THE GNU CONTROL SYSTEM AT CAMD: PART TWO

P. Jines and B. Craft
Center for Advanced Microstructures and Devices, Louisiana State University
6980 Jefferson Highway, Baton Rouge, LA 70806, USA

Abstract

Louisiana State University Center for Advanced Microstructures and Devices (CAMD) began a project to upgrade the storage ring and linac control systems in February 1997. At that time, control systems for the storage ring and linac were DEC VAX/VMS and VME/OS9 based systems, respectively. The objectives were to utilize inexpensive hardware, free software, and provide a flexible architecture where new projects could be easily integrated [1]. The storage ring control system has now been replaced with a PC/Linux based system, and the VAXes removed. A superconducting wiggler from the Budker Institute has been commissioned and integrated into this system [2]. CAMD is now focusing on expanding the control system to include the linac [3] and a second RF system, and reengineering subsystems to provide more reliable control. Automationdirect.com (formerly PLCDirect) PLC hardware has been chosen as the hardware platform for these upgrades, while PC/Linux will provide the man/machine interface. The design for these upgrades and their integration into the current control system will be presented.

1 CONTROL SYSTEM DESIGN

The key to the CAMD control system design is the unified API layer known as the "cht" (for "channel text" layer). Most of the control system programs, displays, and utilities are written in Tcl/Tk and rely upon this layer. The programs reference channels by a unique ASCII string, or channel name. The cht layer performs any necessary engineering unit conversion, and routes requests to the appropriate handlers for each different type of supported hardware. Using this approach, channels can be relocated among CAMAC modules or crates, or changed from CAMAC to PLC modules with no changes in the application code.

The cht layer relies upon the PostgreSQL database from the University of California for channel definitions. These contain the channel's hardware platform (i.e. CAMAC, GPIB, PLC, etc.), engineering unit conversion information, the type of channel (including analog vs. digital, input vs. output, bipolar vs. unipolar), channel limits, and the channel's location, such as CAMAC crate, module slot, module type, and subaddress.

The CAMD storage ring is “ramped” from 180MeV injection to 1.3 or 1.5GeV energy levels over a twenty five second interval. Approximately thirty channels are driven with up to two hundred and fifty synchronized setpoints per second. This requires greater throughput and level of synchronization than the PC/CAMAC crate controller combination is able to provide if channels are written individually from PC software. Hytec list processors are used to accommodate the speed and synchronization needs. Lists of CAMAC instructions (i.e. F’s, N’s, and A’s) and lists of setpoints are downloaded to each list processor and executed. The PC, using normal CAMAC communication, monitors the progress of the ramp. This system is used for the ring ramp, wiggler ramp, and transport line conditioning ramp.

In the VAX based control system, each type of ramp had its own special software. In the updated control system, the ramping software has also been genericized. Each ramp is controlled by a "scenario file". These scenario files describe the type of ramp, the setpoint file, the type of interpolation used, and ramp "load" and "run" programs. These two programs control the use of the CAMAC list processors. Different programs can be specified to allow for activities such as ramping the main magnets and RF systems separately for hardware testing, slew rate measurements, or ramp performance data collection.

By using generic ramping software and the unified API, the integration of the superconducting wiggler into the system was a very quick process. The wiggler was installed and tested using the vendor supplied control system. After a short cabling changeover, the wiggler was commissioned using the CAMD control system, and the new ring and wiggler ramp files were developed jointly between CAMD and the Budker Institute.

