Chapter 18.
The E-821 Slow Control System

Revised March 1994

18.1. Introduction

The $(g-2)$ muon storage ring is more like a small accelerator than an experiment in an extracted beam. There is a need for a more automatic presentation of device status, and a more central initiation of control signals, than is usually the case in a beam line. These requirements imply that the data acquisition systems for the hardware status, and the software for display, be fully integrated at the planning stages, rather than separately matched. An efficient way to take advantage of an integrated approach is to buy commercial plant-floor type monitors and their associated software packages. These customize easily to various industrial applications, and they seem to be applicable to the muon storage ring as well, even though it is not quite like a plant floor.

There are architectural considerations in an experiment that one would not confront in industrial applications. These differences make the $(g-2)$ control system not quite a plug and play toy, but careful consideration of how various ring subsystems interact should make the continuous operation of the ring fairly painless. The main idea in this architecture is that all hardware subsystems operate in a high-priority, local control loop, but talk to the slow control system when they can. The slow control system then logs status information or trouble reports. The local control loops also have priority over control signals sent down to them from the slow control system. This is to prevent the human operator of a system from helplessly losing control to the slow control system.

At the level of the actual hardware, the slow control system will communicate to modular industrial controllers. These controllers are microcomputers, generally configured to operate scanning analog to digital converters or standard industrial signal conditioners. They exist as self-sufficient processors, and have the ability to run the local control functions easily. The ones that E-821 will use will be Allen-Bradley PLC's (Programmable Local Controller's), or their equivalents. They are nearly ubiquitous in plant floor environments, and they have sufficient communications ability to fulfill the reporting requirements for the slow control system. The newest versions also allow ethernet connections, which simplifies their integration into an E-821 network.

The whole control system will use an ethernet network to communicate among its parts, and to communicate at a higher level to the data acquisition host. It will also have a hidden network strictly for automatic control purposes and free of any monitoring responsibilities. The experiment will have its own subnet on the internet, but will be accessible in only a limited fashion to the rest of the internet. This will prevent an anonymous user in a far-off place from running a portion of the experiment in an unforeseen way. However, the ability quickly to communicate news items or experimental data is quite feasible and appropriate.

18.2. The Local Controller Hardware

The controllers for the muon $(g-2)$ experiment will be Allen-Bradley PLC-5 or AEG Modicon controllers. These controllers have a small complement of memory with a battery backup, some communications ports, the ability to scan several hundred to several thousand input or output circuits, and options for EEPROM program backup. Basically, they are scanners that read to or write from a bank of circuits. Separate modules determine the identity of these circuits, and the types of signals they process. The controller simply reads channels and stores them in its memory. It then provides the channel information to a display, which can be just a screen with a display of status indicators, or a highly integrated front end of a control system.

The communications channels that the controllers use come in three types, two quite useful and one less so. There is the proprietary Data Highway, which connects all the PLC's, and which has a reasonably high capacity. There is an ethernet link, which exists independently of the Data Highway, and has about twenty times its transmission capacity. And finally, there is an RS-232 port, useful as a debugging tool, on each PLC.

The controller has all the characteristics of a small, dedicated computer. There is some data handling ability, interrupt capability, status flags, and the ability to run multiple control programs. Their use in E-821 will be to run local control loops for the major subsystems, and to pass news reports back up to the slow control host computer.

The controllers themselves do not have the ability to process raw input signals from various devices. Specialized input modules will handle these tasks. As an example, the controls for the liquid helium refrigerator require temperature readings from junction sensors. A specialized module linearizes and buffers these junction signals before passing them on to the processor, which likes to see only well-conditioned, standard industrial inputs. Other electronic front ends perform similar conditioning in their respective environments.

This separated function design of the electronics also minimizes confusion in the field wiring to the processors. All wiring from the front end modules to the processors is on terminal strips, and is identical with respect to wire type and termination. The specialized input wiring we will segregate by function onto the appropriate electronics. This input bundling is easier to maintain and repair than just running wires in sequential channel order, heedless of their functions.

