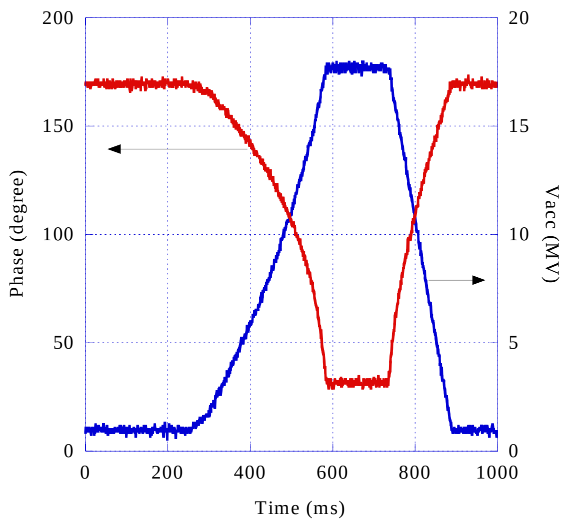


Fig. 3: Relation of the phase between two klystron outputs and the Vacc during 1 cycle.

4 PERFORMANCE

A fluctuation of the RF phase between the synchrotron and the storage ring is less than 0.3 degree by PLL's. We can correctly control the RF bucket position to inject a beam.

The measured time jitters of a gun trigger to the RF signal is 18 ps as a standard deviation. The intensity of the beam with 1 ns time width fluctuates about 5 %, mainly due to other causes.

Fig. 4 shows the electron beam current in the synchrotron during 1 cycle when eight pulses with 1 ns time width are injected and ejected. The Bending Magnet (BM) current is also drawn which represents the ramping pattern of the beam energy from 1 GeV to 8 GeV. We can control the beam intensity by changing the number of injected pulse to the synchrotron in eight steps.

The impurity of the single bunch in the storage ring is defined as the ratio of the number of electrons in the satellite RF buckets to the one in the main RF buckets. It is less than 10^{-9} order or under detection level. When the stored beam current decreased with time, we add the electrons to the same RF buckets. However, the impurity is not changed.

We started the study to eject the low energy beam during ramping up for low energy operation at the storage ring. We are planning to use the synchrotron as a storage ring. The DC operation of magnet power supplies is feasible up to the beam energy of 4.8 GeV. The functions to pause and to restart the ramping are already prepared.

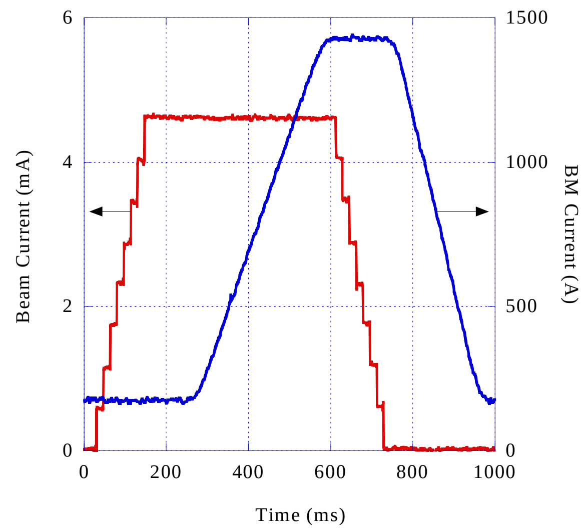


Fig. 4: Electron beam current in the synchrotron during 1 cycle when eight pulses with 1 ns time width are injected and are ejected. The BM current represents the ramping pattern of the beam energy from 1 GeV to 8 GeV.

5 CONCLUSION

The new timing system of the synchrotron has been constructed and confirmed to satisfy our requirements. The character of this system is the flexibility to meet the advanced operation of synchrotron.

REFERENCE

[1] H. Suzuki *et al.*, "Formation of a single-bunch beam in the booster synchrotron at SPring-8", Nucl. Instr. and Meth. A444, (2000) p.515.

[2] S. Tanaka *et al.*, "Precise Timing Signal Transmission by a New Optical Fiber Cable", KEK Report 90-5 (1990).

[3] Y. Ohashi *et al.*, "Performance of the 508MHz Reference and Timing Systems at SPring-8", ICALEPCS2001, San Jose, Nov. 2001.

[4] H. Suzuki *et al.*, "508.58 MHz synchronous universal counter for beam control system of SPring-8", Nucl. Instr. and Meth. A431, (1999) p.294.

[5] N. Hosoda *et al.*, "Integration of the Booster Synchrotron Control System to the SPring-8 Control System", ICALEPCS'99, Trieste, Oct. 1999.

TUAP048

AN EXPERIENCE ON FIXING PROBLEM ON VMEBUS MODULES

A. Akiyama, T. Katoh, K. Kudo, Y. Mori, T. T. Nakamura, J-I. Odagiri, M. Tejima, N. Yamamoto,
KEK, Tsukuba, Ibaraki 305-0801, Japan

Abstract

It is very difficult to fix the hardware errors happen in the complicated system where a common bus and various kinds of modules are used. In the KEKB control computer system, it happened when we put a VME-MXI driver module and a CPU module with Power PC in a subrack of the beam monitor IOC. It took us some time to identify the true source of this problem: a malfunctioning VME-MXI module in the IOC and fix it.

The CPU in the beam monitor IOC stopped as often as once in a day or less depending on the configuration of the VME subrack, e.g. number of modules, position of each module, and so on. We have 20 such computers and the frequency of the CPU halts was too much for stable operation of KEKB accelerators. We observed the noises on the signals on the VME backplane when the VME-MXI driver module was accessed. We first tried to make the shape of the waveform better and decrease the number of system halts. For that purpose, we put additional load on to the bus by putting a bus extender module into a slot. It gave us a preferable decrease of system halts but was not the final solution. Then we started to analyze bus signal carefully and found abnormal bus cycles happened when the CPU module requested a write bus cycle to the VME-MXI module and the CPU module did not complete the bus cycle. We reported the fact to the manufacturer and in reply, they sent us patch information about the module. Since we put the patch on all the VME-MXI modules we have, we have not observed any halt. The process of this experience will be described.

1 INTRODUCTION

The control computer system for KEKB Accelerators [1] has a three-layer architecture called as “standard model” and is popular in accelerator control systems. In KEKB accelerator control system, VMEbus computers are used as equipment control layer, the middle layer, for easiness of upgrading, and alternate sourcing, and for reliability, availability and serviceability reasons. EPICS [2] was adopted as the software toolkit for KEKB accelerator control system and the computer in the middle layer is called as an “IOC(Input/Output Controller)”.

Since VMEbus was proposed about 20 years ago as an international standard bus for microprocessors, many compatible boards have been developed and sold in the world. Some modules have been already obsolete and are not supported by the manufacturer, however, the bus itself is still used widely because its nature of a good standard. There often happen problems arisen from the combination of several modules produced by different vendors or even by the same manufacturer.

The ordinary system configuration of the IOC for beam position monitoring system in the KEKB control system is schematically shown in Fig. 1. As the CPU module, we have been using FORCE CPU-40 (MC68040), CPU-64(MC68060), FORCE PowerCore 6603 (PPC 603e) and now, FORCE PowerCore 6750 (266 MHz and 400 MHz PPC 750). We use VXI modules made by Hewlett-Packard Japan for switching and detecting beam position monitor signals. Between VME subrack and VXI mainframes, we are using MXIbus driven by a VME-MXI module originally produced by National Instruments.

The VXI modules were developed and tested at the manufacturer by using HP9000/V743 system and VEE software. At KEK, we developed and tested driver software for EPICS by using FORCE CPU-40 and then driver software was converted from CPU-40 to PowerCore 6603 and 6750. The transition seemed to be done smoothly and well.

2 PROBLEMS HAPPENED

2.1 Sudden CPU Halts

During installation, while we were fixing initial hardware and software problems as in the usual installation, IOCs for beam monitors were found stopped once in a few days without any signs or messages. But the hardware indicator showed the CPUs were accessing VXI modules. We tested some combinations of CPU modules and VME-MXI module. For CPU-40 and CPU-60, they stopped with “BUS ERROR” light on but PowerCore 6603 and PowerCore 6750 stopped without any sign.

The accumulated numbers of rebooting were shown in Fig. 2. The inclination of the plotted curve shows the rate of CPU halts. The number may include number of rebooting due to software development but is small compared with the number of CPU halts.

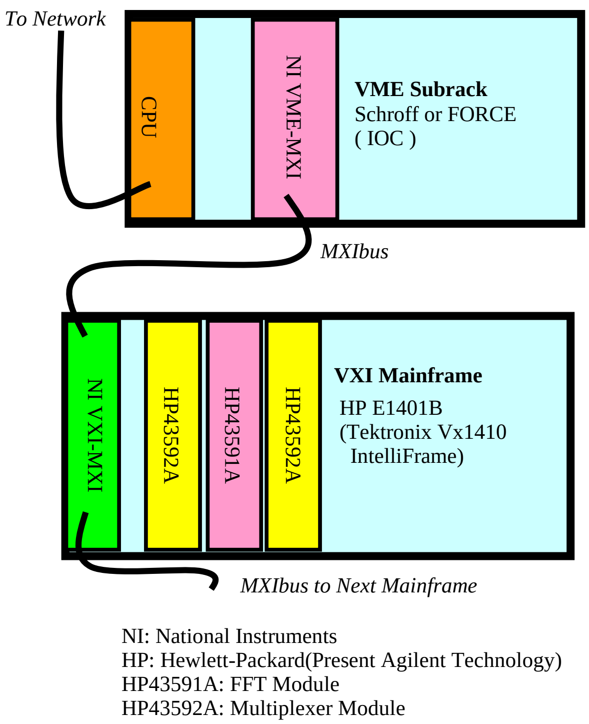


Fig. 1. Basic system configuration of a BM IOC.

2.2 For the Commissioning of KEKB

The first step was to know well about what were happening in the VME subracks when the IOCs halted. We used a VMEbus analyzer, oscilloscopes, and a VMEbus extender to put probes to the module. We waited but the frequency of the events got low due to the addition of the module and some IOCs never stopped again. There was not enough time to fix the problem completely before commissioning of the KEKB accelerators and we decided to put bus extender into all the related VME subracks. For some important IOCs, we put bus analyzers into their slots. Since then, the frequency of the IOC halts was decreased to once a week or less. It is also shown in Fig. 2 as the sudden decrease of the inclination of the plotted lines. Sudden increases of the inclination last December was caused by the increase of sampling frequencies of the beam positions.

2.3 Detailed Tests

It was found that the CPU stopped when it accessed to the VME-MXI module in write mode when we tested later. Then we could get higher frequency by increasing the number of write accesses for the traps at the test bench. We caught an unjust signal when the PowerCore 6750 accessed the VME-MXI module in write mode by using an oscilloscope. The CPU module

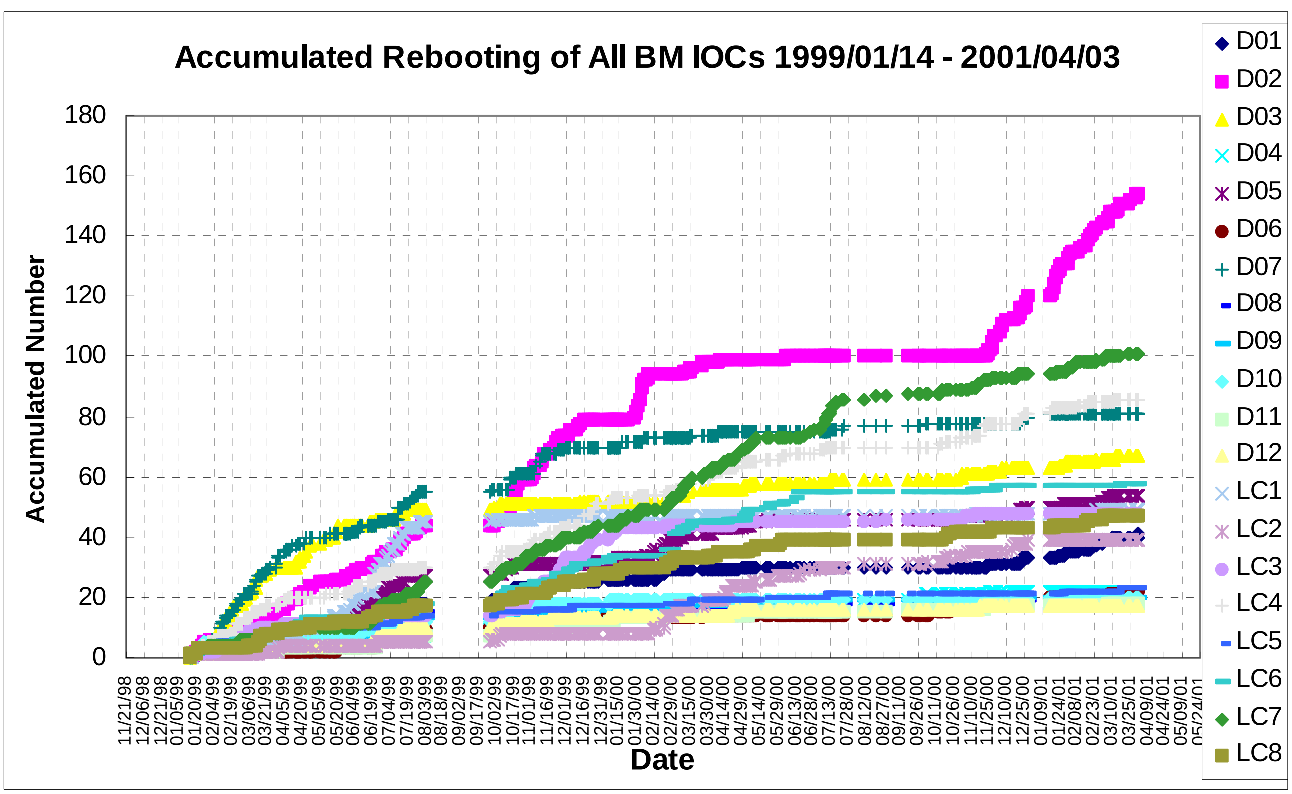


Fig. 2: Accumulated number of CPU reboots of BM IOC from 1999/01/14 through 2001/04/03.

does not release the usage of VMEbus due to unjust DTACK* signal. Examples of the normal and unjust DTACK* signal are shown in Fig. 3. In Fig. 3, normal(left) waveform has not large ringing but abnormal(right) waveform has a sharp and deep one.

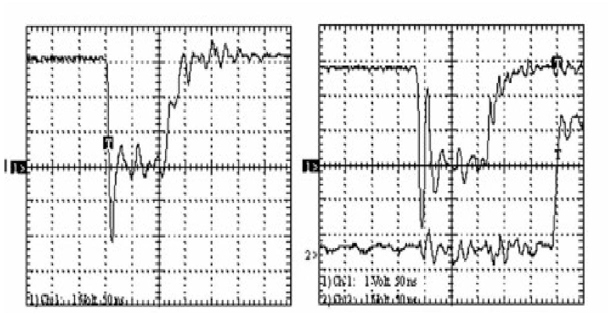


Fig.3. Examples of normal(left) and abnormal(right) DTACK* signals.

2.4 The Cause of Bus Locking

When a VMEbus write cycle is initiated by the CPU module, AS* , WRITE* and other signals are asserted on the VMEbus and sent to the VME-MXI module, which repeats these signals and sends through the MXIbus to the VXI-MXI controller in the VXI mainframe, and finally to the VXI module. The selected VXI module responds to the AS* and WRITE* by sending DTACK* back to the VXI-MXI module and to the VME-MXI module. We observed the DTACK* signal on the VXI mainframe was normal and very clean. And more, other modules than VME-MXI module did not send such unjust DTACK* signal as VME-MXI module. Therefore, the VME-MXI module was pointed as the source of the unjust DTACK* signal.

3 CPU BOARD RESPONSES

All the CPU boards use VMEbus as the external bus but the local buses inside the CPU boards are different. CPU-40 uses 68040 bus and CPU-6750 uses PCI bus as the local bus. They all have bus conversion mechanisms with custom chips to bridge internal and external buses. The CPU-40 uses FGA-002(FORCE) chip, CPU-64 uses CIX64, CPU-6603 uses Universe and CPU-6750 uses Universe II or Universe IIB(Tundra) chip. CPU-40 and CPU-64 detect “BUS ERROR”. CPU-6603 stops as CPU-6750 but less frequently than CPU-6750. The responses to the unjust DTACK* signal are different depending on the CPU and bus-bridge chips. But it is very clear that the DTACK* signal driven by VME-MXI module is unjust.

4 THE SOLUTION

We examined the circuit carefully following the circuit diagram obtained from the manufacturer and sent them more detailed information. In response to it, National Instruments pointed out that there happens a “Setup-Time” problem and the module returns unjust DTACK* signal, and finally, they sent us a patch information. After we put a patch on the board to latch the original DTACK* signal again by the unused D-type Flip-Flop in an FPGA the module started working perfectly with beautiful DTACK* signal without the dip. After applying same patches, IOCs have never stopped again.

5 CONCLUSION

It is the best way to make conformance tests by using all the modules that will be used in the actual system from the development phase. But it is quite natural that we tend to use the latest version of electronic devices and exchange components with the latest ones to get higher performance after several years. In some cases, we have to upgrade the system due to the new requirements. There happens that old modules become obsolete even if you use standard bus modules. If the modules that should be exchanged are inexpensive, it may be bought and replaced by new ones, but if they are expensive, we have to keep using old ones.

It is very important to be careful to use new modules and it is recommended to have a standard test bench and standard procedure to make conformance test.

ACKNOWLEDGMENT

Authors wish to thank members of the KEKB accelerator team for their endurable understanding, advices and help.

REFERENCES

[1] A. Akiyama, et al., "KEKB CONTROL SYSTEM: THE PRESENT AND THE FUTURE", Proceedings of the 1999 Particle Accelerator Conference, New York, 1999, pp.343-345

[2] L. Dalesio et al.: The Experimental Physics and Industrial Control System Architecture: Past, Present, and Future, Proc. ICALEPCS, Berlin, Germany, 1993, pp.179-184

TUAP049
cs.SE/0111019

APPLICATION OF DIGITAL REGULATED POWER SUPPLIES FOR MAGNET CONTROL AT THE SWISS LIGHT SOURCE

A. Lüdeke, PSI, Switzerland

Abstract

The Swiss Light Source (SLS) has in the order of 500 magnet power supplies (PS) installed, ranging from from $3A/20V$ four-quadrant PS to a $950A/1000V$ two-quadrant $3Hz$ PS. All magnet PS have a local digital controller for a digital regulation loop and a $5MHz$ optical point-to-point link to the VME[1] level. The PS controller is running a pulse width/pulse repetition regulation scheme, optional with multiple slave regulation loops. Many internal regulation parameters and controller diagnostics are readable by the control system. Industry Pack modules with standard VME carrier cards are used as VME hardware interface with the high control density of eight links per VME card. The low level EPICS[2] interface is identical for all 500 magnet PS, including insertion devices. The digital PS have proven to be very stable and reliable during commissioning of the light source. All specifications were met for all PS. The advanced diagnostic for the magnet PS turned out to be very useful not only for the diagnostic of the PS but also to identify problems on the magnets.