2 CONTROL SYSTEM UPGRADES

Several control system upgrades are currently in progress: 1) The CGR-MeV linac currently runs on a VME/OS9 platform. Several of the cards were custom made for this project, and have large silicon components which we have be unable to identify. The equipment protection interlocks are in software, and the system is loaded to the point that if the system is "ping"ed via TCP/IP, the watchdog timers expire and the linac shuts down. The software documentation only exists at the source code level, so determining program flow is difficult. Due to the fact that CGR-MeV is no longer in the injector linac business, CAMD is left with an unsupported linac with few spare parts and "unknown" software. The replacement of the linac control system is the highest upgrade priority. 2) New protein crystallography efforts at CAMD are driving the need for a second RF system. A major requirement is to have both RF systems use identical "off the shelf" components for ease of maintenance. 3) The kicker magnet controls are provided by Allen-Bradley PLCs. The PC scanner card required to communicate with the PLCs has two problems. First, the scanner card has acknowledged firmware problems which can cause the bus to "hang". A power cycle of the PC is required to continue normal operations. Secondly, Linux software for the card is not available. Development of the software required a non-disclosure agreement and custom driver development. 4) An upgrade is desired for the water control system. The current controllers have proven to be unreliable, and have presented interface difficulties.

The Automationdirect.com DL405 series PLCs with D4-450 CPU's were chosen for several reasons. 1) This PLC series can be used for ALL of the above upgrades. This reduces software development, maintenance, and spare parts costs by reducing the number of separate hardware platforms that are in use at the facility. 2) The hardware is inexpensive. Automationdirect.com hardware is approximately half the cost of a similarly equipped Allen-Bradley system. For a small number of channels, a PLC system provides greater capability at a smaller cost than a similarly equipped CAMAC system. As the number of channels increases, however, that savings decreases. 3) Sufficient documentation is available to communicate with the hardware using Linux, including the source code and a user supplied Linux port of their software development kit. The documentation and source are a "work in progress", but we had no difficulties in getting the system up and running. 4) The system can be used with either the PC or a dedicated PLC CPU as the controller. In the first case, the PLCs can be used simply as I/O. In the second case, local intelligence can be provided for equipment protection, ramping, or PID control. 5) The D4-450 CPU's can generate exceptions if the expected scan time is exceeded, providing a watchdog against failure of the equipment protection modules. 6) Interrupt inputs are available. If driven by a timing signal, these can be used for synchronization purposes, allowing the use of both the PLC and the CAMAC systems during ramping. 7) PID control is available locally in the CPU.

Using PLCs has some drawbacks. Analog voltage input and output modules are limited to 12-bit conversion, and are only available with up to four channels per module. Memory on each CPU is limited to approximately 28K words, so complicated ramps may have to be either split among several CPUs, or a multi-buffered ramp data approach may have to be used. The interface for programming the PLCs is Automationdirect.com's DirectSoft software. This is a graphical environment running on Windows, requiring a Windows machine for development. Also, any changes to the ladder logic of the PLC must take place within the graphical environment, so a "text based" code update is currently not possible.

3 PLC SOFTWARE DESIGN

The Automationdirect.com Ethernet communications module allows the PC to read and write the PLC's memory without special code in the PLC. This led to the idea of using a "dual-port" memory model, where machine status is communicated between the PLC and the man/machine interface on the PC thru the PLC's memory, requiring no special hand-shaking or communications code. By also storing equipment protection limits in the dual-port memory, this allows the same flexibility with equipment protection code: it can be located in the PC, the local PLC, or in a second PLC connected on the Ethernet network. For all upgrade projects, the software is therefore organized into three sections: the hardware input/output section, the man/machine interface, and equipment protection.

The hardware input/output module's responsibility is to keep the dual-port memory and the hardware status synchronized. Tables containing hardware setpoints are written to output channels, and the status of input channels are written to "readback" tables in the dual-port memory. Preliminary versions of this code are "hardwired": the number and types of hardware channels are fixed, as well as the corresponding addresses in the dual-port memory. As long as the increase in complexity does not impact the effectiveness of the equipment protection modules, the code will be upgraded to a more flexible architecture. Input/output configuration will be based on a combination of the PLC's input/output module probing and a table driven mapping between each channel and a dual-port memory location. This table would be provided by the PC, and correspond to the channel mapping as recorded in the PostgreSQL database. This would also provide the benefit of allowing all PLC CPUs in the facility to run the same code, regardless of I/O configuration or specific hardware task.