1 INTRODUCTION

The SLS design required a large number of magnet PS with a wide power range (see figure 1) and a variety of different

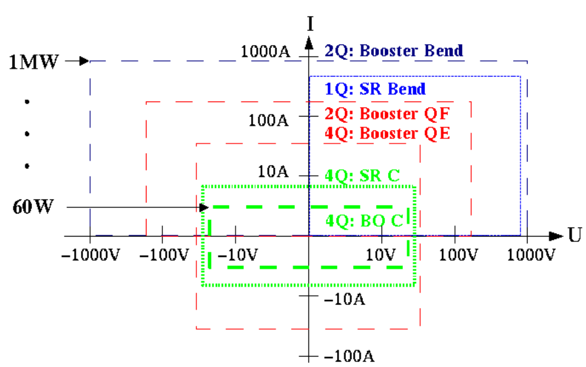


Figure 1: Operation quadrants range of the SLS magnet PS

features. For the booster a $3Hz$ sine wave ramping mode is needed. The beam-based alignment option requires the ability to introduce an individual $5Hz$ wobbling of the current for each of the 192 quadrupole magnets of the Linac, transfer-lines and storage ring. The ring orbit feedback requires $1kHz$ control bandwidth with more than a 17 bit ($15ppm$) granularity of the corrector current set-points.

The PSI division Electrical Engineering, responsible for the delivery of the PS to the SLS project chose a fully digital controlled, uniform solution for all magnet PS at the SLS. The prototypes of the PS were developed at PSI but the series were manufactured by outside companies.

2 CONTROL OVERVIEW

The overall control hardware scheme is shown in figure 2. Each PS has a digital regulation loop build by a digital sig-

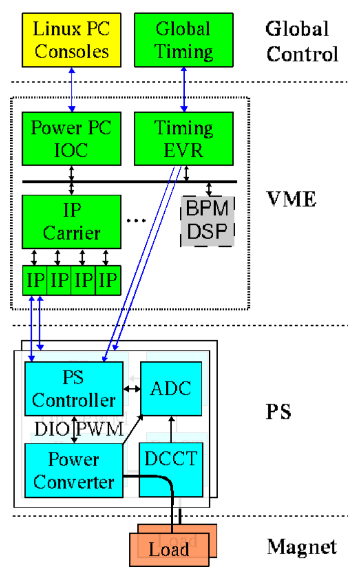


Figure 2: The control hardware scheme

nal processor (DSP) and Pulse width modulation (PWM) card, a power converter, a current measurement (DCCT) and a precise analog to digital- and digital to analog converter (ADC/DAC) card. A point-to-point optical fiber link connects the PS controller with an industry pack module hosted on a VME carrier board. An industry pack (IP) module, developed by the Accelerator Control Group at PSI, serves two PS, each one with one bidirectional $5MHz$ link. Four IP modules fits on each IP carrier VME card. Therefore even with the point to point high bandwidth link a high control density of eight PS per VME card can be reached. On average 16, at maximum 24 PS are controlled by one

[1] Versa Modular European
[2] Experimental Physics and Industrial Control System

crate.[3] The VME side hardware costs are approx. 180 \$ per PS link. The PS controller card and the ADC/DAC card were manufactured by outside companies for about 2000 \$ per PS, mainly for the controller. Plastic Optic Fiber were used for the link, a very cheap equivalent to optical fiber for short (up to 50 *m*) connections.

3 POWER SUPPLY CONTROLLER

The choice of a digital regulation loop reduces the drift problems of a PS to mainly one source: the analog digital conversion. A solution with two true 16 bit ADCs of 100 *kHz* bandwidth each and self calibration capability is used to generate a 17 bit plus sign analog signal for the 50 *kHz* control loop. Additional ADCs on the ADC/DAC card are used to measure voltages of the power converter. Two programmable DAC outputs on the card can be used for local diagnostics. Each analog register value of the controller can be assigned to each DAC with a programmable offset and scaling. This for example makes it possible to look at fluctuation of the 14th-20th bit of the read-current at a constant set current.

The high desired precision of the PS were reached by applying new methods on the PWM loop: a rounding correction and a pulse repetition modulation (PRM). Some PS have several digital controllers connected in a master and slave chain via a 30 *MBaud* backplane link of the 60 *MHz* DSP cards. For example the 1 *MVA* booster PS has 7 regulation loops on 4 DSP cards.

The optical link is controlled by a FPGA on both ends: on the DSP card and on the IP module. The protocol supports read/write access from the IP modules (bus-master) to 256 registers of 32 bits of the controller. A throughput rate of 10k float values per second is guaranteed and a maximum latency of 30 μs to set a current. Current waveforms and controller programs can be downloaded to either the DSPs SRAM or the 8 *MBit* Flash EEPROM.

The industry pack module has a special high priority register to allow a direct VME access to the PS without going through the PSC driver. This is needed for the fast orbit feedback, where a local VME DSP board writes set currents to the corrector PS through the VME bus with a rate of 1 *kHz*.

Each Controller has an additional optical trigger input. The trigger input is used by the controller to start a programmable current waveform. A precise 50 *kHz* clock on the DSP board clocks the set-point steps, one set-point each 80 μs and three interpolation steps. An offset and a scaling of the waveform can be controlled independently of the actual waveform.

A more detailed description of the digital PS controller can be found in [1].

[3] Up to 160 PS can be controlled by one VME crate, if a reduced control bandwidth is acceptable.

4 SOFTWARE LAYER

The software functionality is organized in several layers as shown in figure 3.

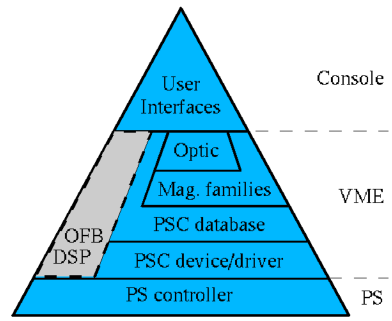


Figure 3: The control software scheme

4.1 PS Controller Software

The regulation loop and the control of fast current waveforms is handled by the PS controller. All the internal regulation parameters can be read from there and many self tests are performed here. The software of all PS controller DSPs is in general identical and differs only by regulation parameters in the Flash EEPROM.

4.2 VME Software

The next two layers – the PS control (PSC) EPICS driver / device support and the PSC EPICS database – are also identical for all PS. The EPICS templates have a total of 120 records. A third of these are connected to registers of the PS controller. Beside of the standard parameters like the switch, set and read-back current and the status of the PS there are parameters for an enhanced fault diagnostic like the average load resistance calculated by the controller. Another 40 channels are used for diagnostics of the optical fiber link and were mainly needed for the development. Only three channels are used frequently here: the flags for a broken transmit link[4], a broken receive link and the flag for local control of the controller. For the download of DSP software and current waveforms, 25 channels are used, mainly just to hold the data to download. The other channels are soft channels for extra features, like to control the initialization after booting the VME, a proper hysteresis handling, advanced alarm-handling and for VME controlled slow current waveforms. The latter ones are needed for a synchronized ($< 1ms$ Jitter) slow (seconds to minutes) ramping of an arbitrary set of magnet PS, for example local orbit bumps or an energy ramping of the storage ring.

The storage ring has individual PS for each of the 174 quadrupoles. From the optics they are grouped into 31 families with an identical theoretical current. These families are controlled by another EPICS database. All public functionality of an magnet PS is reproduced by this magnet family

[4] the PS controller detects this and sends a message to the IP module

database. Therefore the same user interfaces can be used for a magnet family as the ones used for individual magnet PS control. Each family member PS can have an offset or a scaling relative to the set value for the family. This feature is used to compensate for measured local distortions of the focusing.

Since the storage ring still has 31 quadrupole families and 9 sextupole families a further reduction of the adjustable parameters was desired. Therefore the EPICS database "Optic" reduces the number of control parameter to basically five physical quantities: the Energy, a horizontal and a vertical tune shift and a horizontal and a vertical shift of the chromaticity. A theoretical set of magnet currents and the matrices to compute the actual set currents from the physical quantities are off-line generated from a beam dynamics modeling program and downloaded to the optic database. Saturation effects are calculated by the model and are taken into account in the database by individual gradients of the families.

The software of the orbit feedback directly accesses the PS controller level without disturbing the PS device / driver.

4.3 Console Software

The standard user interface for magnet control is implemented as a special widget in the generic tcl/tk[5] program "panel.tcl" to control a list EPICS devices. The main pa-

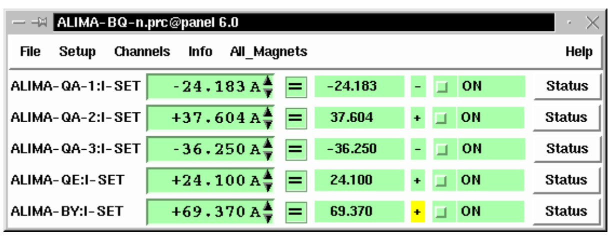


Figure 4: The generic magnet user interface

rameters are: a set current, read current, compare flag, hysteresis state and status button. The compare flag shows an alarm if set- and read current differ more than the limits given by the specifications. The hysteresis state shows if the magnet is still on its nominal hysteresis branch or if it would need to be standardized by the individual current cycle. Further status information and controls can be accessed by the more extensive pscStatus application launched by the "Status" button. This graphical interface is customized to the individual functions of the specific power supply.

Although the panel.tcl program has configuration files for all magnet PS and all magnet families, it is mainly used to control magnets in the Linac and the transfer lines and for the DC offset of the booster ramps. As mentioned above, the storage ring quadrupole and sextupole magnets are controlled by optic parameters only. The booster magnet current ramps are generated and downloaded by a ramp editor and Corrector magnets are rather controlled by the orbit correction applications. The generic interface is mainly used for fault handling of these PS.

[5]The Tool Command Language (tcl) and its graphical ToolKit (tk)

5 EXPERIENCES DURING THE SLS COMMISSIONING

The first 11 prototypes were ready for Linac Commissioning in November 1999. All layers of hard- and software had just a limited set of functionality then (no waveform handling, one link per IP module, ..) but the setup worked fine and all nessesary features were availiable. The Linac magnet powersupplies were used from then on as a testbed for continous further developments on the hard- and software, without disturbing the comissioning progress.

Another 125 PS were operational for the Booster from July 2000 on. At that time also several parts of the linac hardware were replaced by new releases of PS controllers and IP modules. Soft- and hardware worked reliable during booster comissioning. Only about 1% of downtime was encountered due to failures of power supplies, PS controllers or PS VME systems. The main source here were defective DC-DC converter on the PS controller board. The manufacturer later acknowledged a production error and the hardware was replaced.

Another 348 PS were operational for the storage ring commissioning starting December 2000. A new IP FPGA software was used with enhanced diagnostic capability and some enhancements were done to the DSP software. On VME level an alarm on the magnet resistance was introduced when short circuits were detected on some sextupole windings. This helped also to find a cabeling error which lead to two interchanged quadrupoles in the storage ring.

The electromagnets of the insertion devices will also be powered by digital PS. The first electromagnetic device UE232, consisting of two 4.4 meter electro-magnetic crossed field undulators, started comissioning in August 2001. Even this device has the same PSC EPICS database and device / driver, additional functions are added by superordinated EPICS databases.

6 CONCLUSION

The choise of the new technology of digital regulated power supplies had no drawbacks for the commissioning of the swiss light source. No mentionable extra time was needed for the commissioning of the power supplies and even the prototypes allowed reliable operation. The total development time was rather short since one solution could be used for all of the 500 power supplies. This is valid for hardware and software development of all layers.

7 REFERENCES

[1] F. Jenni, L. Tanner, “Digital Control for Highest Precision Accelerator Power Supplies”, PAC’01, Chicago, May 2001.

TUAP050
physics/0111116

CORRECTOR POWER SUPPLIES WITH A DAC RESOLUTION UP TO 24 BITS BASED ON 16 BIT DAC DEVICES*

K. Bürkmann, B. Franksen, R. Lange, I. Müller, R. Müller, J. Rahn, T. Schneegans, BESSY, Berlin, FRG
G. v. Egan, EuKontroll, Berlin, FRG

Abstract

At BESSY the standard 16 bit resolution of the corrector power supplies was insufficient for the continuous orbit drift correction [1]. A new sophisticated design of the analog/digital I/O board [3] increases the resolution of the analog output up to 24 bits without suffering on losses in the long term stability or the dynamic range of the correctors. This is achieved by a cost-efficient board design using standard 16 bit DAC devices in a range-overlapping fine/coarse architecture.

1 INTRODUCTION

The third generation light source BESSY II started operation equipped with power supplies using low thermal drift 16 Bit DAC devices. First experiences with the implemented SVD based automatic orbit correction scheme showed that the 16 bit resolution of the power supplies was not sufficient to correct the orbit without unacceptable influences to specific experiments[1].

A typical solution to solve this problem is to reduce the dynamic range of the correctors. A single bit of the DAC device then represents a smaller current step of the corrector power supply output. The obvious disadvantage of this solution is that large kicks and bumps are not achievable. This leads to an inacceptably restricted use of the power supplies for diagnostic and recalibration purposes.

To avoid the disadvantages of a reduced dynamic range a new I/O board with an increased DAC channel resolution of up to 24 bits has been introduced.

2 24 BIT DAC BOARD DESIGN

2.1 Compatibility Demands

At BESSY the power supplies are CAN bus controllable. Therefore an I/O board, together with a piggy back embedded controller including the CAN bus interface, is plugged directly into the power supply. This board combination provides the whole functionality needed for low level control of a typical power supply[3].

Taking this into account, a new 24 bit I/O board, ADA2x16-IO8, has been developed[1]. The design has the same form factor as the former 16 bit I/O board. The new board is fully compatible to the former design and consists of:

- 8 digital inputs and 8 digital outputs
- a 24 bit analog output
- a fast 16 bit flash ADC, multiplexed to 4 inputs
- a slow dual slope ± 15 bit + sign ADC, multiplexed to 4 inputs
- a connector to house the BESSY embedded controller including a CAN bus interface
- a configurable bus interface unit supporting several bus types (e.g. ISA96, VME)

2.2 Analog Output Stage Design

The long term stability of the corrector power supplies directly affects the static orbit stability of the BESSY II storage ring. This requires a high stability analog stage with low thermal drifts (in our case typically 1.5 ppm / °C or better) for the I/O board. The long term stability depends mainly on the drift of the voltage reference and the DAC output. The significant differences in thermal drifts of typical DACs available on the market are shown in Table 1.

Table 1: Errors and Drifts of Typical 24 Bit Audio DAC vs. 16 Bit Monolithic DAC

	PCM1704 Audio DAC 24 bit	AD7846 Monolithic DAC 16 bit
Gain Error	± 3 % FSR	± 0.05 % FSR
Bipolar Zero Error	± 1 % FSR	± 0.024 % FSR
Gain TC	± 25 ppm FSR / °C	± 1 ppm FSR / °C
Zero TC	± 5 ppm FSR / °C	± 1 ppm FSR / °C
FSR = full scale range, TC = temperature coefficient		

* Funded by the Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie (BMBF) and the Land Berlin

[1] The design, development and production of this board has been performed by the EuKontroll GmbH, Berlin, Germany. For detailed information please contact Georg v. Egan, EuKontroll@t-online.de

Because long term stability of the power supplies is a key feature to achieve a sufficient static orbit stability, the design of the 24 bit DAC board is based on a combination of two low thermal drift 16 bit DAC devices. A range-overlapping architecture is used to accomplish the 24 bit resolution (see Figure 1). The 16 bit DAC devices are the same as in the design of the 16 bit board. The higher 8 bits of DAC1 are used for coarse setting and the 16 bits of DAC2 for fine setting. The lower 8 bits of DAC1 are available to linearize the relative accuracy or endpoint nonlinearity of the 24 bit output if needed (in our case this feature is not used).

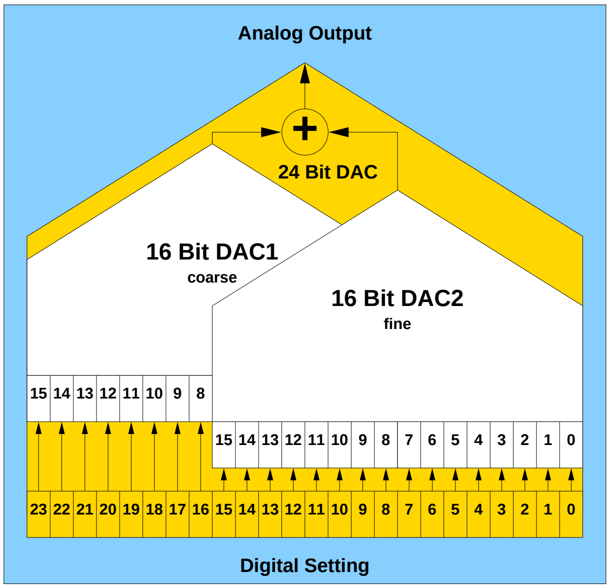


Figure 1: 24 Bit DAC Stage Principle

Critical in this design is the transition point between the two 16 bit DAC devices (e.g. bit 16 switches to one and bits 0..15 are switching to zero). This directly affects the differential nonlinearity and therefore the monotonicity of the design. In our case the board is calibrated to provide monotonicity up to 17 bits resolution. Considering a full scale range of $\pm$ 10 V a single bit of 24 bit resolution represents 1.192 μV and 17 bit represents 152.59 μV; i.e. the design is calibrated to be monotonic for relative settings down to 152.59 μV or better.

2.3 Measurements

Figure 2 shows the measured relative output steps of one randomly selected 24 bit DAC when the input is incremented by steps of 2^5 digits, which is equivalent to a 19 bit operation mode. In the ideal case of a 19 bit operation mode, every relative output step of the DAC should be 38.15 μV. In Figure 3 the majority of measured output steps are in the 38.15 μV region and only some few are in the 152.59 μV region.
Because of these encouraging measurement results we expected significant improvements regarding the resolution of the corrector power supplies and therefore performance improvements of the orbit correction scheme.

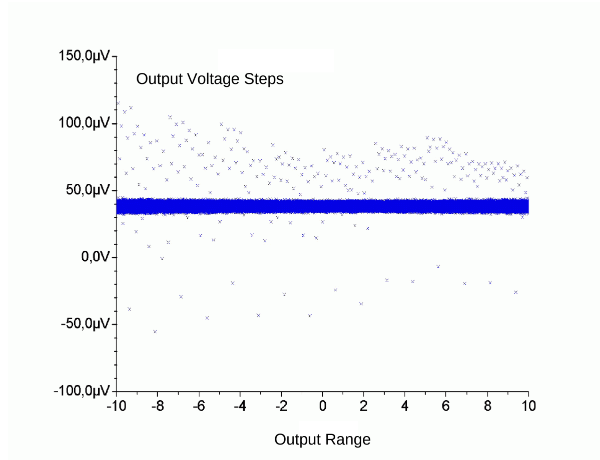


Figure 2: Output Voltage Steps vs. Input Setting Steps (19 Bit Operation Mode)

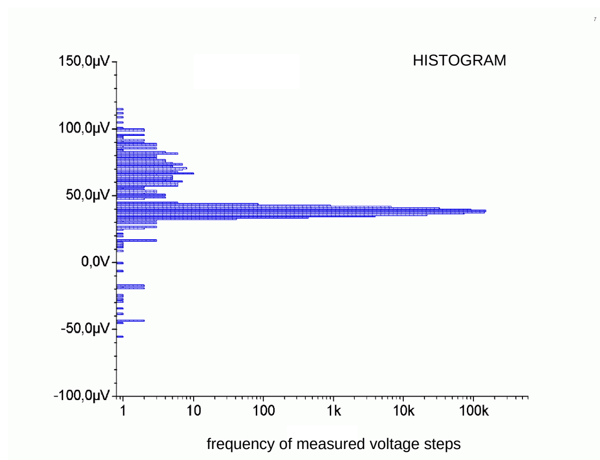


Figure 3: Histogram of Output Voltage Steps (19 Bit Operation Mode)

3 PERFORMANCE

After installation of the ADA2x16-IO8 24 bit DAC boards, first tests with the SVD based orbit correction scheme have been done using different resolutions for the setting. Figure 4 shows the FFT performed over the resulting time dependent vertical beam position data of all beam position monitors (BPMs). A drastically reduced orbit drift could be seen when the resolution was increased from 16 bit to 18 resp. 20 bit [2].

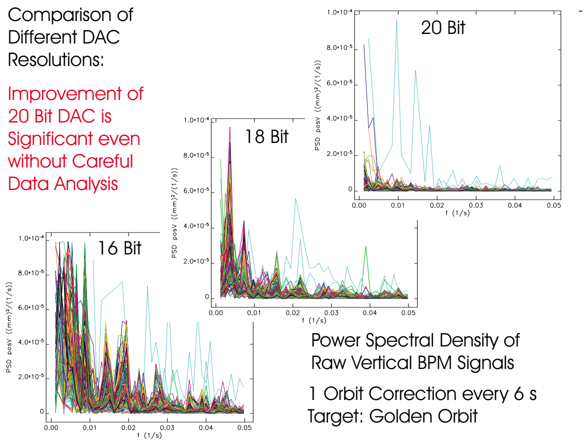


Figure 4: FFT of Time Dependent Beam Position Data

4 CONCLUSION

The new 24 bit design of the analog/digital I/O board provides the higher resolution needed for the correctors of the third generation light source BESSY II. Due to the range-overlapping architecture of the board using the low thermal drift 16 bit DAC devices, the known static orbit stability is guaranteed even without active orbit correction. Applying a SVD based orbit drift correction algorithm the closed orbit is now being corrected with a stability of typically $< 1..2\mu$m.

5 REFERENCES

[1] Orbit Control at BESSY II, R. Bakker, K. Holldack, R. Müller, P. Kuske:BESSY, EPAC 2000, Vienna, Austria, p. 666

[2] Orbit Drift Correction Using Correctors with Ultra-High DAC Resolution , K. Bürkmann, R. Bakker, F. Falkenstern, J. Feikes, B. Franksen, R. Görgen, P. Kuske, R. Lange, I. Müller, R. Müller, J. Rahn, T. Schneegans:BESSY, PAC 2001, Chicago, USA

[3] CAN: a Smart I/O System for Accelerator Controls, J. Bergl, B. Kuner, R. Lange, I. Müller, G. Pfeiffer, J. Rahn, H. Rüdiger:BESSY , ICALEPCS 1997, Beijing, China, p. 292

TUAP051
hep-ex/0111025

THE NEW MAGNETIC MEASUREMENT SYSTEM AT THE ADVANCED PHOTON SOURCE *

Yu.Eidelman, B.Deriy, O.Makarov, I.Vasserman, APS, Argonne, IL 60439, USA

Abstract

A new system for precise measurements of the field integrals and multipole components of the APS magnetic insertion devices is described. A stretched coil is used to measure magnetic field characteristics. The hardware includes a number of servomotors to move (translate or rotate) the coil and a fast data acquisition board to measure the coil signal. A PC under Linux is used as a control workstation. The user interface is written as a Tcl/tk script; the hardware is accessed from the script through a shared C-library. A description of the hardware system and the control program is given.

1 INTRODUCTION

A new system for precise measurements of magnetic field integrals and multipole components of different types of insertion devices (IDs) is created as an upgrade of the Magnetic Measurement Facility at the APS [1]. The main reasons for upgrading the existing system were:

- To increase accuracy and reproducibility of acquired data. Long cables between motor drives and stepper motors in the existing system produce an excessive noise interfering with the measured signal. Integrators in the existing system take 4 data values per coil turn. The new system uses ≈ 1000 data points of coil signal for analysis, leading to better accuracy. Different error sources presented in the coil signal can be analyzed, particularly coil vibrations.
- To simplify the system and increase reliability for ease of maintenance, future developments, and possible duplication of the system in other projects. In the existing system VME, Euro, as well as non-standard crates, have been used. It has been very hard to maintain and almost impossible to reproduce the system.
- To allow integration of the results of the magnetic measurements with other APS databases and utilities. The Linux OS instead of Windows has to be used in order to reuse software modules developed by Operations Analysis Group [2] for the APS storage ring.
- To provide more convenient data analysis in both standalone and network operation.
- To add more features, such as a translation mode, for measuring multipole magnetic field components.

* Supported by the U.S. Dept. of Energy, BES, Office of Science, under Contract No. W-31-109-Eng-38.

2 PRINCIPLE OF THE MEASUREMENTS

The stretched coil is used to measure magnetic field integrals. The coil signal is proportional to the time derivative of the magnetic flux through the coil cross-section. The measurements are performed at constant coil rotation (translation) speed. There are four different options for measurements.

Integrals and **rotation** modes. Signals are measured for full turns of the coil. Two turns, clockwise (CW) and counter-clockwise (CCW), are performed to exclude systematic errors. The measured data are applied to calculate absolute values of the first or the second (if the coil is twisted by 180° [3]) magnetic field integrals. A set of approximately ten points across the gap of the ID is measured to calculate multipole magnetic moments.

Translation mode. The linear motion of the coil to measure the magnetic flux changing in the horizontal or vertical direction is examined, allowing significant reduction in the measurement time of the multipole moments.

Variable field mode. The coil does not move. This mode is designed for measurements of IDs with switching electromagnets [4].

3 HARDWARE DESCRIPTION

A block diagram of the new system is shown in Fig. 1. Eight smart motors from Animatics Corp. [5] are used. Each servomotor has an encoder, a motor drive, and a motor controller—all integrated in one unit, thus reducing the level of the electrical noise, compared to the existing system. These smart motors have two standard serial ports—RS-232 and RS-485. All motors connected in parallel to the RS-485 communication line are controlled by the Pentium III computer under Linux OS through a RS-232/RS-485 converter.

The stretched coil conventional configuration with parallel wires is being used for measurements of the first field integrals and multipole components. The measured signal is amplified by a SCXI-1120 National Instruments amplifier. The gain is set to 2000, and the bandwidth is set to 10 kHz. A 500 kHz data acquisition board PD2-MFS-500/16 from United Electronics Industries [6] is used to measure the am-

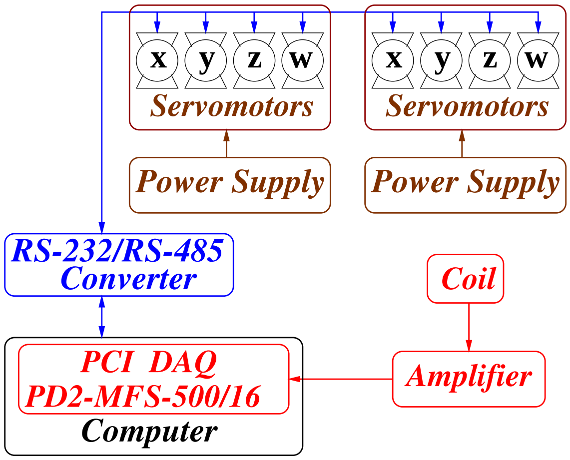


Figure 1: Layout of the upgraded system.

plified signal. The Linux driver for this PCI DAQ board was provided by the manufacturer.

4 SOFTWARE DESCRIPTION

A new Multipoles & Integrals—Software System (MISS) has been written. It consists of two parts: a Tcl/tk script and a shared C-library.

4.1 *Tcl/tk script*

The Tcl/tk script describes a user interface and data processing and uses a shared C-library for hardware access. The script language includes all embedded instruments to create the user-friendly interface and to use a shared C-library. Besides, MISS uses utilities developed by the APS Computer Support Group for this language. MISS operates under Linux and provides the user interface through a set of windows and worksheets (see Fig. 2).

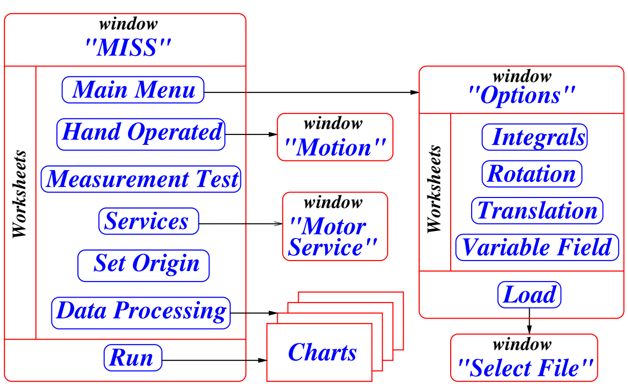


Figure 2: Software chartflow diagram.

Figures 3 and 4 demonstrate examples of the user interface.

4.2 *Shared C-library*

The hardware is accessed from the script through a shared C-library. This library consists of two command sets—one

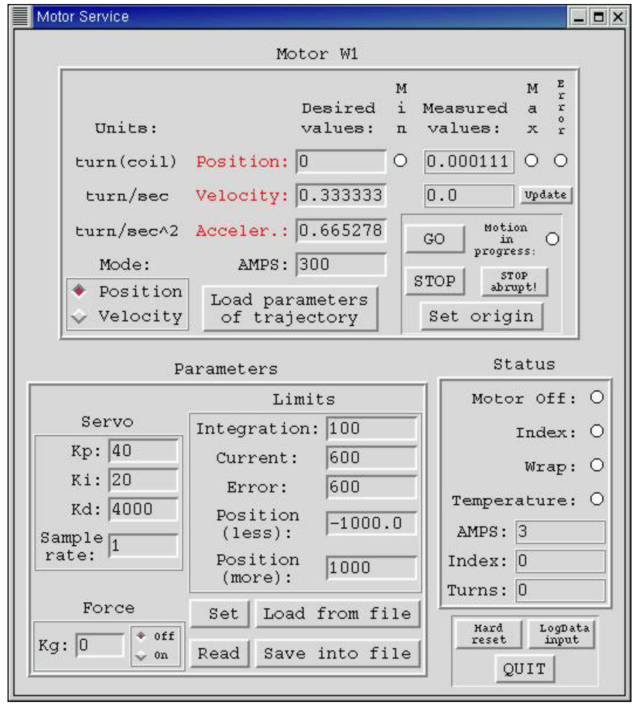


Figure 3: Window “Motor Service.”

for smart motors and another for the DAQ board. The serial port driver “ttyS” with appropriate setting of the structure “termios” is used to control the motors. The Linux driver “pwrdaq” is used to control the DAQ board. Some of the DAQ board commands are based on the corresponding C-functions of this driver. The complex procedures providing synchronization between coil motion and data acquisition are included in the C-library. These procedures combine calls to the motors and the DAQ board in the required sequence. So, the execution of such commands cannot be interrupted by a Tcl/tk script command.

5 SYNCHRONIZATION OF THE COIL MOTION AND DATA ACQUISITION

This process includes two steps.

Integrals and **rotation** modes. At the first step, the choice of coil “trajectory” (velocity and acceleration of the rotation) determines the calculated delay between initiating the coil motion and starting the data acquisition. The second step includes matching the additional delay to achieve agreement between the signals measured for CW and CCW coil turns. This matching is based on the assumption that plots of measured signals (voltage versus the coil’s turn angle α) for both of these rotations must be mirror symmetric, if delays are chosen correctly. Figure 5 shows the measured signals for both cases after integration and subtraction of offsets. It was found that the additional delay is equal to 25 ms for a calculated delay of 550 ms.

Translation mode. The first step is identical to the first step in the rotation mode. The additional delay was determined

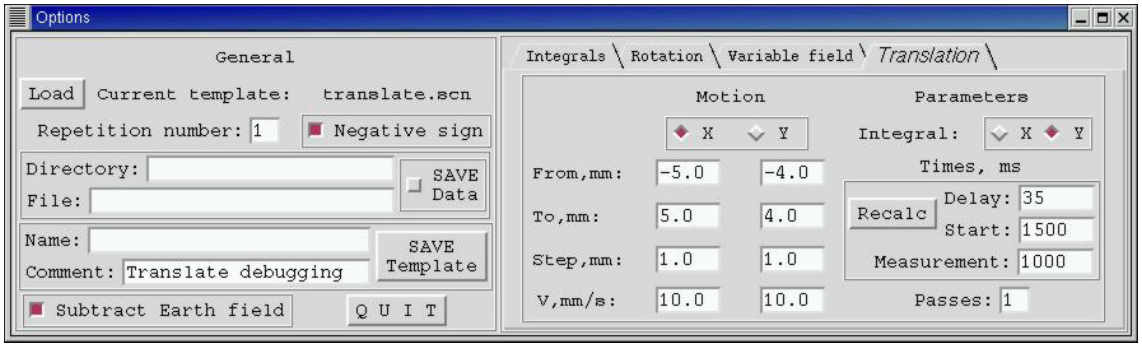


Figure 4: Worksheet "Translation" of the window "Options".

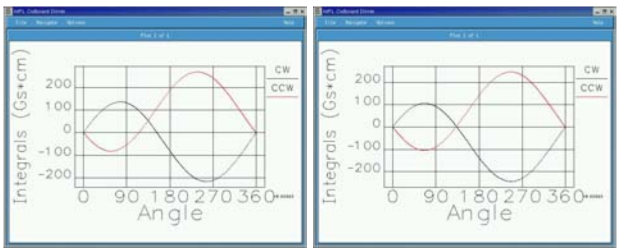


Figure 5: Integrals of measured signals for calculated (left) and matched (right) starts of the data acquisition.

by using the rotation mode with the same velocity and acceleration of the smart motors. This delay is equal to 35 ms for a calculated delay of 1500 ms.

6 DATA PROCESSING AND SOME RESULTS

Different algorithms are used to obtain the magnetic properties of the IDs.

A) Measuring of the first and second integrals includes the following steps:

1. Fitting of the functions $A_i \sin(\alpha + \phi_i)$ on noisy coil signals for CW ($i = 1$) and CCW coil ($i = 2$) turns.
2. Calculating of magnetic field integrals I_x, I_y for each coil turn from fitting parameters A_i, ϕ_i.
3. Repetition of previous steps to improve accuracy and define rms errors of measurements.

The statistical error of integrals I_x, I_y (measured for the Earth's magnetic field) is equal to $\pm\, 0.4$ Gs·cm (improved from $\pm$ 2 Gs·cm before upgrading).

B) Measuring of the integrated multipole moments in the rotation mode includes all steps from case A) for a set of points r_i across the ID gap and, after that:

4. Fitting of the polynomials $\sum_{k=0}^{n} b_k r^k$ of the power n on the sets $I(r)$ (separately for arrays I_x and I_y).

Coefficients b_k are integrated multipole moments.

C) Measuring of the integrated multipole moments in the translation mode includes:

1. Fitting of the polynomials $\sum_{k=0}^{n} b_k r^k$ of the power n on noisy signals for a few coil passes across the gap of the ID.
2. Averaging of obtained coefficients b_k over the set of coil passes to reduce the statistical errors.

Note that the translation mode saves significant time in measuring the integrated multipole moments in comparison with the rotation mode. Measurements have to be done for both horizontal and vertical coil orientations.

A strong short magnet was used to evaluate the accuracy of the measurements of the multipole moments. Table 1 shows these results which were obtained for the translation mode.

Table 1: Multipole moments measurements.

k	b_k	δb_k	Units
0	652.8	$\pm$0.1	Gs·cm
1	2208	$\pm$2	Gs
2	1840	$\pm$7	Gs·cm^{-1}
3	-1167	$\pm$48	Gs·cm^{-2}
4	-2929	$\pm$127	Gs·cm^{-3}
5	1985	$\pm$203	Gs·cm^{-4}
6	460	$\pm$518	Gs·cm^{-5}

A few programs from the SDDS Toolkit [7] are used for polynomial fitting, data management, and displaying charts.

7 REFERENCES

[1] L. Burkel et al., "The Insertion Device Magnetic Measurement Facility: Prototype and Operational Procedures," Argonne National Laboratory Report No. ANL/APS/TB-12, March 1993.

[2] http://www.aps.anl.gov/asd/oag

[3] D. Frachon et al., "Magnetic Measurements of the Elliptical Multipole Wiggler Prototype," Argonne National Laboratory Report No. ANL/APS/TB-22, March 1995.

[4] E. Gluskin et al., “An Electromagnetic Helical Undulator for Polarized X-rays,” AIP Conf. Proc.: Synchrotron Radiation Instrumentation, vol. 521, P. Pianetta, J. Arthur, and S. Brennan, eds., AIP (2000) pp. 344–347.

[5] http://www.smartmotor.com.

[6] http://www.PowerDAQ.com.

[7] http://www.aps.anl.gov/asd/oag/manuals/SDDStoolkit/SDDStoolkit.html

TUAP052
physics/0111062

GPIB ADDRESS CONVERTER

Y. Suzuki, M.Ieiri, Y.Katoh, E.Kusano, M.Minakawa, H.Noumi, M.Takasaki, K.H.Tanaka, and Y.Yamanoi
KEK, Tsukuba, Ibaraki 305-0801, Japan

Abstract

A GPIB address converter (GAC) has been constructed. This paper reports on the function and test results. The GAC has two GPIB connectors (upper and lower ports). The upper port has a GPIB primary address, and is connected to a GPIB system controller. The lower port acts as a GPIB controller of the lower side GPIB line. The GPIB system controller can access the lower side GPIB devices through the GAC by using an extended two-byte address function. The two-byte address (primary + secondary) is shown in the combination of the GAC address and the address of the lower side device. The GAC converts the secondary address into the primary address of the lower side GPIB device. By using of 30 GACs, the GPIB system controller can access 930 devices assigned only primary addresses.

1 INTRODUCTION

In controlling and monitoring the accelerator or the experimental physics, the GPIB is one of the useful field buses. When the primary address is used, one GPIB controller can control 30 devices. According to the specifications: (IEEE-488.1), one GPIB controller can control 960 devices when the extended two-byte address function is used. However, there is one inconvenience. There are many useful instruments equipped with GPIBs: oscilloscopes, multi-meters, and accelerator control devices. However, they are scarcely equipped with the extended two-byte address function. In this situation, the GPIB address converter (GAC) was developed. Figure 1 shows a photograph of the GAC. Figure 2 shows an example configuration of the GAC in a GPIB system. Figure 3 shows a block diagram of the GAC.

Figure 1. The GAC is assembled into a NIM module.

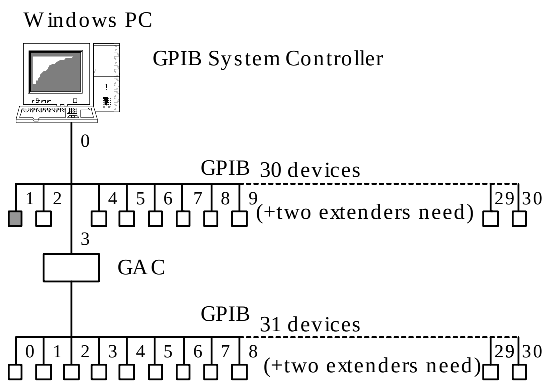


Figure 2. Configuration example

When the GPIB controller addresses the GAC, the GAC converts the secondary address into the primary address, and then the GAC controls the devices with the primary address through the lower GPIB port. The data, which pass through the GAC, do not change. Thus, the GPIB system controller does not need to add any program or to change. By using of this GAC, one GPIB controller can control 930 devices assigned only a primary address.