The equipment protection modules run in the PLC, and check the machine status against a table of hardware limits located in the dual-port memory. These tasks run in the PLC because the D4-450 CPU can start scans on a timed basis, and generate exceptions if scan times take longer than their allotted time. The PC, running a multitasking operating system, cannot guarantee as precise scheduling as the PLC. Failure to complete a scan on time will be reported by status bits in the dual-port memory. This status can also be reported by a "report on exception" feature of the Ethernet module, allowing instant notification to both the PC, and any other processors performing equipment protection tasks. As a added benefit of making the equipment protection table driven like the input/output configuration tasks, a generic software program can be used even in cases where equipment protection is not needed: an empty table implies no work for the task to perform.

Lastly, the PC is responsible for man/machine interface and configuration tasks. Monitoring machine status and updating hardware setpoints are performed by reading and writing values in the dual-port memory. Equipment protection ranges are adjusted by updating the corresponding tables. Initial input/output configuration information is handled by querying the PostgreSQL database for channel vs. dual-port memory location mapping, then updating the appropriate tables in the PLC's memory. Once the PLC's are configured by the PC, equipment protection modules are no longer dependent on the PC for operation. This frees the PC for operator control, automation, and logging functions.

4 PC/LINUX UPGRADES

One of the major benefits of the linac control system upgrade will be automated logging of linac machine parameters and limits. The control system currently keeps historical records of orbits, ramp files, beam current, power supply diagnostic tests, and channel default values. Specialized analysis programs are used to determine gradual drifts in component performance. Visual stacking of multiple orbit files helps identify failure of beam position monitor components. Magnet power supply data is collected on a per-injection basis, and is used to track power supply performance. Future programs will help identify long-term parameter drift, and provide indications as to possible failing components.

Increased automation of operator activities is also being achieved. Integration of the linac into the ring control system will allow full automation of machine turn on, conditioning, and shutdown procedures. In the present state, the ring RF frequency is derived from an oscillator in the linac. When the orbit correction software requires that changes be made to the RF frequency, this requires that the change be made on the linac control system. Due to the inability to communicate with the linac control system via network communication, this requires that the operator manually update the linac parameters. The linac upgrade will allow this to be a fully automated procedure.

5 CONCLUSION

For several upgrades, CAMD is reducing the investment in centralized CAMAC channels, and moving toward distributed, localized control using Automationdirect.com PLC systems. By using inexpensive programmable hardware, generic coding techniques, localized control, and distributed memory models, CAMD hopes to decrease development and maintenance costs for these upgrades. Integration of the superconducting wiggler and kicker racks has been straightforward, thanks to the control system's flexible architecture. Integration of the PLC based linac control system will remove the last major obstacle to automating most operator tasks.

ACKNOWLEDGEMENTS

The authors would like to express their appreciation of Linus Torvalds and the entire Linux community for the kernel we are using. We would like to thank the Free Software Foundation for the GNU compiler and libraries. We would like to thank The Regents of the University of California, Sun Microsystems, and John Ousterhout for Tcl/Tk. We would like to thank C. Schroeter for the Linux GPIB Package. This author (BC) would like to thank Alan Cox for numerous e-mails elucidating the organization of the Ethernet software within the Linux kernel. We would also like to thank the developers of PostgreSQL and XFree86.

REFERENCES

[1] B. Craft and P. Jines, *The GNU Control System at CAMD*, Proceedings of the 1997 ICALEPCS, 194-196 (1997)

[2] V. Borovikov, V. Djurba, M. Fedurin, G. Kulipanov, O. Lee, N. Mezentsev, V. Shkaruba, B. Craft, V. Saile, *Proposal of superconducting 7 Tesla wiggler for LSU-CAMD*, Nuclear Instruments and Methods in Physics Research A 405, 208-213 (1998).

[3] B. C. Craft, A. M. Findley, G. L. Findley, J. D. Scott, F. H. Watson, *Status Report on the CAMD Project*, Nuclear Instruments and Methods in Physics Research, B-40/41, 379-383 (1989)