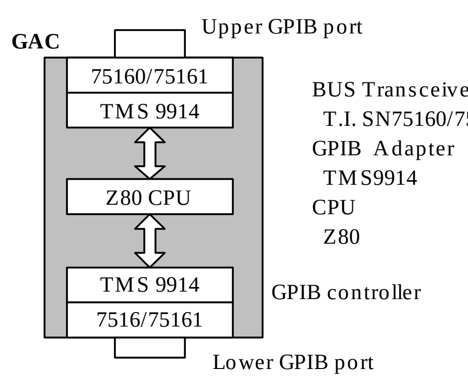


Figure 3. Block diagram of the GAC

2 GAC

2.1 Hardware

The main parts of the GAC are two GPIB adapters, TMS-9914 and an eight-bit microprocessor Z80. In the block diagram of Figure 3, the upper port acts as a GPIB device, not a controller, and occupies one GPIB primary address number between 0 and 30, except for the address number of the system controller. The lower port GPIB adapter acts as a GPIB controller of the TMS-9914's local mode, and thus it does not occupy any GPIB address. Thus, 31 numbers of 0 to 30 can be assigned to the lower side devices.

In an actual use situation, the maximum number of the GPIB devices directly connected to the GAC's lower port is limited to 14. The reason is based on the fan-out ability of the bus-driver: 75160/75161 (regulated by IEEE488). A GPIB extender is available for expanding of the bus.

2.2 Address Conversion

The secondary address number, received from the system controller, is converted into the primary address by following equations:

LAn: Listener primary Address n code
codes 20H to 3EH are assigned to LAn 0 to 30
codes 3Fh is assigned to UNL: Un-Listen
TAn: Talker primary Address n code
codes 40H to 5EH are assigned to TAn 0 to 30
codes 5FH is assigned to UNT: Un-Talk
SCm: Secondary Address m code
code 60H to 7FH are assigned to SCm 0 to 31

1) In the listener condition
LAn=SCm-40H

2) In the talker condition
TAn=SCm-20H

Then n=m.

LAn and TAn are used to control of the lower side device.

2.3 Listener Sequences

The GAC receives the following message bytes from the system controller:

[Upper GPIB port]
UNL : inhibits all current listeners (Un-Listen)
TAD : system controller is assigned to the talker
LAD : GAC is assigned to the listener
SCm : receive secondary address

At this point, the GAC accepts all SCms.

The GAC sends the following messages to the lower side GPIB port:

[Lower GPIB port]
UNL : Un-Listen
LAn : LAn=SCm-40H

At this point GAC checks the upper GPIB port. Received interface messages are transferred to the lower side GPIB. Those messages are as follows:

SDC : Selected Device Clear
GTL : Go To Local
GET : Group Execute Trigger

When a **DATA** byte is received, the lower GPIB adaptor, TMS-9914, is set in the Talk Only mode, and the **ATN** (attention) signal is set to L (0). Then, the data read out from the upper GPIB port is repeatedly written in the lower GPIB port. **EOI** (end or identify) signal is also checked, and it is sent to the lower GPIB port with the data.

2.4 Talker and Serial Poll Sequences

The GAC receives the following message bytes from the system controller:

[Upper GPIB port]
UNL : Un-Listen
LAD : system controller is assigned to the listener
TAD : GAC is assigned to the talker
SCm : receive secondary address

The next step has two cases: one is **SPOLL** (Serial Poll); the other is a data request. However, it is not possible to know the serial pole in advance. Then, every time SPOLL is executed, STB is read out from the lower GPIB device. The flow is as follows:

[Lower GPIB port]
UNL Un-listen
SPE : serial poll enable
TAm : the lower device is assigned to Talker
STB (SBN or SBA): Status byte
SPD : serial poll disable
UNT : Un-talk

The STB from the device is set to the serial poll register of the upper GPIB adapter (TMS-9914).

Then, the operation of the upper GPIB is permitted (by the Z80-cpu), and the operation is continued. When SPOLL has not happened by the system controller, STA is stored in the STB memory for the next SPOLL (for when receive SCm). Then, the memory is cleared after SPOLL has been received.

(STA: represent a status byte sent by a device in which a request for service is indicated (bit 7=1)).

(STB: represents a status byte sent by a device in which a request for service is not indicated (bit 7=0)).

[Upper GPIB port]
ATN=0 means a demand for DATA
The next procedures are as follows:
[Lower GPIB port]
UNL : Un-Listen
TAm : lower side devise is assigned to Talker
Set GPIB adapter TMS-9914 listen-only mode
ATN=0

DATA : read data are repeatedly sent to the upper GPIB port. The EOI signal of the lower GPIB port is checked, and is also sent along with the DATA byte.

2.5 SRQ, SDC, DC, and IFC,

The SRQ line in the lower GPIB port is always watched, and the SRQ signal is transferred immediately to the upper GPIB port. When STA is stored in the STB memory, an SRQ signal is also set.

The lower GPIB port executes SDC (selected device clear), DC, or IFC(Interface Clear) when those interface messages are received in the upper GPIB port.

3. OPERATION RESALT

To confirm the operation of the GAC, used equipment are shown in Table-1.

Device	Address
GAC	3
Multi-meter HP34401A	3+22
Multi-meter HP34420A	3+23
Function generator HP33120A	3+10
Oscilloscope HP 54602B	3+07
PSCx8 Power supply controller	3+13
Windows PC	
Agilent VEE 6.0	
Lab VIEW 6.0	

Table-1

3.1 Connection Test

When using a Windows PC, the GAC and the measuring devices were connected in the GPIB cables in series. The test program was written in VEE. The following GPIB functions are included for tests: Listener, Talker, Serial poll, SDC, DC, Remote/Local, GET, IFC, and binary data transfer with EOI. The Bus Monitor of the VEE monitors the message transfer on the GPIB. It has been confirmed that the data are smoothly transferred through the GAC. Also, concerning the operation of the application program Agilent VEE: the Panel Drivers of HP34401A, HP34420A, HP33120A, and HP54602B have been confirmed to mount on the Windows PC display correctly and to work without any additional program or change.

3.2 Transfer Speed

The transfer speeds were measured under the following two conditions: controller is connected directly to the device (PSCx8); the other is connected through the GAC.

Test-1: Loop of 1000 times of (UNL, MTA, LAD, 16 byte DATA with EOI).

Test-2: Loop of 1000 times of (UNL, MTA, LAD, 4 byte DATA with EOI, + UNL, MLA, TAD, 21 byte DATA with EOI).

Test-3: Serial poll 1000 times.

The test results are shown in Table-2.

Time (sec)	Test-1	Test-2	Test-3
Direct	7.299	12.22	7.236
with GAC	7.459	12.74	7.575
+dt (%)	2.2	4.3	4.7
Windows PC 450MHz, Agilent VEE 6.01			

Table-2

The delay time (+dt) % of the data through the SAC on Test-1, Test-2, and Test-3 are, respectively, 2.2%, 4.3%, and 4.7%. The data-transfer speeds of the other devices are slower than the PSCx8, and thus the delay time is smaller.

4. CONCLUSION

A GPIB address converter (GAC) has been developed. The function has confirmed that the GAC adds devices with the extended two-byte address function. By using the GAC, it is possible to configure up to 930 ordinal devices equipped with the primary address function on one GPIB.

To improve the SAC transfer speed, two CPUs, H8/3048-(16-bit) and SH7045 (32-bit), are being tested in the SAC.

TUAP053
cs.AR/0111029

VERSATILE DATA ACQUISITION AND CONTROLS FOR EPICS USING VME-BASED FPGAS*

T. Allison, R. Flood, TJNAF, Newport News, VA 23606, USA

Abstract

Field-Programmable Gate Arrays (FPGAs) have provided Thomas Jefferson National Accelerator Facility (Jefferson Lab) with versatile VME-based data acquisition and control interfaces with minimal development times. FPGAs have been used to interface with VME controllers using standard A16 and A24 address modes. VME vectored-interrupt capability has also been implemented in some applications. FPGA designs have additionally been used to provide control logic for numerous systems by interfacing with Analog to Digital Converters (ADC), Digital to Analog Converters (DAC), various interlocks, and other drive signals. The building blocks of these logic designs can be tailored to the individual needs of each system and provide system operators with read-backs and controls via a VME interface to an EPICS based computer. This versatility allows the system developer to choose components and define operating parameters and options that are not readily available commercially. Jefferson Lab has begun developing standard FPGA libraries that result in quick turn around times and inexpensive designs. There have been approximately twelve VME-based FPGA designs implemented in the RF and Electronic Support (RFES) Group at Jefferson Lab. FPGA logic density and device speed continue to increase which enables many system designs to be incorporated onto one FPGA. FPGA designs can manipulate data quickly due to the small processing overhead associated with a custom design. This coupled with physical performance advances and optimized logic from compiler tools makes FPGAs solutions faster than many microprocessors. The ability to modify and simulate this firmware enables a designer to easily add new enhancements to a system or modify existing parameters, permitting the design to be both flexible and expandable for future applications.

1 INTRODUCTION

In the early years of VME design, interfaces typically required a substantial number of discrete TTL devices and Printed Circuit Board (PCB) space to decode and generate the necessary timing signals for proper VME communication. The interface was hard-wired leaving little ability to change or add features to the module after committing the design to fabrication. Commercial modules were typically expensive and often did not provide the specific features required by unique laboratory applications.

With the introduction of fusible-link logic and eventually, ultraviolet erasable and electrically erasable Programmable Logic Devices (PLDs), VME interface designs began to take on a more generic appearance. PLDs contained the application-specific logic, such as control logic, address decoding, and register decoding which defined the board's function. The ability to reprogram the firmware allowed for logic corrections to be made to the design even after board fabrication. Increased use of Computer Aided Design (CAD) tools permitted generic designs to easily be re-used or enhanced for new applications.

The Altera 7000 series FPGA devices were among the first FPGAs to be incorporated in VME designs at Jefferson Lab. These devices provided much greater logic densities than first-generation Electrically Programmable Logic Devices (EPLDs), often at lower cost and in smaller packages. In addition to incorporating a simple VME slave interface, several registers and complex state-machines could now be implemented directly in the device. Component size was also changing with smaller Plastic Leadless Chip Carrier (PLCC) and Quad Flat Pack (QFP) packaging becoming more commonplace. Most members of this logic family still required removal from the circuit board for re-programming.

Recent VME and stand-alone designs at Jefferson Lab have used Altera FLEX10K or ACEX1K devices. These are SRAM-based components requiring a companion Electronically Erasable Programmable Read Only Memory (EEPROM) configuration device. All of theses components are available in a variety of packages, including PLCC, QFP, and Ball Grid Array (BGA). Device voltages of 5V, 3.3V, and 2.5V have been utilized, depending on the logic family chosen and overall project design. All Altera SRAM-based and EEPROM devices include a Joint Test Action Group (JTAG) interface. JTAG is a boundary scan technology used for board level interconnection integrity testing but also provides In System Programming (ISP) capability. This allowed for easy reprogramming of the FPGA without having to remove the component or use socket devices.

Many applications have been embedded into FPGAs since their introduction at Jefferson Lab. Data

*This work was supported by the U.S. DOE Contract No DE-AC05-84-ER40150

acquisition and system controls have been predominating. More recent applications have been performing mathematical operation and will expand to DSP algorithms and embedded processors.

2 TOOLS

The logic for the FPGAs is mainly developed using Altera Hardware Description Language (AHDL) with Altera's Maxplus II software. This is a specialized hardware description language that is optimized for Altera devices. It is largely based on Very High Speed Integrated Circuit (VHSIC) Hardware Description Language (VHDL) but targets the timing characteristics and physical structure of Altera devices to provide accurate timing simulations and help the designer to determine the correct Altera FPGA for an application.

Once the generic VME interface was constructed in AHDL, the code was reused in multiple designs with slight modifications for each application. This is also the case when communicating to DACs, ADCs, SDRAM, FLASH memory, and any other device. This reuse allows for short development times and can be molded to the needs of each system.

VHDL has also been used in some designs with the ALDEC Active-HDL software. The ALDEC compiler does not target a device like the Altera compiler so the simulations do not incorporate the timing characteristics of the FPGA. This compiler does allow the designer to fit a design to almost any FPGA from any manufacturer however only Altera FPGAs have been implemented in the RFES group at Jefferson Lab.

P-CAD is the PCB design tool used to develop boards at Jefferson Lab. Once the VME interface is defined in a schematic and on a PCB layout then it can also be reused to cut down on development time. The most popular FPGAs used at Jefferson Lab are from the Altera FLEX10K series. This family was chosen because they are available in a 240-pin QFP that does not take up a large amount of board space but allows for many input and output pins. The devices range for 20,000 gates to 100,000 gates and come in 5 volt, 3.3 volt, and 2.5 volt packages. This allows for the same PCB footprint to be used as well and the same logic code. Other devices have also been used and have similar characteristics.

3 APPLICATIONS

3.1 System Catch All Module (SCAM)

The SCAM was developed as a replacement for an existing hard-wired VME module that had been in service for a number of years. The SCAM module generates timing pulses used to modulate the three-beam laser used in the polarized source of the Continuous Electron Beam Accelerator Facility (CEBAF) injector at Jefferson Lab. Thus supporting independent beam delivery to each of the three experimental halls.

The PCB is a 4-layer 6U module with an 8-bit VME slave interface. The front panel supports 16 optical input/outputs as well as 18 Lemo connectors. A GAL20V8 provides VME A16 address-decoding for seven 8-bit registers contained in an 84-pin Altera EPM7160 EPLD operating at 16Mhz. Numerous state-machines and counters were implemented in the device.

3.2 Injector High Voltage Controller

The Injector High Voltage Controller provides control of four high-voltage relays and a 100kV power supply used to power the electron guns in the CEBAF injector. Two voltage-to-frequency converters and two 16-bit DACs are used to set and read back the power supply voltage. Many interlocks that shut off the power supply are incorporated to ensure that personnel are not exposed to high voltage. The module also slowly ramps the voltage to the set point and provides an over-current limit to prevent damaging the electron guns. When the power supply is turned off, a timer and a voltage read-back is used to ensure the high voltage bleeds off before it can be turned back on. This helps avoid arcing.

The PCB is a 4-layer 6U board with an 8-bit VME slave interface. The front panel supports 2 optical outputs as well as a JTAG connector. A GAL20V8 provides VME A16 address-decoding for nine 8-bit registers contained in a 240-pin Altera FLEX10K50 FPGA operating at 10Mhz. The device is re-programmable via the front-panel JTAG interface.

3.3 30Hz Board

The 30Hz Board provides synchronous timing signals to various Input/Output Computers in VME crates located throughout the CEBAF accelerator [1]. Communications are established via fiber optic cable in a token ring arrangement. Each board is assigned a unique serial address via on-board jumpers. Address 0 is reserved as a broadcast address in which all boards decode the serial message. Upon receipt of such a message, each board decodes the data and generates a VME interrupt if enabled.

The PCB is a 4-layer 3U board with a 16-bit VME slave interface, which includes D08 vectored-interrupt capability. The A24 address decoding, the VME interface, twelve 8-bit registers, and the serial message encoding/decoding are contained in a 2.5 volt 144-pin Altera ACEX EP1K50 FPGA operating at 20 Mhz. The FPGA is re-programmable via a front-panel JTAG interface.

3.4 Dual DSP Board

The Dual DSP Board is a versatile general-purpose digital signal processing board utilizing two Texas

Instrument TMS320C6711 floating-point signal processors [2]. An FGPA is used to provide arbitration between the two DSPs and 128 kilobytes of dual port memory. The FPGA also maps the dual port memory to VME address space and provides communication across a custom P2 back plane.

The module is a 10-layer 6U board and uses an Altera FLEX10K50 FPGA operating at 25 Mhz that supports VME A24 address decoding, a 16-bit VME slave interface, block-transfer cycles, read-modify write cycles, address pipelining, and D08 and D16 vectored-interrupts. Programming of the DSPs and the Altera FPGA is accomplished via two separate front-panel JTAG interfaces.

3.5 Machine Protection System (MPS) Comparator

The BCM Comparator module is used to receive and manipulate data from eight of the Dual DSP Boards over a custom P2 back plane. In this application the Dual DSP Boards are configured to measure the beam current in the CEBAF injector and at the multiple end stations. The MPS Comparator module computes beam loss by tallying the end-station currents and comparing the sum to the measured injector current. This difference is the instantaneous loss. The module also compares the current from each location to maximum limits set by the operators. If these limits are exceeded then CEBAF is shutdown. The instantaneous loss is integrated in an adaptive algorithm to produce the integrated loss. If the integrated loss becomes large enough then CEBAF is shutdown. Information about the beam loss is stored in an 8 megabyte SDRAM circular buffer. The SDRAM is mapped to VME address space with a pointer to the start of the buffer. If CEBAF is shutdown then the buffer can be read and the beam loss history reconstructed. A DAC is also used to represent the instantaneous loss and loss data is sent directly to the Machine Control Center via a fiber optic interface.

The module is a 4-layer 6U board and uses a 3.3 volt Altera FLEX10K100 FPGA operating at 40Mhz that supports VME A24 address decoding and a 16-bit VME slave interface. Programming of the Altera FPGA is accomplished via a front-panel JTAG interface.

3.6 Phase Lock Loop (PLL) Module

The PLL Module uses a voltage-controlled crystal oscillator to phase lock to a high-Q super-conducting RF cavity, which is then used for testing and commissioning. The operators provide amplitude and phase information to the FPGA through either a remote VME interface or local optical encoders. The FPGA uses the phase and amplitude information to derive in-phase (I) and quadrature (Q) values. I and Q are calculated by multiplying the amplitude times the sine (or cosine respectively) of the phase. According to the input values, the FPGA looks up the corresponding sine, cosine, and amplitude values in FLASH memory and then performs the appropriate multiplications. The resulting I and Q values are then presented to the PLL circuit through two 14-bit DACs. An eight channel ADC is also connected to the FPGA to provide read-back voltages throughout the PLL circuit.

The module is a 4-layer 6U board and uses an Altera FLEX10K50 FPGA operating at 10Mhz that supports VME A24 address decoding and a 16-bit VME slave interface. Programming of the Altera FPGA is accomplished via a front-panel JTAG interface.

4 CONCLUSIONS

FPGAs have become an indispensable component in VME board design at Jefferson Lab. Their small size and high logic density make them ideal for VME based data acquisition and controls. The ability to re-configure FPGAs through a JTAG interface provides the designer the ability to incorporate new features into existing design after fabrication and installation. The use of FPGAs and associated embedded logic will continue to be integrated into future designs as they develop. FPGAs will play a major role in the development of the new Low Level RF systems at Jefferson Lab. Digital down-conversion techniques and DSP filtering algorithms are ideal for embedding in devices such as FPGAs. Newer FPGA devices are becoming available with embedded processors surrounded with dense high-speed logic. These advances in technology will make FPGA applications almost limitless.

REFERENCES

[1] M. Keesee, “The Timing Synchronization System At Jefferson Lab”, ICALEPCS 2001, November 2001.

[2] H. Dong, “A Dual Digital Signal Processor VME Board For Instrumentation And Control Applications”, ICALEPCS 2001, November 2001.

TUAP054
physics/0111164

SAN/AFS: DEVELOPMENTS IN STORAGE DATA SYSTEMS ON FRASCATI TOKAMAK UPGRADE

A. Bertocchi, G. Buceti, C. Centioli, F. Iannone, M. Panella, V. Vitale
Associazione Euratom/ENEA sulla Fusione,
C.R. ENEA Frascati – Rome, Italy

Abstract

In the last three years, the architecture of Frascati Tokamak Upgrade (FTU) experimental database has undergone meaningful modifications, data and codes have been moved from mainframe to UNIX platforms [1] and AFS (Andrew File System) has been adopted as distributed file system.

A further improvement we have added regards the data storage system; the choice of SAN (Storage Area Network) over Fiber Channel, combined with power and flexibility of AFS, has made data management over FTU very reliable.

Performance tests have been done, showing a better transfer rate than in the previous system, based on JBOD modules with SCSI connection.

1 INTRODUCTION

During the last year FTU, Frascati Tokamak Upgrade, a compact high magnetic field tokamak, has reached the complete working configuration with three auxiliary heating systems to investigate radio frequency power deposition. It operates producing 20-25 shots per day, each shot being 1.5 s. long and producing presently about 30-40 MB of data for 1400 channels. This paper will discuss the FTU data distributed architecture and the storage systems solution.

2 FTU STORAGE SYSTEMS

The IBM-mainframe to UNIX migration of data and analysis software in FTU has been done trying several storage systems and server hardware platform. The main item of new operating environment was to adopt a scalable file sharing in multi-platform data processing.

Initially the 150 Gbytes of FTU data shots was stored on the RAID Array 450 of DIGITAL UNIX using RAID 5 raidset to have a proper data reliability and a good transfer rate in read data. Unfortunately, this solution was not performant under shared file system (AFS 3.5/6) because of the insufficient memory cache of the raid controller (32 Mbytes) and a very poor transfer rate in written data. We substituted the RAID array with a SCSI JBOD (*Just a Bunch of Disks)* type IBM 36 Gbytes 10000 rpm and Seagate 50 Gbytes 7000 rpm, both SCSI Ultra Wide. No raidset level was configured for that storage system to require a good transfer rate, in spite of the fact that losing redundancy means low data reliability. Subsequently we evaluated a new storage product considered as one of the hottest topics in IT: Storage Area Network (SAN) over Fibre Channel (FC).

SAN, a high-speed, high-performance network, permits multi-platform servers with heterogeneous operating system to access to multi-vendor storage devices as equals; FC allows to attach up to 126 devices over a range of 10 kilometres (in optic fibre) with ~100 MB/sec data transfer rate, exceeding SCSI limitations.

2.1 RAID systems

We replaced the provisional JBOD storage system with a Dot-Hill's SANnet storage system that offers a family of advanced disk storage products for open systems servers.

SANnet 4200 is an external hardware RAID storage system that can be used with stand-alone servers, server clusters, or as part of a SAN. It includes:

1. Two FC host channels providing throughputs ranging from 100 MB/s in single host server to 200 MB/s for clustered host configuration and five 80 MB/s Ultra SCSI (LVD) disk drive channels
2. Four FC connections that support single or multiple servers simultaneously and are compatible with many server platforms including Unix, Linux and Windows
3. A capacity from 2 to 10 disk drives standard, expansible up to 50 disk drives with expansion system
4. A storage capacity that ranges from 18 GB to 9 TB and support disk drive of: 9.2, 18.4, 36.7, or 73 GB with rotational rate up to 10000 RPM and 180 GB with 7200 RPM

In addition it allows:

1. High reliability because of redundant hot-swap removable components: disk drivers, RAID controller, battery backups, event reporting cards, power supply and cooling fans.
2. Single or dual redundant (active-active) RAID controllers with RAID levels: 0, 1, 0+1, 3, 5, 10,

30 and 50; RAID controllers can be equipped up to 1024 MB of cache memory per system (512 MB per RAID controller in active-active redundant FC networks) with automatic fail-over and fail-back.

A SANnet 4200 can be configured by first creating Logical Drives (LD), then partitions and lastly mapping host LUNs (Logical Unit Number).

SANnet 4200 is a high-reliable system that supports three levels of redundancy: 1) spare drive, 2) active-active redundant controller, 3) multi-servers operation.

At first level, hot swapping is supported through automatic disconnection of a failed drive and detection of a reserved Local (to serve a specified LD) or Global (to serve multiple LD) Spare Disk, followed by automatic background rebuilding of data.

At the second level, the active-active redundant controller mean is the back up mechanism in case of failure of the primary RAID controller and it becomes necessary for the redundant controller to take over all servers servicing control.

The last level can be achieved in a switched topology: a redundancy of servers means that a RAID controller - in point-to-point otherwise loop - will attempt to login a public loop and the partitioned LD can be simultaneously mounted by many servers.

2.2 FTU Storage architecture

Presently, we have collected about 200 GB of experimental data, including 140 GB of FTU data (raw and processed) and 60 GB of users data. To storage them we have installed two RAIDs SANnet 4200 in *single controller* configuration, each one with ten 73 GB (UW SCSI LVD, 10000 RPM) Seagate disks for a total capacity of 1.46 TB. Each RAID controller has 256 MB of cache memory.

We adopted a *point-to-point otherwise loop* topology, with four SUN ULTRA 10 Servers (CPU UltraSPARC-Iii, 440 Mhz, 256 MB of RAM and SunOS 5.8) with FC Host Bus Adapter Emulex LP 8000 in DE9 layout (copper duplex cabling with DB9 connectors).

We configured the SANnet systems in RAID 5 level with one Global Spare Disk for each RAID. It has been created two Logical Drives with capacity of 280 GB and 209 GB respectively, each one mapped on two different host channels. In this way, each server mounts a disk partition of about 200 GB and it is possible to achieve a full throughput of 200 MB/s for each RAID controller. An Automatic Tape Library – ATL – (15 AIT Cartridge Treefrog of Spectralogic) with differential SCSI adapter is used for cold data back up. The ATL has a maximum native capacity of 375 GB and a maximum compressed capacity of 975 GB. Presently SCSI adapter directly connects the ATL to a server, but it is possible to move to FC link and to connect to FC-switch. In this way, all servers can back-up their data through SAN without interfering with the traffic on the LAN.

Fig.1 shows a schematic block diagram of FTU storage systems.

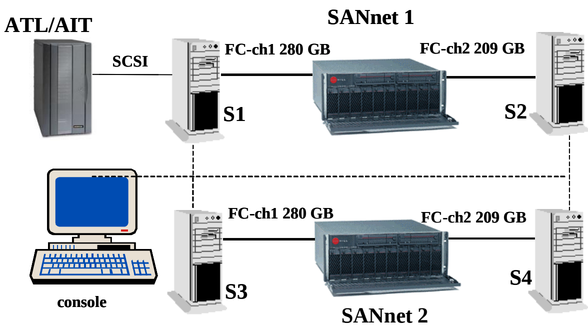


Figure 1: FTU Storage system

To show how the reliability has a cost, it should be noted that at the first level of redundancy, between Global Spare Disk and RAID 5 level, the native storage capacity of 730 GB of each RAID decreases to 489 GB with a capacity lose of 33 %.

Furthermore, to upgrade to the second level of redundancy, where it is adopted a dual RAID controller configuration, it is needed to add a secondary controller with 512 MB of cache memory and to expand to 512 MB the cache memory of primary controller, with a cost that is about 60 % of RAID system in single controller configuration.

2.3 Bechmarks

To confirm the right choice of new architecture, we have carried out transfer rate tests on two different systems. Benchmarks consisted of moving data volumes including 100 shots (130124 files and 6057 folders, about 1.52 GB of data) from one directory to another and vice versa.

Table 1 reports test results; the RAID system based on SANnet 4200 has a better transfer rate (between 26% and 31%) than the previous one, based on JBOD modules with SCSI connection.

Table 1: Transfer rate benchmarks

Storage System	**Move A -> B** (mm:ss)	**Move B -> A** (mm:ss)
JBOD (#2 36 GB, 10000 rpm, UW SCSI IBM; #2 50 GB, 7200 rpm, UW SCSI Barracuda Seagate)	35’ 55”	35’ 10”
SANnet (#20 73 GB, 10000 rpm, UW SCSI3 Seagate)	26’ 39”	24’ 24”

3 DISTRIBUITED FILE SYSTEMS

The adoption of AFS (*Andrew File System)* as client/server architecture allows to access files and directories on geographically distributed machines like a unique virtual machine, under the */afs* file system. An AFS client can approach FTU experimental data simply connecting under */afs*, where a single channel/file is addressed as:
/afs/fus/project/ftudati/das/hundred/number/family name/channel name
regardless platform, OS and real physical data location. Fig.2 shows the AFS *fusione.it* cell structure.

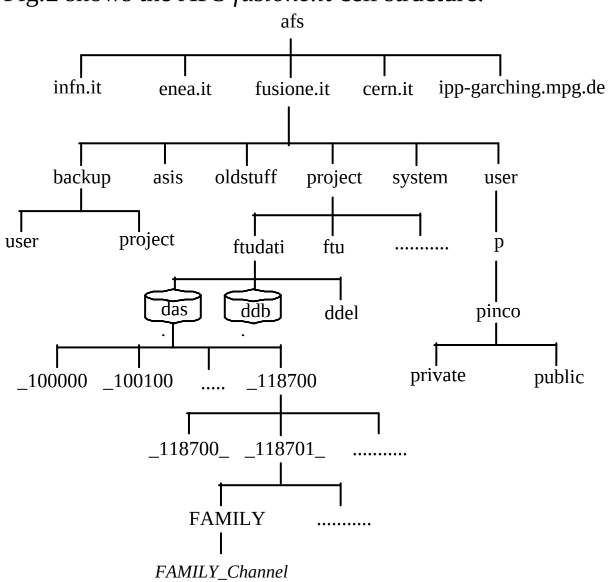


Figure 2: AFS *fusione.it* cell structure

Because AFS automatically replicates information across multiple servers within the enterprise to eliminate single points of failure and uses client-side caching which helps to reduce server and network loads, AFS provides uninterrupted access to data during routine maintenance, backup, or temporary system failures. Combined with powerful SAN features, this solution supplies full data availability.

4 CONCLUSIONS

From the lack of on line space and the need of tape robot in the mainframe environment, the fast change in the storage technology has moved the FTU archive to be completely on line. The flexibility of AFS has given us the chance to test several storage architectures until the adoption of SAN. The modularity of this solution allows us to face any predictable future space need for the FTU experiment.

REFERENCES

[1] Data analysis software on the FTU experiment and its recent developments, G. Bracco, G. Buceti, A. Imparato, M. Panella, S. Podda, G. B. Righetti, O. Tudisco, V. Zanza, **Fusion Engineering and Design 43 (1999) 425-432,** First Workshop of the IAEA Technical Committee Meeting on Data Acquisition and Management for Fusion Research, July 22-24, 1997, IPP Garching, Germany

TUAP055
physics/0111075

PLANS FOR THE SPALLATION NEUTRON SOURCE INTEGRATED CONTROL SYSTEM NETWORK*

W. R. DeVan, E. L Williams Jr., ORNL, Oak Ridge, TN 37830, USA

Abstract

The SNS control system communication network will take advantage of new, commercially available network switch features to increase reliability and reduce cost. A standard structured cable system will be installed. The decreasing cost of network switches will enable SNS to push the edge switches out near the networked devices and to run high-speed fiber communications all the way to the edge switches. Virtual Local Area Network (VLAN) technology will be used to group network devices into logical subnets while minimizing the number of switches required. Commercially available single-board computers with TCP/IP interfaces will be used to provide terminal service plus remote power reboot service.

1 INTRODUCTION

The Spallation Neutron Source (SNS) is an accelerator-based neutron source that will be used primarily for neutron scattering R&D. National laboratories participating in the project and their areas of responsibility are:

- LBNL: ion source and front end
- LANL: linac
- TJNAF: cryogenic helium liquifier (CHL) and linac cryomodules
- BNL: accumulator ring and beamlines
- ORNL: target and conventional facilities
- ANL: neutron instruments

In general, these labs have responsibility for the control systems corresponding to their technical system. ORNL has responsibility for implementing the control system communications network that ties these control systems together.

SNS will use the Experimental Physics and Industrial Control System (EPICS) for accelerator, target, CHL, and conventional facilities controls. This distributed control system for SNS has been dubbed the “Integrated Control System” (ICS). An integral and critical part of the ICS is a large-scale local area network dedicated to control system communications. This paper describes plans for the ICS communications network.

2 NETWORK OVERVIEW

The SNS control systems group will maintain four physical Ethernet networks. The primary network supports EPICS channel access and PLC communications. The remaining three networks support maintenance of the primary network. One of these provides “out of band” maintenance access to communication room switches. (Maintenance access to edge switches is handled “in band” as described later). There are also two networks to support “sniffing” of channels on the primary network.

A conventional structured cabling system is used. The main hub will be in the Front End (FE) Building (the first building constructed). Fiber optic “backbone” cables will radiate out from the hub to communication rooms (14 total) scattered throughout the facility. These cables will provide fibers for other SNS communication systems as well (e.g. the timing system and the administrative network).

A description of the primary network follows. A conventional hierarchical switch architecture is used. Gigabit Ethernet (1000BASE-SX and -LX) is used for backbone communications. Redundant core switches in the FE Building link to a layer of aggregator switches in communication rooms. Gigabit Ethernet is then extended from the aggregator switches to a third layer of access switches in the service buildings. These switches provide 10/100BASE-TX service to IOCs, PLCs, OPIs, and a fourth layer of switches. The fourth layer of switches services beam diagnostic devices and temporary connections. The network is fully switched; no repeaters are used.

Table 1 provides a summary of hosts and network drops connected to the primary network.

In the interest of minimizing the number of network switches, separation of subnets will be maintained by the use of 802.1q VLAN technology (vs. each subnet having its own set of switches). The primary control system network includes separate VLANs for accelerator, target, cryogenics, and conventional facilities controls. There is also a corresponding set of maintenance VLANs for accessing edge switch console ports and IOC console ports. (See more below).

* SNS is managed by UT-Battelle, LLC, under contract DE-AC05-00OR22725 for the U.S. Department of Energy.

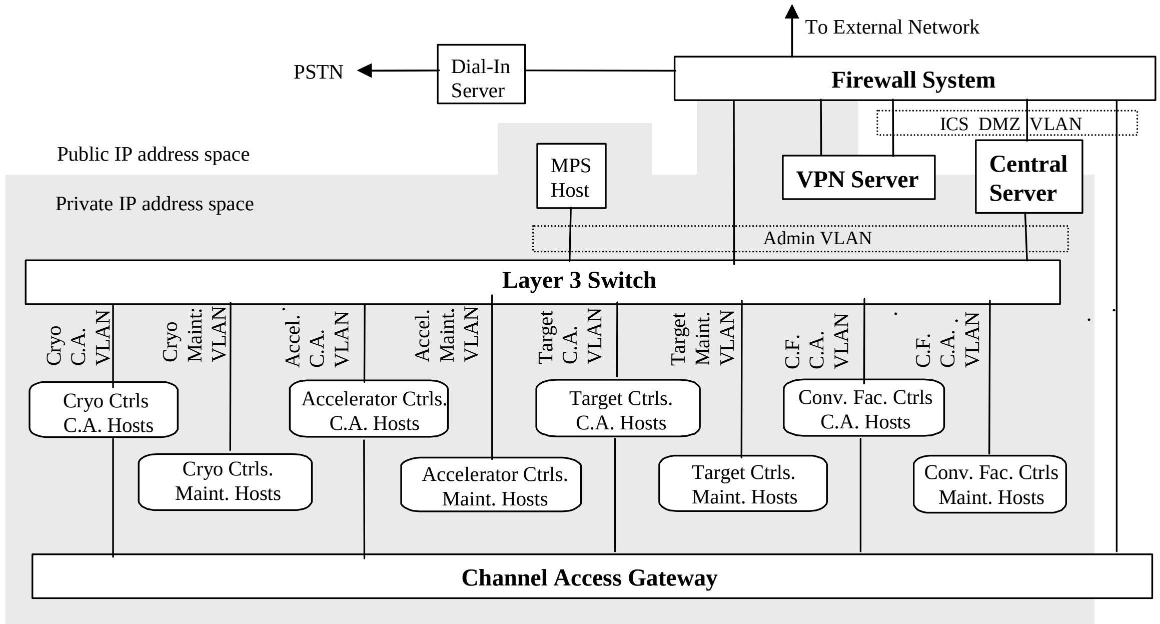


Figure 1: ICS Control System Network

Table 1: ICS Network Hosts and Ports

Host Type	Quantity
Operator Interfaces (OPIs) and Servers	38
I/O Controllers (IOCs)	178
Programmable Logic Controllers (PLCs)	118
Beam Diagnostic Devices	558
Misc. (Scopes, function generators, power monitors, etc.)	36
Terminal Server / Remote Power Rebooters	200
Utility network drops at IOCs and PLCs	180
Tunnel network drops	117
Total no. of ports	1425

To enhance reliability, the cryogenic control system subnet is physically separate from the other VLANs. This subnet will be able to continue operating should any non-cryo communications equipment fail.

The primary network's core switches are Cisco model 6509. Second and third layer switches are Cisco 3500 series. Fourth layer switches are typically Cisco 2950 series.

It should be noted that the ICS network performs no personnel safety functions. Nor is it directly responsible for machine protection. It is however utilized in the configuration of the machine protection system, and does provide some "defense in depth" for machine protection. Consequently adequate security must be provided to prevent unauthorized persons from interfering with these functions.

3 NETWORK MAINTENANCE

Remote terminal service is provided to network switch console ports in order to reduce "mean time to repair" of network problems. This service is implemented via Rabbit Semiconductor single board computers (at SNS commonly referred to as "Rabbits"). These boards feature a 10BASE-T Ethernet port, 4 digital inputs, 4 digital outputs, RS-232 port, and 512K flash memory. (These boards will also provide terminal and remote power reboot service to IOCs). A software library, including a telnet routine, is provided with the boards. The ORNL Hollifield Radioactive Ion Beam Facility (HRIBF) has modified the telnet routine to include a password-protected control sequence to initiate remote power reboot of the connected device. Remote power reboot will not be used for network switches until it is determined that the rebooter is more reliable than the switch.

An out-of-band network supports terminal service to communication room switch console ports. This out-of-band network has a "star" topology with a core switch in the FE Building. Backbone fibers provide connections from the communication rooms to the FE Building.

Terminal service to edge switches is handled "in-band". The RS-232 console port of each edge switch is connected to a Rabbit device. The Rabbit's ethernet port is connected to another nearby switch. Maintenance personnel can connect to the console port of the edge switch via a telnet connection to the Rabbit.

Rabbits will be connected to dedicated maintenance VLANs.

Two sniffer networks are provided for remote sniffing of switch channels. SNS has dedicated one port on each switch for port mirroring. For the first sniffer network, mirror ports on aggregator switches in communication rooms are connected directly to the FE Building communication room via backbone fibers. A gigabit Ethernet sniffer in the FE Building can then monitor any primary network gigabit Ethernet channel. For the second sniffer network, a similar architecture is used for sniffing 10/100BASE-TX channels. However, there are too many mirror channels in this case to directly connect every one to the FE Building communications room. Instead, each mirror channel is connected to the nearest communication room (via a 100BASE-TX link). Then one fast Ethernet connection is provided from each communication room to the FE Building. Maintenance personnel will need to manually patch in the channel they want to sniff.

The use of commercial network management software is planned but the software has not been selected yet.

4 ADMINISTRATIVE SERVICES

A design goal is for the control system to be able to operate SNS independently of any temporary loss of resources outside the ICS network perimeter. Consequently any administrative services required for network operations are included as part of the control system network.

Current plans are for redundant DNS and DHCP servers to provide name service and address management. A RADIUS server will provide authentication for the dial-in server and VPN access. LDAP will be used for account management.

Central servers will be backed up over a dedicated back-up VLAN. This requires all central servers to have at least two Ethernet interfaces, one for normal network traffic and another for back-up VLAN access.

5 NETWORK SECURITY

The ICS network will utilize a private IP addressing scheme per RFC 1918. A Class B IP address space will be provided. Hosts within the private address space will not be announced outside the ICS network perimeter.

A firewall will be provided to restrict access to the ICS network. Restrictions will be implemented via standard commercially-available firewall components (e.g. packet filtering, proxy service, and/or stateful inspection). Access control lists (ACLs) on the "Layer 3" core switches will be used to control access between VLANs.

EPICS offers a standard set of security features, including the ability to designate which hosts and/or users can access a given control parameter. To help prevent unauthorized access to control functions, SNS plans to limit direct channel access across VLANs. An EPICS "channel access (CA) gateway" will provide indirect access to control parameters across VLANs and to the outside world.

Remote access to the ICS network will be allowed but will be strictly controlled. Allowable methods of access include SSH access to central servers, VPN access, and dial-in service. SSH and VPN encrypt network traffic outside the ICS network perimeter, so passwords remain protected. Dial-in access will be password protected, monitored, and logged.

Due to the rapidly changing product offerings in the network security arena, specification of network security equipment is being put off as long as possible. We expect to have these devices specified and prototyped by Spring of 2002.

6 NETWORK RELIABILITY

The primary network's core switches are redundant. Links from the core switches to aggregator switches in communication rooms are redundant as well. The aggregator switches and all switches below that layer are not redundant.

For core switches and aggregator switches, two independent power circuits (one being UPS) will feed each switch. Edge switches will be fed from a single power circuit (UPS where available). Air conditioning to communication rooms is not redundant, so network switch temperatures will be monitored via a network management workstation in the main control room.

7 PLANS AND SCHEDULE

A prototype network has been implemented at the temporary SNS office building, with required features being added in an evolutionary fashion. It is planned for all required features to be prototyped at this office location before moving the network to the site.

Construction of SNS buildings has started and will continue through 2004. The first segment of the control system network will become operational in July 2002. Network equipment will be added as buildings become available. Completion of the ICS network is expected by the end of 2004.

TUAP056
cs.NI/0111027

UPGRADE OF SPRING-8 BEAMLINE NETWORK WITH VLAN TECHNOLOGY OVER GIGABIT ETHERNET

M. Ishii, T. Fukui, Y. Furukawa, T. Nakatani, T. Ohata, R. Tanaka
SPring-8, Hyogo 679-5198, Japan

Abstract

The beamline network system at SPring-8 consists of three LANs; a BL-LAN for beamline component control, a BL-USER-LAN for beamline experimental users and an OA-LAN for the information services. These LANs are interconnected by a firewall system. Since the network traffic and the number of beamlines have increased, we upgraded the backbone of BL-USER-LAN from Fast Ethernet to Gigabit Ethernet. And then, to establish the independency of a beamline and to raise flexibility of every beamline, we also introduced the IEEE802.1Q Virtual LAN (VLAN) technology into the BL-USER-LAN. We discuss here a future plan to build the firewall system with hardware load balancers.

1 INTRODUCTION

At the SPring-8 beamline, the network system is an indispensable element. The beamline VME system and workstations communicate with server workstations over the network [1]. The logging data of beamlines are accumulated via the network to the database. Beamline users can control an insertion device and beamline components such as monochromators, mirrors and slits from applications with a TCP/IP socket. Users can transmit experimental data to their home institutions via the network.

The high brilliance X-ray source in SPring-8 made various experiments possible. The large size data were generated as the result of measurement. For example, the data of 256Mbyte/min are generated in the time resolved measurement of 2-dimensional images. The number of users' computers on the BL-USER-LAN has increased because of the rapid construction of beamlines. The network traffic between the BL-USER-LAN and the OA-LAN has also increased. Now 38 beamlines are operational. For the infrastructure of user's experimental environment, the backbone network with high performance was required.

Beamline users connect the computing system and measurement system to the BL-USER-LAN. The IP segments of each beamline were separated. However we could not protect a beamline, e.g.BL01B1, against incorrect packets from other beamline, e.g.BL02B1, if a computer with an IP address of BL01B1 connected with the BL-USER-LAN at BL02B1. We required a BL-USER-LAN with the independency of each beamline.

2 ORIGINAL NETWORK

The backbone of the accelerator control network (SR-LAN) is the FDDI [2]. The BL-LAN is separated into four network segments, as A,B,C and D-zone and connected to the FDDI via routers with 10Mbps Ethernet. The backbone of BL-USER-LAN was optical fiber Fast Ethernet. Beamline users connected to the backbone with 10Mbps Ethernet. The OA-LAN is separated into four network segments by a router. A firewall system was introduced to protect the network security at the SR (BL)-LAN and the BL-USER-LAN. The firewall system in the SPring-8 control system consists of five firewalls. One master firewall in the central control room manages other slave firewall modules distributed in the experimental hall. We define the firewall rule that the SR (BL)-LAN is the clean zone, the BL-USER-LAN is the demilitarized zone for experimental users, and the OA-LAN is the public zone for the Internet.

The BL-USER-LAN is separated into 65 IP segments. One is for network management, 62 for all the beamlines, and two for beamline management staff. One C-class IP address is assigned for every beamline. When users' computers access the OA-LAN from the BL-USER-LAN, the firewalls translate the IP address with IP masquerade technique.

3 UPGRADE

3.1 Gigabit Ethernet

In the winter of 2000, to construct a high performance network system, we upgraded the backbone of BL-USER-LAN from Fast Ethernet to 1Gbps bandwidth optical fiber Gigabit Ethernet. Each beamline uplink connection to the backbone was upgraded from 10Mbps to 100Mbps at the same time.

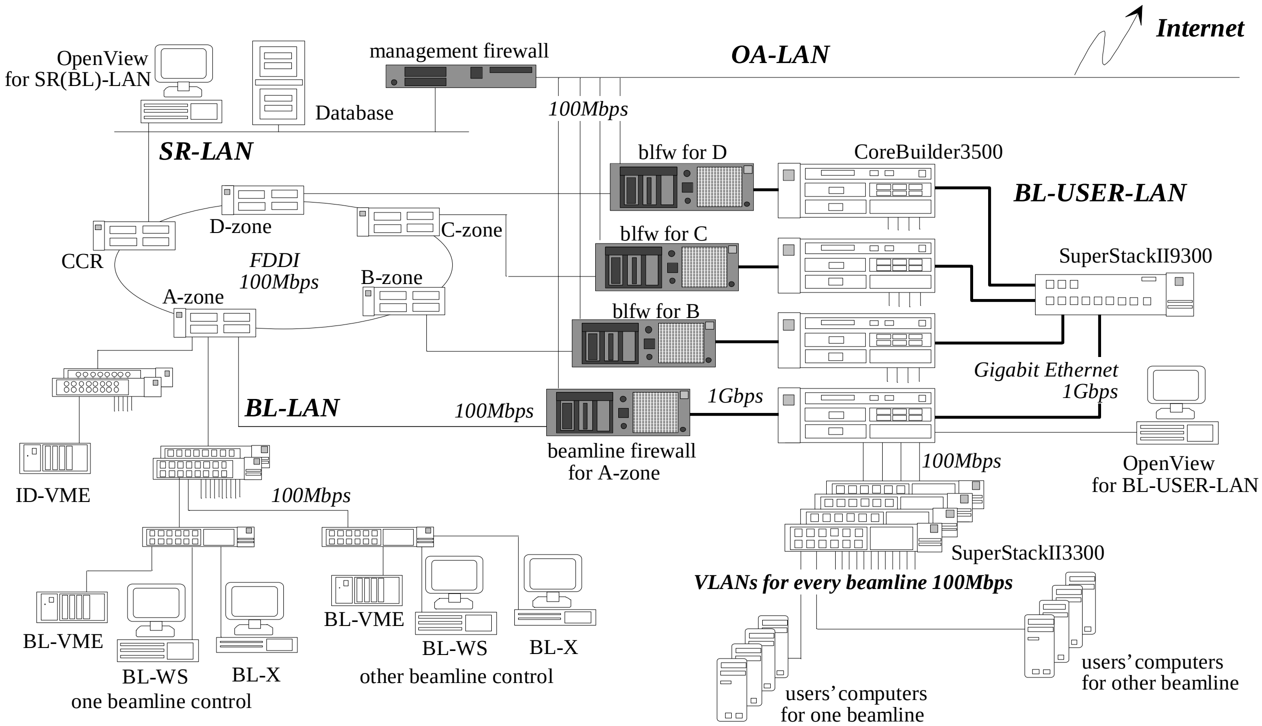


Figure 1: Schematic view of the SPring-8 beamline network

Figure 1 shows the schematic view of the SPring-8 beamline network. There are 500 or more computers and switches in the BL-USER-LAN. Each firewall controls the network traffic for up to 17 beamlines. Firewalls connect to the BL-USER-LAN with 1Gbps bandwidth. In the test bench of Gigabit Ethernet, a firewall throughput was measured to be 170Mbps [3].

From September 2001, the users' group of the laser-electron photon beamline began to transmit experimental data (100Gbyte/day or more) from their experimental system on the BL-USER-LAN to their home institute. We built the network infrastructure responding to the user's request.

3.2 VLAN in BL-USER-LAN

When we introduced Gigabit Ethernet, IEEE802.1Q VLAN [4] was applied to the BL-USER-LAN to build a secure and flexible network system.

The VLAN has a mechanism in which each port of a switch is assigned one group, a switch runs like two or more independent switches and each group belongs to an independent broadcast domain. To create many VLANs ranging over many switches, the VLAN-tag is set to the connection port between switches. The field that specifies a VLAN attribute is in an ethernet frame and the VLAN-tag information is set in this field. We can assign the physically separated computers to any VLAN without changing physical cable configuration.

We set a separated IP segment for each beamline with a unique VLAN. No packet from a beamline ever flows to any other beamline since each beamline belongs to an independent VLAN. With the introduction of VLAN, incorrect accesses or broadcast frames from other beamlines are completely shut out.

The introduction of VLAN made flexible network configuration possible. For example, it is easy to set a computing system at the 1km beamline to the same segment in the experimental hall. We only set the same VLAN to the port of switches without changing physical cable configuration.

3.3 Network Management

In the BL-USER-LAN, there are four Layer3 switches (3Com [1] CoreBuilder3500), a Gigabit Ethernet switch (3Com SuperStackII9300) and 32 Layer2 10/100Base-TX switches (3Com SuperStackII3300). We manage the BL-USER-LAN using an HP OpenView Network Node Manager[2] with SNMP in the central control room. We can set easily the VLAN by using Web browser to access the http server built in the switch.

[1] http://www.3com.com/
[2] http://www.openview.hp.com/

When network trouble arises, OpenView indicates the switch in trouble and the http server shows the port status of the switch. We can find the beamline network trouble from the central control room quickly because each beamline belongs to the unique VLAN.

4 PLAN

We recognize that the firewall system is not robust enough and it could be a critical point failure. If a firewall is down in the current network system, users of up to 17 beamlines will have serious damage to their experiment. It will take several hours to recover.

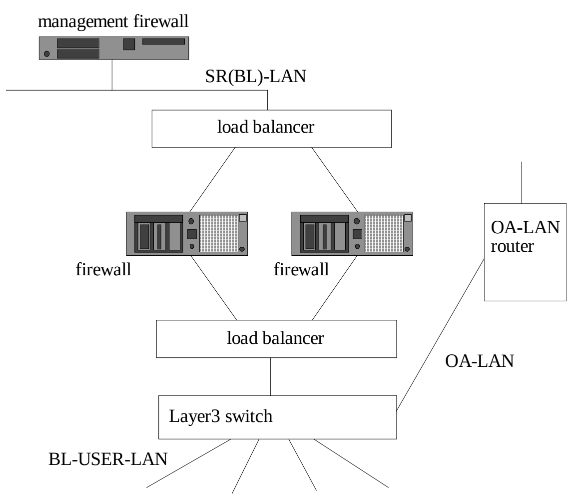


Figure 2: Schematic diagram of the firewall system with hardware load balancers

We are planning to build a redundant firewall system. Figure 2 shows the schematic diagram of the firewall system with hardware load balancers. In this scheme, two firewalls are active and the load balancers distribute packets. Two load balancers check the status of two paths through each firewall. Even if one side path is down with firewall trouble, the network is alive because the load balancer transfers all packets to the effective path.

The layer3 switch in Figure2 connects the SR(BL)-LAN, the BL-USER-LAN and the OA-LAN. The Layer3 switch is required to reject an access from the OA-LAN to the BL-USER-LAN with certain security. The required specifications for the Layer3 switch are as follows:

- We can set up 66 or more VLANs and interfaces.
- 64 or more entries of IP masquerade are possible.
- We can define IP masquerade by the destination of routing.
- The redundant configuration of the management module and the power supply are available.
- Link aggregation between the switches is available.

The access speed from the BL-USER-LAN to the OA-LAN will become higher than the present network system.

5 SUMMARY

The Gigabit Ethernet was introduced to meet the increasing network traffic. The VLAN was introduced to establish the independency and flexibility of BL-USER-LAN. The BL-USER-LAN has turned into a system with high speed, secure and flexible.

A future network system will have to support the redundant configuration. We will design the network system including firewall load balancing and high availability.

REFERENCES

[1] T. Ohata *et al*., "SPring-8 beamline control system", ICALEPCS'99, Trieste, Italy, 1999, p.666.
[2] T. Fukui *et al*., "Design and performance of the network system for the storage ring control at SPring-8", ICALEPCS'97, Beijing, China, 1997, p.312.
[3] SPring-8 Annual Report, 1999, p.22.
[4] http://www.ieee802.org/1/

TUAP057
astro-ph/0111142

ACTIVE OPTICS CONTROL OF THE VST TELESCOPE WITH THE CAN FIELD-BUS

D. Mancini, P. Schipani, G. Mazzola, L. Marty, M. Brescia, F. Cortecchia, F. Perrotta, E.Rossi
Osservatorio Astronomico di Capodimonte, Napoli, Italy

Abstract

The VST (VLT Survey Telescope) is a 2.6 m class Alt-Az telescope to be installed at Mount Paranal in the Atacama desert, Chile, on the European Southern Observatory (ESO) site. The VST is a wide-field imaging facility planned to supply databases for the ESO Very Large Telescope (VLT) science and carry out stand-alone observations in the UV to I spectral range, [1]. This paper will focus on the distributed control system of active optics based on CAN bus and PIC microcontrollers. Both axial and radial pads of the primary mirror will be equipped by astatic lever supports controlled by microcontroller units. The same CAN bus + microcontroller boards approach will be used for the temperature acquisition modules.

1 VST ACTIVE OPTICS GENERAL CONCEPTS

The active optics of the VST telescope is based on a complex control scheme; both primary and secondary mirrors must be supported and accurately positioned.

The aberrations corrected by the VST active optics control system are essentially the coma, defocus (secondary mirror re-alignment) and the spherical, astigmatism, quad-astigmatism and tri-coma (primary mirror sag deformation) aberrations, [3].

Hereinafter the description will focus on the primary mirror active optics system.

The VST primary mirror figure is controlled by a set of active supports based on astatic levers. They are driven in order to elastically compensate wave-front deformations measured by a wave-front sensor in a feedback loop. The mirror is supported by means of axial and lateral pads.

VST has 84 axial pads distributed on four concentric rings, which include respectively from the center out, 12, 18, 24 and 30 pads, (figure 1). The axial pad distribution was analyzed by means of Finite Elements Analysis (FEA).

24 lateral pads provide the lateral support. The lateral component of the weight of the mirror is supported by the 24 lateral levers, when the altitude axis of the telescope is moved. The forces are applied at the center of the rim and the force vector lies in the neutral plane at the outer edge.

Each M1 mirror support system must be driven to a required force and this force must be monitored in a feedback loop scheme. The monitoring of the net force developed by each single actuator is obtained from a load-cell. The speed of the single motor is monitored with a tachometer.

The wavefront analysis system is based on a Shack-Hartmann sensor and is sensitive enough to work on guide stars of magnitude +14 for integration times of about 30 seconds. The size of the VST pupil sampling sub-apertures is on the order of 250 mm. The number of the Shack-Hartmann spots across the telescope pupil is equivalent to 10. The lenslet array shows a f/45 focal number and a lenslet diameter Dl ≈ 0.5 mm.

The Adapter/Rotator of the VST includes a sensing and guiding arm that provides the telescope guiding and the on-line wavefront analysis.

The guiding and sensing arm is coaxially mounted on a rotating support with the rotator and its optics are mounted on a radial rail. These two devices implement a polar system, which allows the arm to reach the center of telescope field during the off-line calibration phases.

The polynomial base used to fit the wavefront is determined by the elastic mode of the primary mirror itself. In this way the energy required to deform the primary mirror is minimized. The deformation of the primary mirror is obtained by variation of the overall pattern of forces applied by the astatic lever supports.

There are three different kinds of corrections to apply to the M1 sag through the astatic levers:

- compensation of the axial component of M1 weight (Passive force)
- correction of the optical distortion of M1 (Passive distortion)
- compensation of the optical distortion changing with altitude angle (Active distortion).

While the passive distortion is constant, both the passive force and active distortion depend on the altitude angle. The sag of the primary mirror can be corrected to eliminate the following aberrations:

spherical, astigmatism, triangular coma, quadratic astigmatism and fifth order coma.

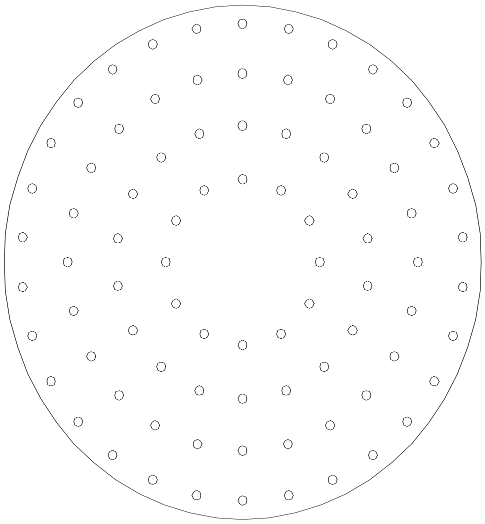


Figure 1: M1 axial pad distribution

2 M1 CONTROL CAN BUS

The VST M1 84 axial and 24 radial pads are controlled by a CAN bus network of astatic lever controllers connected to a CAN bus interface board in a Local Control Unit (Figure 2).

The Local Control Unit (LCU) is a VME bus computer equipped with a Motorola MVME 2604 CPU card. The CAN bus interface module is a TEWS Datentechnik TPMC816 2 channel CAN bus PMC module, mounted into the PCI mezzanine socket available on the Motorola CPU card. The CAN bus interface is based on Intel 82527 chipset. This board is the master of the CAN bus network and can address the slave modules (Astatic Lever Controllers) to send configuration parameters or to request data. The address mechanism is based on the acceptance filtering capability. All CAN implementations provide some hardware acceptance filters to relieve the microcontroller from the task of filtering the needed messages from those not of interest one. The address space managed by the acceptance filter is equal to 127 devices, which is sufficient for the VST active optics application, [3].

The LCU runs a VxWorks real time operating system. The standard ESO architecture is adopted both for the hardware and software architecture. The LCU running the active optics pad control software is one of the nodes of the overall VST telescope control network. This is composed by higher level coordination and control HP-Ux workstations and low level LCUs to directly interface with the controlled devices.

The physical layer is based on a twisted pair for data transmission. Two other wires are present to distribute 12V power to all modules connected to the network. The maximum data rate envisaged for the CAN bus is 1 Mbit/s.

The power for the slave modules electronics is provided by the CAN bus; one DC/DC converter for each module performs 12V/5V conversion. The power supply for the actuators, sensors and the electronics connected after the insulation barrier is distributed by a field power bus.

The slave modules contain a microcontroller that provides for communication protocol management and control of the hardware devices present on the module. The operational parameters are communicated from the LCU to the slave module, which locally implements the closed loop control without any other intervention of the LCU.

The driving of the array of actuators using a distributed system of local microcontrollers has some advantages, such as quick positioning loop, in place, with the highest allowed bandwidth, not limited by intercommunication mechanisms, [1].

A custom design was chosen for the astatic lever controller, because no commercial solution able to completely fulfill the application requirements was available.

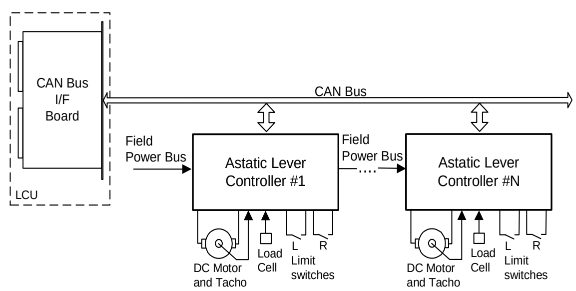


Figure 2: CANbus network configuration

The functional requirements for the Astatic Lever Controller are:

- Control of the applied force by linear regulators
- Control of the DC motor
- Acquisition of force from the load cell
- Acquisition of the end of travel switches status
- Acquisition of motor speed from the tachometer
- Implementation of the physical and data link layer according the ISO 11898
- Implementation of the data link layer according to the CAN 2.0 B standard
- Implementation of monitoring and debugging functions via CAN bus and local RS232 serial port
- Galvanic insulation of motors and sensors by opto-couplers and insulated DC/DC converters

Components of the Astatic Lever Controller are:

- Microcontroller
- CAN bus communication controller
- CAN bus interface drivers
- Opto-insulated Load Cell Interface
- Opto-Insulated Tachometer Interface
- Opto-Insulated DC Motor Driver
- Software Limit Switch Interface
- RS232 serial port for monitoring and debugging functions

The microcontroller platform selected to implement the custom modules is the PIC controller by Microchip, which is well supported by a development kit for CAN bus application.

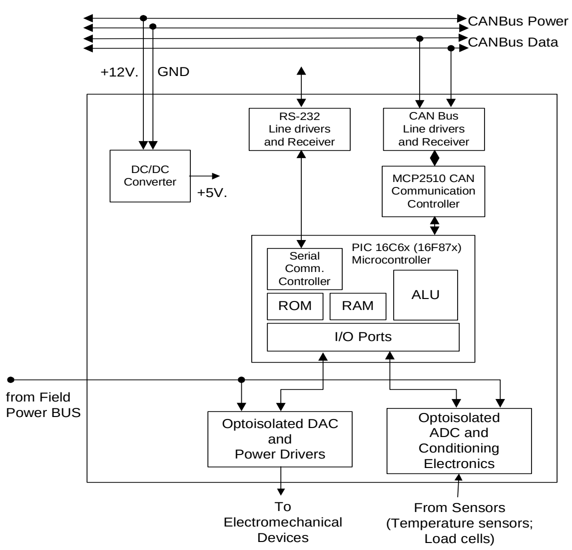


Figure 3: General architecture of the microcontrolled modules

The PIC component family chosen for the application development is the 16F87x, a flash memory microcontroller providing program memory range from 2k to 8k bytes. A component with flash memory is employed during the development activities because it allows for fast programming (no erasing process is needed) and the use of the in-circuit debugger, [3]. Both aspects speed up the development time. The implementation of the final version of the firmware will use the 16C6x OTP (One Time Programmable) microcontroller family. This family is equivalent to the flash component. The OTP components are cost effective and less subject to memory erasing problems.

All the firmware is written in ANSI C language.

A general architecture/diagram of a microcontrolled module for CAN bus is shown in Figure 3.

The microcontroller integrates an ALU, ROM for firmware storage and RAM for program execution. The CAN bus communication controller for protocol management is the MCP2510 by Microchip. This controller implements full CAN 2.0 A and B, at a transmission rate of 1 Mbit/s. The following components are also present in the module: CAN bus and RS232 line drivers, opto-insulated driver for motors activation, opto-insulated serial ADC and conditioning electronics to interface external analog sensors such as the load cells.

3 REFERENCES

[1] Mancini, D., Sedmak, G., Brescia, M, Cortecchia, F., Fierro, D., Fiume, V., Marra, G., Perrotta, F., Rovedi, F., Schipani, P.,: 2000, “VST project: technical overview”, in “Telescope Structures, Enclosures, Controls, Assembly / Integration / Validation, and Commissioning”, eds Sebring, T., Andersen, T., SPIE, 4004, 79

[2] Fantinel, D., Bortoletto, F., Pernechele, C., Gardiol, D.: 2000, “TNG: the active optic system”, http://www.tng.iac.es

[3] Mancini, D., Sedmak, G., Brescia, M, Cortecchia, F., Fierro, D., Fiume Garelli, V., Marra, G., Perrotta, F., Schipani, P.,: 2000, “VST Final Design Review”, http://vst.na.astro.it

TUAP059
hep-ph/0110167

MAIN INJECTOR LCW (LOW CONDUCTIVITY WATER) CONTROL SYSTEM

K. C. Seino, FNAL, Batavia, IL 60510, USA

Abstract

There are six service buildings uniformly spaced along the perimeter of MI (Main Injector). A total of 18 LCW pumps were installed around the MI ring with 3 pumps per building. Approximately 8,000 GPM of LCW is required to cool magnets, bus and power supplies in the MI enclosure and service buildings. In each service building, a PLC control system controls pumps and valves, and it monitors pressures, flow, resistivities and temperatures. The PLC hardware system consists of a Gateway module and a variety of I/O modules, which are made by Sixnet of Clifton Park, NY. The control system communicates with other buildings including MCR (Main Control Room) via an Ethernet link and front-end computers. For more details of the MI LCW control system, refer to [1] and [2]. One of the key elements of the PLC software is called ISaGRAF workbench, which was created by CJ International of Seyssins, France. The workbench provides a comprehensive control-programming environment, where control programs can be written in five different languages. For more details of ISaGRAF, refer to [3].

1 HARDWARE

1.1 PLC

The PLC hardware system consists of a Gateway module (local controller/remote communicator) and a variety of I/O modules. The product line of these modules is called as Sixtrak and is made by Sixnet of Clifton Park, NY. The Gateway module can be connected up to 20 I/O modules without expander modules, and it can be connected up to 128 I/O modules with expanders. The Gateway scans I/O modules and updates the information on them under a program control, and it communicates with host computers via an Ethernet link. The Gateway has 1 Mbytes of flash memory (for firmware), 256 Kb of battery backed RAM (for data), a real time clock and a RS232 serial port for local diagnostics.

For our applications, we are using several different types of I/O modules. They are analog input modules (4-20ma input, instrumentation and RTD), analog output modules, digital input modules (5VDC and 24VDC) and digital output modules (relay).

The modules are DIN rail mountable for instantaneous installations, and they can be removed from their wiring bases for easy installation and maintenance. Moreover, every module is isolated from a common bus and other modules for fault-free operations.

1.2 Instrumentation

The sensors, transmitters, local indicators, actuators and other field devices had been specified, purchased and installed by the Mechanical Support group. However, we did cabling/wiring, set key parameters, calibrated, did some adjustments and did trouble-shooting on these field devices. For this reason, I will briefly make reference to them as follows.

(1) Pressure Transducers (Measurement Specialties MSP-400-P-0064A, 0-400 psig), (2) Flow Transmitters (Peek Measurement 2120R, 0-100 GPM), (3) Differential Pressure Transmitters (Rosemount 2024), (4) Liquid Level Indicators (Gems SureSite 86158), (5) Resistivity Meters (Thornton 200CR, 0.0013-50 Mohm-cm with 0.1 constant cell), (6) RTD Probes (Devar #RTDSF2.5), (7) Electric Valve Actuators (Keystone EPI-13, 1/4 turn (close/open)), (8) Electric Valve Actuators for Temperature Regulation (EIM 2000 M2CP, 1/4 turn (continuous)), and (9) Controller for EIM2000 above (Powers 535, PID control).

1.3 Cables

The cables were pulled in MI<10:50>, 52 & 60 in 10/97 and 11/97. Since then, we had spent about ten months on an inconstant basis to terminate the cables, doing tests and fixing problems on the cables, the instrumentation and the PLC hardware for a total of nine systems excluding one for CUB.

We used several different types of cables -- (a) Belden #8761 for general use, (b) Belden #8719 for high voltage, (c) Belden #9533 for RTD, (d) Belden #9886 for 450 FT travel from MI-40 service building to Beam Dump, (e) Belden #8760 for Beam Dump, (f) Omega KK-J-20 for Type-J TCs (thermocouples) and (g) Omega KK-K-20 for Type-K TCs.

2 SOFTWARE

2.1 Sixnet Plant Floor

Plant Floor is a configuration and maintenance tool for Sixtrak I/O systems. Using Plant Floor's windows and menus, one assembles a graphic representation of each I/O system. Configuration choices let him customize each module in a given system. Once a system configuration is complete, the configuration is downloaded to the Gateway module of the system. Plant Floor is also a calibration tool for analog I/O modules, and it provides real-time displays for the maintenance and diagnostics on I/O modules.

2.2 ISaGRAF

ISaGRAF is a comprehensive control-programming environment that makes Sixtrak I/O a high performance, yet inexpensive controller. ISaGRAF uses standard industrial PLC programming methodologies for designing powerful applications without the need of high-level computer languages. ISaGRAF is created by CJ International of Seyssins, France, and it is sold by Sixnet as a part of the Sixnet software package.

An ISaGRAF project is a collection of individual programs and functions that form a complete control application. Each program controls one particular part of the application. In February 1993, responding to the need for standards to reduce training costs and guarantee portability, the IEC issued the IEC 1131-3 standard: a specification of five PLC languages that can be mixed in the same application. The five languages are (a) Sequential Function Chart (SFC), (b) Function Block Diagram (FBD), (c) Ladder Diagram (LD), (d) Structured Text (ST) and (e) Instruction List (IL).

An ISaGRAF project consists of programs, sub-programs and functions, which are placed in four different sections as follows.

(1) Beginning: Programs in this section are systematically executed in the beginning of a cycle after updating external inputs and outputs.

(2) Sequential: Programs in this section are executed confirming to the SFC rules and the implementation on a series of steps and transitions.

(3) End: Programs are executed at the end of a cycle just before updating external outputs.

(4) Functions: Sub-programs which can be called by programs in any of the other three sections.

The ISaGRAF dictionary is simply the collection of internal, input and output variables and defines that are used in the programs of a project. Variables and defines are specified as Local (specific to one program), Global (in any program within a project) or Common (in any project in ISaGRAF), when they are created in the dictionary. The Sixtags utility (a shared tag database): The I/O and module tag names that are created in the Plant Floor configuration can be exported to the dictionary of an ISaGRAF project by the Export command of Sixtags.

The ISaGRAF I/O simulator can be run by clicking on Simulate in the Debugger menu. The simulator allows us to try out a program before it is run on a live system. This convenient tool saves time by discovering problems and fixing them before a real start-up. I/O variables can be locked (disconnected) from their corresponding external devices. Once they are locked, their status/values can be altered by the debugger to proceed with the simulation.

Fig. 1 shows an example of the sequential program. For further details on ISaGRAF, one should refer to Ref. [1].

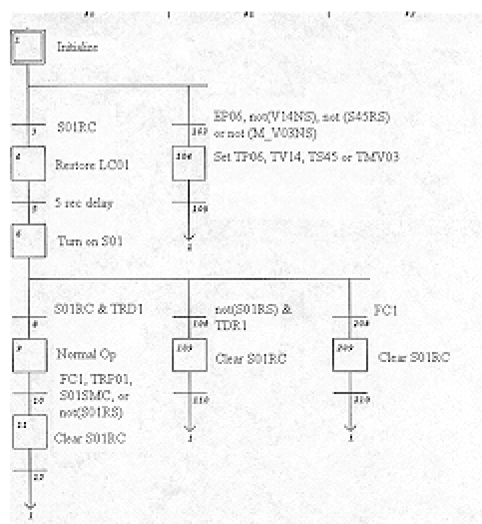


Fig. 1: Example of Sequential Program

2.3 Interlocks/Trips

A number of analog and digital signals are examined on the LCW control systems, and if certain conditions exist, the programs use them to interlock the pump/valve operations. For each of the above analog signals, upper and lower trip limits are specified, and if the signal exceeds these limits, the program interlocks the related devices.

At two times (Start-up and Run times), conditions are tested. At the start-up time, if certain conditions exist, the LCW pumps cannot be turned on by the console command, and the trips are set true. In this situation, the operator firstly removes the conditions and clears the trips. Secondly he sends the second command from the console to turn on the LCW pumps, because the first command has been cleared when the trips occurred.

During the run time, if certain conditions occur, the LCW pumps are turned off, and the trips are set true. In this situation, the operator follows the same procedure mentioned above.

There are two special rules to observe. One is 'Initial 15 sec Delay'. For 15 sec after the start-up, some conditions are ignored for tripping. The other is '15 sec Filtering'. Signals are sampled at a certain rate. If the conditions exist when sampled, the occurrences are counted. If the counts exceed their specified limits over the 15 sec periods, the LCW pumps are tripped off.

2.4 ACNET Connections

In each of the service buildings (MI-10, 20, 30, 40, 50, 52 & 60), the Gateway module communicates with the front-end computer of the house via an Ethernet link. The front-end in turn communicates with the console system via the Ethernet link.

There are a total of 9 front-ends that handle tasks related to the LCW controls. Some of them perform special tasks as well as communication. The MI-60 front-end performs a number of special tasks -- (a) it reads the status on four local valves and writes it at CUB for MI Makeup, (b) it reads the status on M_V03 (MI-60 Magnet V03) and writes it to all the other locations, and (c) it reads the status of the LCW pumps around the MI ring, examines it for a condition (if all the pumps are off or not) and uses the result for an action, and (d) it also writes the result to CUB. The front-ends at MI-52, MI-60 and CUB do calculations to convert the liquid levels to their volume equivalents.

2.5 Console Pages

The I56 page and its subpages show all the activities of the MI LCW system. The Global subpage shows an overall view of the pump status/controls, where the operator can view the status on LCW pumps, Pond pumps, transfer pumps and other pumps, where he can turn on pond pumps individually, and where he can turn on LCW pumps using a list (specifying the turn-on order and pauses between).

The MI-10 subpage shows a view of the MI-10 LCW system, which displays the status/controls on pumps, valves and PLC, analog readings, trip flags and limits. Fig. 2 is a graphic view of the MI-10 LCW system.

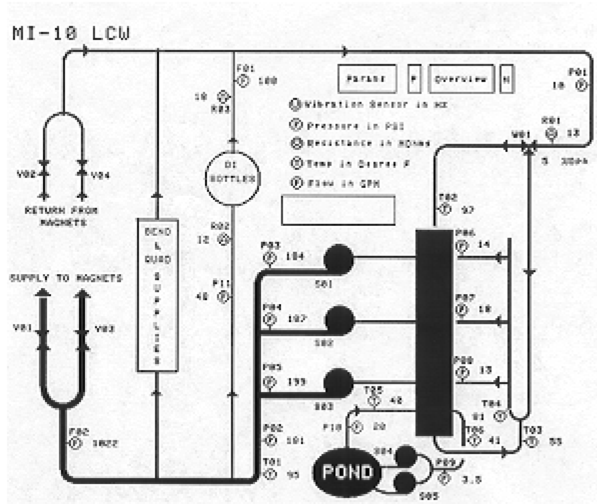


Fig. 2: Graphic View of MI-10 LCW System

REFERENCES

[1] K. C. Seino, "Main Injector LCW Control System", FERMILAB-TM-2085, June 1999.

[2] K. C. Seino, "CUB DI Column Control System", FERMILAB-TM-2086, June 1999.

[3] CJ International, "ISaGRAF Version 3.00 User's Guide", 1995.

NSLS CONTROL SYSTEM INTERFACE TO MODICON PLC

S.Ramamoorthy, S.Buda, J.Oliva, G.Ramirez, BNL, N.Y. 11973, USA

Abstract

The hardware engineering efforts at the National Synchrotron Light Source facility are currently geared towards the use of programmable logic controllers (PLCs) for the control of machine hardware. The PLC provides a cost-effective solution combined with easy maintenance and upgrade. Modicon control products from Schneider Automation Inc. have been chosen since a wide range of PLC's, inexpensive I/O modules and networking options are offered by the company. The NSLS Control Monitor software uses the Modbus/TCP which is a variant of Modbus protocol to communicate with the PLCs. This paper describes the software interface to the Ethernet attachment on the PLCs.1

1 INTRODUCTION

The PLCs have been extensively used in industry to automate a variety of control and monitoring functions for many years. They are flexible, versatile and are built to withstand harsh factory conditions. However, their role was confined to industries due to their slow response, slow processing power and lack of communication standards. Because of recent evolution towards PLCs with fast processors, a variety of compatible I/O products and communication adapters that support TCP/IP network, the PLCs have made a quantum leap into the arena of distributed-control systems for large accelerator facilities.

The control system at the NSLS has a two-level distributed architecture consisting of HP/c8000 series workstations for the high-level (the operator level) and 80 VME-based microprocessor subsystems (referred to as micros) for the lower level. The communication link is a high-speed Ethernet. The micros are responsible for the control and monitoring of the storage ring hardware. The equipment to be controlled, dictates the types of I/O boards for a micro. The I/O peripherals include ADC, DAC and digital IO cards in the micro crate.

Though fast and reliable, VME I/O cards are very expensive and may not be required when controlling and monitoring devices with slow response. PLC was considered as an alternative. The hardware engineers also preferred to replace the traditional digital circuit hardware with PLC. This will allow them to implement logic at the equipment level without the help from the software group. Future upgrades and maintenance will be very easy.

2 DEVELOPMENT OF PLC-BASED CONTROLS AT NSLS

The decision factors when selecting the Modicon products were cost, technical support, availability of a wide range of inexpensive I/O modules and networking solutions. The introduction of MODBUS-TCP protocol combined with the networking hardware by the Schneider Corporation, has greatly simplified the interface between the control system and the PLCs.

At NSLS two types of processors are used. One is the TSX Quantum Controller. The logic is stored and run in the Quantum controller. The processor communicates with I/O module using the Modbus Plus protocol. An Ethernet module connected to the processor via quantum back-plane provides the link to the control system. The other processor is from the family of TSX Momentum products which include processor adapters, communication adapters, option adapters (to provide the processor with additional networking capability), and a variety of I/O modules The modular design of the adapters and the simple plug-in and wiring of the products allow easy integration of a system that meets the NSLS requirements.

The Concept Software is used for PLC program development. This complies with the Microsoft Windows GUI interface and the IEC 1131-3 standards for PLC programming. The Concept software provides a great development and debugging environment and generates documentation for the programmers.

3 INTEGRATION OF PLCS TO THE NSLS CONTROL SYSTEM

The operations at the NSLS facility rely heavily on a set of high-level programs that run on workstations. These programs access the various analog and/or digital hardware parameters and software variables by meaningful names (for example Uvrf1trip, Xraycurrent, etc).

The complexity and diversity of various hardware used in the facility make each micro unique at the

[1] Work performed under the U.S. DOE Contract No: DE-AC02-98CH10886

hardware control level. However, the real-time software (referred to as NSLS Control Monitor [1] and is executed by all the micros), presents a standard interface to the high-level programs. The monitor views all I/O signals in any micro as a set of devices with definite types. The types define a device as analog IN/OUT or Digital IN/OUT or a combination of analog and digital or as an array. A standard set of commands (READ/SET) has been defined for access by high-level programs. The application modules in the micro interpret the commands and operate on the hardware.

3.1 PLC as a remote Slave I/O

To fit into the framework of NSLS controls, the PLC is treated as a remote slave I/O to a micro, the physical link being the Ethernet. Using the TCP/IP protocol, the micro communicates with the server that resides in the firmware of the PLC's Ethernet interface. The server uses the registered port 502. Once the client makes the connection to the server, it can periodically send messages to the PLCs. The server automatically closes the connection if it is inactive for more than 10 seconds. The server is normally designed to support multiple connections (5 to 8) by the vendors.

3.2 Data transaction between PLC and Micro

The underlying protocol for the data transaction is called Modbus Protocol. It uses a query-response cycle between a master and the slave devices. The query is normally initiated by the micro. Every query gets a response from the PLC. The request contains the address of a register, a function code and data items, if any. The basic function codes are Read or Write a single register or multiple registers. The response function code indicates whether the reply has valid data or an error code. The protocol used over the Ethernet is called Modbus TCP/IP. It basically inserts a MODBUS frame into a TCP frame and sends it as a message. The information for packing messages is available in MODBUS/TCP specification manual [2].

The PLC logic is responsible for data acquisition and control of the hardware in response to messages from the micro. It allocates a block of 16-bit wide registers (referred to as 4x registers in Modbus PLC language) for data exchange with the micro. During the execution of its logic cycle, the digitized data are stored in the registers. All discrete inputs are packed into 16-bit words and stored. The micro can read a maximum of 125 registers in one message transaction. This reduces the network traffic considerably. To write set-point data or On/Off control commands to a hardware the micro sends the data or command to a register assigned by the PLC logic. The PLC logic executes the commands, which can be "setting a DAC" or "turning a supply ON/OFF" etc.

4 PLC DRIVER SOFTWARE

The driver supports multiple clients. The maximum number of clients has been arbitrarily chosen as 5. It can be extended if required. The multi-client driver is useful for the following. (1) The maximum number of registers for one Modbus transaction is 125. If the number of I/O signals requires more than 125 registers one can use multiple clients for data access from different register segments. (2) In some micro systems, more than one PLC unit with different IP addresses may be used. The micro can concurrently communicate with all the PLC units by setting up multi-clients. The driver provides three utility functions to the micro application module. All the functions return a status code and the data, if any. A non-zero status code indicates failure. The value of the code describes the type of the error.

4.1 Init Function

The application should first invoke this function with the following parameters: the upper and lower limit for the registers that will be used for data exchange, IP address of the PLC and the polling period. The application program specifies the frequency at which the “read register” command is to be sent to the PLC. This is based on the response time of the hardware and the PLC logic cycle time. The function checks the validity of the parameters. A successful **init** call will spawn a client task with the same priority as the application module.

4.2 Read Function

The application module uses the **Read** function to retrieve the data from any register range so long as it is within the declared set. The application should first test the status code for error before updating the data field within the device record. There is no need to call this function faster than the polling rate.

4.3 Write Function

When a high-level program sends a command (analog or digital or an array) to a device, the application module encodes the command in an appropriate format and invokes the write function to post the information to the client task. The first register number, number of registers and the data items should be specified.

4.4 Client Task

The client task opens a TCP connection to the specified PLC server. If the connection is successful, the client will periodically send a "Read Multiple

Register" request for the specified registers. The response and error status, if any, will be stored in the memory accessible by the application module. In order to keep the connection alive, the maximum delay should not exceed 5 seconds. The task checks whether any new SET message (ON/OFF, set-point etc) has been posted by the application module. This is checked at a frequency of 50 Hz. The message is formatted using the specified register number and the data and sent to the PLC. Set messages always take precedence over Read messages. The client uses the "select call" with 5 seconds time-out before reading the socket. If the PLC is turned off or the physical link is lost the task closes the connection and retries again.

5 MICROS WITH PLCS

1. The UV beam line status micro continuously monitors the status of the safety shutters, photon mask, vacuum valves etc. for the 16 beam lines in the UV ring. It generates a real-time display on a local CATV and reports alarm conditions to the Alarm Handler. The interlock system for the beam line user stations consists of PLCs to operate the vacuum and fast valves, the photon mask and the safety shutter. The sequence of operation and interlocking of these components is controlled by PLC logic executed in a local PLC-CPU. The local processor for each beam line communicates with others via Modbus Plus communication bus. The Modbus Plus bus is also connected to a master PLC that controls the master shutter and the RF permit for operation. Information from all the beam line components is passed to the master PLC using peer cop communication through the Modbus Plus bus. The data is available in 16 bit registers in the master PLC. The master PLC processor is equipped with an Ethernet port for communication with the control system. Its registers contain information such as the open/closed status of the shutters, vacuum valves and safety shutters. The TSX momentum family products are used in this system.

2. The Booster-UV transport micro controls the operation of the transport power supplies and the UV trim micro controls the trim power supplies. For the ON/OFF control and status read-backs of these power supplies four non-intelligent I/O modules each with 16-bit inputs are used for reading the status of the power supplies. One non-intelligent I/O module with 16 inputs and 16 outputs is used to control the main and auxiliary contactors for the power supplies. The quantum PLC processor communicates with the IO modules via modbus plus ports. The quantum controller is coupled to an Ethernet module for communication with the VME micros.

3. Four micros control the RF systems for the UV and X-ray rings. The logic in the VME systems checks for various fault conditions before turning the amplifiers and the RF system on. A backup hardware logic board is also used in each system. The future plan is to eliminate the hardware and control the amplifier and RF system turn on/off sequences with PLCs. A PLC system for amplifier controls has been implemented and is ready for integration with the micro system. The micro will still implement DAC controls that require fast response while ramping. The PLC system for the amplifier controls has a TSX momentum processor and 2 I/O bases for analog read-backs and one I/O base for digital inputs/outputs. The PLC processor communicates with the I/O modules via I/O bus. The PLC program residing in the processor's flash memory implements all the amplifier turn on/off sequences, monitoring of fault status and digitization of analog signals. These data are available in 16-bit registers in the PLC memory. The Ethernet on the processor adapter communicates with the control system. One more PLC unit will be added to the system for RF system control.

6 CONCLUSIONS

The Modicon quantum processors and the TSX momentum family products and the Modbus/TCP protocol have facilitated seamless integration between PLCs and the micros. The software can be easily incorporated in any operational micros. The integrated systems have been stable and reliable.

7 ACKNOWLEDGEMENTS

The authors wish to thank R.Biscardi, E.Blum, M.Fulkerson, N.A.Towne J.Vaughn and E.Morello for their discussions and assistance

REFERENCES

[1] S. Ramamoorthy et al, "NSLS Control Monitor Upgrade", Proc IEEE, 1849,PAC 1993.

[2] A. Swales, "Open Modbus/TCP Specification", http://www.modicon.com/openmbus/eio/enetio.html, March 1999.

TUAP062
physics/0111101

SNS VACUUM INSTRUMENTATION AND CONTROL SYSTEM*

J. Y. Tang†, L. A. Smart and H. C. Hseuh, Brookhaven National Laboratory, Upton, NY, USA
P. S. Marroquin, and L. R. Dalesio, Los Alamos National Laboratory, Los Alamos, NM, USA
S. A. Lewis, C. A. Lionberger, Lawrence Berkeley National Laboratory, Berkeley, CA, USA
K. Kishiyama, Lawrence Livermore National Laboratory, Livermore, CA, USA
D. P. Gurd and M. Hechler, Oak Ridge National Laboratory, Oak Ridge, TN, USA
W. Schneider, Thomas Jefferson National Accelerator Facility, Newport News, VA, USA

Abstract

The Spallation Neutron Source (SNS) vacuum instrumentation and control systems are being designed at Lawrence Berkeley National Laboratory (LBNL), Brookhaven National Laboratory (BNL), Thomas Jefferson National Accelerator facility (TJNAF) and Los Alamos National Laboratory (LANL). Each participating lab is responsible for a different section of the machine: LBNL for the Front-End section, LANL for the warm LINAC section, TJNAF for the cold LINAC section and BNL for the Ring and transfer line sections. The vacuum instrumentation and control systems are scheduled to be installed and be in operation at Oak Ridge National Laboratory in 2004 or 2005. Although the requirements vary for different sections of the machine, a collaborative effort has been made to standardize vacuum instrumentation components and the global control system interfaces. This paper summarizes the design of each sub-section of vacuum instrumentation and control system and discusses SNS standards for Ion Pump and Gauge controllers, Programmable Logic Controller (PLC) interfaces, Ladder Logic programming and the SNS global control system interfaces.

1 SNS VACUUM SYSTEM

1.1 Vacuum System Requirements

The successful operation of SNS is dependent upon the reliable operation of the accelerator vacuum system in the high and ultra-high vacuum range under operating conditions. The required vacuum level varies over the different accelerator subsystems, i.e. the Front End, Warm Linac that includes the Drift Tube Linac (DTL) and Coupled Cavity Linac (CCL), Superconducting Linac (SCL), and the Ring that includes the High-energy Beam Transport (HEBT) line, the accumulator Ring and the Ring to Target Beam Transport (RTBT) line. The operational vacuum pressure level requirements have been analytically determined [1]. These levels are summarized as in Table 1.

Table 1: SNS Vacuum Level Requirements

Front End	1×10^{-4} to 4×10^{-7} Torr
DTL	2×10^{-7} Torr
CTL	5×10^{-8} Torr
SCL	$<1\times10^{-9}$ Torr
HEBT	5×10^{-8} Torr
Ring	1×10^{-8} Torr
RTBT	1×10^{-7} Torr

Associated with the vacuum pressure levels are the availability and reliability requirements of the vacuum subsystems and their components.

The subsystems' design basis is required to have a performance margin of 2, i.e. the failure of any single pumping element may degrade the subsystem pressure level, but will not raise it beyond the above listed levels where the accelerator operation is compromised or terminated.

1.2 Vacuum Components Standardization

A major consideration in the design of vacuum instrumentation and controls is standardization among subsystems with the exception of the Front End [2]. The specification and operation modes of the vacuum instrumentation could be quite different among these machine areas. Compromises are being made in specifications to allow for the use of identical or similar device. The aim of standardization is to ease commissioning, maintenance and upgrade efforts, in addition to simplifying the control software development effort, with the added benefit of lower cost due to larger quantity procurements.

* Work performed under the auspices of the U.S. Department of Energy
† jtang@bnl.gov

1.3 Vacuum System Instrumentation

Vacuum system instrumentation includes valves, gauges, ion pumps, turbomolecular pumps, and residual gas analyzers. All valves have +24 Vdc solenoids and have both open and closed limit switch-type indicators. Ion pumps will be used to maintain high vacuum in the machine. Remote serial communication will be used to turn on/off the pump high voltage and to read pump current, pressure, and voltage. Convection-enhanced Thermal Conductivity Gauges (TCGs) will be used to monitor the low vacuum levels within the accelerator, from atmosphere to 10-4 Torr. The high vacuum levels within the accelerator, 10-3 to 10-10 Torr, will be measured using Cold Cathode Gauges (CCGs). A multiple-gauge controller will control both CCGs and TCGs. Both ion pump and gauge controllers provide interlock inputs to the vacuum controls.

Partial pressure levels within the accelerator will be measured using residual gas analyzers (RGAs). The RGAs will be used to characterize the residual gases in the vacuum to aid in determining the gas source such as a water leak, an air leak or component outgassing.

Turbomolecular pump stations (TMPs) will be used to pump the beam line from atmospheric pressure to high vacuum, or to maintain vacuum in case of a leak. Remote operation of the TMP will be accomplished through remote control of analog inputs and discrete inputs and outputs.

2 THE VACUUM I&C SYSTEM

2.1 The SNS Front-End

The SNS Front-End consists of three subsystems: the Source/LEBT, RFQ and MEBT. The full system will entail about 500 signals for valves, pumps, and gauges. When initially designed in 1998, SNS-wide standards were not yet defined [2], therefore, low-risk choices with obvious upgrade paths were made: Allen-Bradley PLC/5 PLCs using the 1794 Flex-I/O interface family, with both IOC-to-PLC and PLC-to-interface are connected via Remote-I/O ("blue-hose") at 115kb/s. The PLC performs through its ladder logic the basic validation of signals and the first level of interlocking. In particular, it smoothes over some of the idiosyncrasies of the three basic types of high-vacuum pumps (turbo-molecular, ion, and cryogenic). It also enforces the SNS standard for valve operation using two normally open contacts for positive confirmation. More complex interlocking is performed in the IOC, such as turning off pumps and closing valves depending on the status of nearby gauges or related devices and requiring valid status before energizing these devices. The IOC logic (implemented with linked EPICS records) also enforces the inter-system constraints on the isolation valves. Special over-rides are provided to permit device testing when appropriate.

2.2 The Warm Linac and the Ring

The Warm Linac and the Ring vacuum systems share the same vacuum instrumentation and control architecture as in Figure 1.

2.2.1 ControlLogix PLC for Vacuum Interlocks

Allen-Bradley ControlLogix PLCs will be used to monitor gauge and pump setpoint outputs and control valves. The SNS Project has selected the ControlLogix PLC as the standard PLC [3]. Features of the ControlLogix system include being able to replace modules under power and read from all input modules from any processor within the ControlNet or EtherNet/IP network.

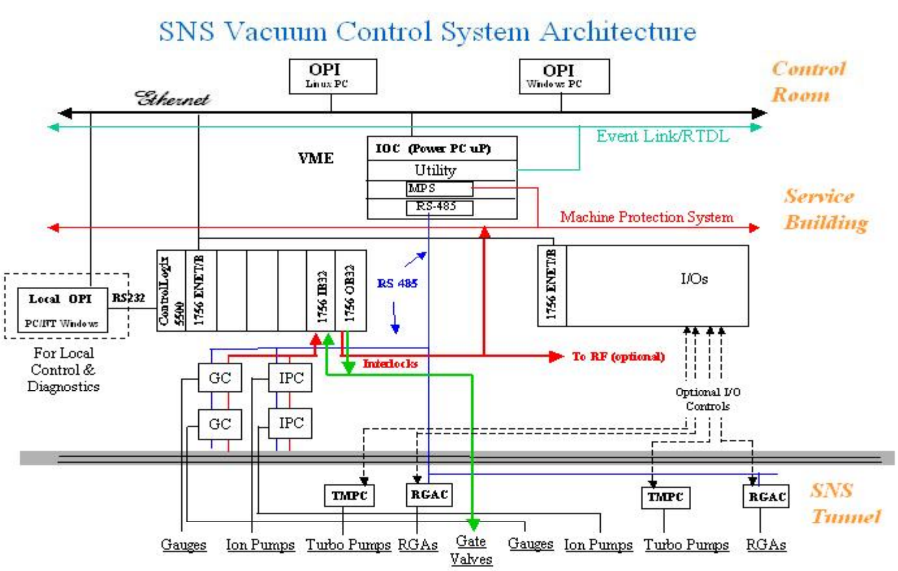


Figure 1. SNS Vacuum Control System Architecture.

The principal function of the PLC is to provide control of the sector gate valves that sectionalize the vacuum systems. The valve control logic will be fail-safe. A sector valve will close in case of a) vacuum conditions deteriorating to a specified limit, b) power loss, and c) operator input from the support building or remote terminal. The vacuum PLCs will also provide interlock outputs to SNS subsystems such as the RF System and machine protection systems, and receive interlock inputs from subsystems such as the Target System. All vacuum system interlocks will use 24 Vdc control power. A vote-to-close scheme will be implemented in the PLC ladder logic and it has been described in a separated publication [5].

The SNS warm linac consists of six DTL tanks and four CCL modules. Each DTL tank or CCL module has been designed to have its own vacuum control system. A total of ten PLCs will be used for the whole warm linac. This design choice was made to allow each tank and module to be vacuum leak checked, conditioned, operationally certified, and operated and maintained on an individual basis. This control system segregation is also consistent with the DTL and CCL water-cooling and resonance control systems and assembly and installation plans.

The control of the HEBT, Ring, and RTBT vacuum systems will be implemented using a network of four PLCs, based on the vacuum device distribution: one each for the HEBT and RTBT subsystems, and two for the Ring subsystem. Data exchange between PLCs is accomplished through a real-time, redundant-media ControlNet network [6]. The high-speed, deterministic ControlNet network is ideal for transmitting time-critical vacuum system information and providing real-time control.

Each PLC in each subsystem will also generate machine protection system (MPS) beam permit signals appropriate for the various machine operation modes. Details of the MPS to vacuum system interface are yet to be determined.

2.2.2 The Input/Output Controller

The Input/Output Controller (IOC) is a VME controller where the control system core resides. The primary function of the IOC is to provide the gateway between the global control system and vacuum instrumentation system. All information for machine operators will be provided via the IOCs. Five IOCs are planned, one each for DTL, CCL, HEBT, Ring, and RTBT vacuum systems. The IOC will reside in a VME chassis located in the same support building as the PLC. The PLC will communicate with the IOC through an EtherNet/IP [5]. The IOC will also interface directly with vacuum device controllers via RS-485 serial bus.

2.3 The Superconducting Linac

The vacuum system for the SNS superconducting linac consists of two differentially-pumped pressure regions to isolate cryo vacuum from the warm linac vacuum and the HEBT vacuum, eleven medium beta Cryo modules, twelve high beta Cryo modules, and vacuum for a drift region for future Cryo modules. While the acquisition of instrumentation for the subsystem is JLAB’s responsibility, LANL will be responsible for the control system. Eight IOCs are planned for the SCL.

3 SUMMARY

The SNS vacuum instrumentation and control system is a cooperative endeavor among six DOE Labs. A major effort on standardization among SNS vacuum subsystems has been made with the aim of easing commissioning, maintenance and upgrade work. The following areas of standardization have be turned out to be successful among the major sections of the machine:

- Instrumentation components
- Control system architecture
- Naming conventions

As of now, the Front-End vacuum subsystem has been completed while the rest is still under development, with hardware procurements in progress for Warm Linac and the Ring subsystems. The collaborative effort is still critical to the final success of SNS vacuum instrumentation and controls for commissioning in 2005.

REFERENCES

[1] H. C. Hseuh, et al, “Design of the SNS Accumulator Ring Vacuum Systems”, PAC’99, New York, p1345, (1999)

[2] S. A. Lewis, et al, “Status of the Control System for the Front-End of the Spallation Neutron Source”, ICALEPCS’01, San Jose, (2001)

[3] J. Y. Tang, et al, “A Distributed and Collaborative PLC lab for SNS”, ICALEPCS’99, Trieste, Italy, p370 (1999)

[4] http://www.controlnet.org/

[5] J. Y. Tang, et al, “Design and Development of the SNS Ring Vacuum Instrumentation and Control Systems”, PAC’01, Chicago, (2001)

[6] K. U. Kasemir, et al, “interfacing the ControlLogix PLC over EtherNet/IP”, ICALEPCS’01, San Jose, (2001)