18.3. The Network

Devices and computers in the muon $(g-2)$ experiment will communicate with each other via an ethernet network. The PLC-5 controllers have an internal ethernet interface, requiring, however, an external transceiver. The data acquisition (DAQ) host computer and slow control computer, if they are not the same, will both have ethernet capabilities. Allen-Bradley software, running on the slow control computer, will handle the communications between controller and the ethernet. This network should handle easily the data transmissions from ring subsystems and file transfers from the DAQ host.

In addition to the ethernet connections, the PLC's will communicate with each other via the proprietary Allen-Bradley Data Highway (actually, Data Highway+, or DH+), a custom network for their PLC interconnections, but well-supported by other PLC vendors. This second "shadow network" increases the slow control reliability by isolating the important control loops at the major subsystem level from the pure news-gathering functions of the ethernet. The Data Highway provides a way for automatic controls to function separately from the slow control system, though one could also activate devices through slow control, with the approprate priority and privileges. Since the two networks share nothing, failure of the slow control system or the ethernet does not compromise the operation of the ring subsystems.

The Data Highway transmission rate is 57.6 kbaud, in contrast to the much faster ethernet rate of 10 Mbits/sec. However, the Data Highway carries much less in the way of actual data than the ethernet. The Data Highway interconnection is responsible for carrying the control signals, global flags, and small blocks of data among the PLC's. The detailed description of these transmissions among the PLC's is actually imbedded in the design of the subsystem control loops, which will be described later. This "hidden" network remains invisible to the ethernet network, so the ethernet must also carry some of the same information. This redundancy is fortuitous, increasing the reliability of the transmissions at no cost, since E-821 requires the ethernet and Allen-Bradley supplies the Data Highway as standard equipment.

Figure 18.4.1: The E-821 Slow Control ArchitectureThe host computer and DAQ host are one computer in this figure. The host communicates with the PLC's on the ethernet, using Interchange software. The Data Highway is strictly for the use of the automatic controls.

18.4. The General Architecture

The basic idea of a slow control system is to provide a monitor on the $(g-2)$ ring function. It should allow the free transfer of data from ring subsystems, unencumbering the extant automatic controls. These automatic controls and their self-linkages are not properly in the realm of the slow control system. However, there is a connection between automatic and slow control systems. It's probably best to describe the entire ring heirarchically, from the controller, at the base, to the slow control monitor, at the apex.

Fig. 18.4.1 illustrates the basic architecture. The PLC controllers share automatic control among themselve on the Data Highway, and they pass news reports up to the slow control host on the ethernet. Note that the only common point between the two networks is at the PLC. From there on, the networks separate permanently. The ethernet network includes the slow control computer, here shown as part of the DAQ host. This may not be the case in the actual implementation, since there may be a temporal advantage to obtaining a separate slow control host. If this were the case, the DAQ host would simply be another ethernet node.

The host computer runs the proprietary Allen-Bradley Interchange software, which allows the host to communicate directly to the PLC's over the ethernet. At this point, the slow control system has gotten access to data and/or news reports from the controllers. It is a collection of "unedited" data packages from the controllers. It will be the function of some higher level software to integrate and present these reports in graphical form. The host computer monitor will display a picture or schematic of some subsystem, and variables or parameters that are changing will appear in graphical form. For example, the helium level in the cryostat would appear on the screen, if that screen were selected. The cryostat, filled with helium, would appear as part of a cartoon illustrating the refrigeration system. The software that drives these displays is a commercial product whose only purpose is to generate graphics such as these. There are several commercial options for the slow control graphics software, as well as non-commercial options such as Los Alamos' EPICS. These are all under evaluation.

The central tenet of the slow control system is that it monitor and control the E-821 subsystems without disrupting the local control or feedback of that subsystem. To achieve this non-interference, it is necessary to assign priorities to any operations which may affect the running of a particular subsystem. Fig. 18.4.2 illustrates, in a schematic way, the loops that may exist in the control structure. A local process, for instance, a power supply, has inherent feedback control, and may have setpoints and software controls, as well. The loop labelled "Control" in Fig. 18.4.2 corresponds to the full set of these local device controls. They have the highest priority, being interruptible only under conditions set forth by the designer of the local loop.

Next higher in priority is the pathway labelled "Triggers," which represents a way that the slow control operator (*i.e.* an E-821 "Control Room" operator) could trigger a switch to control a device, or process. As a garish example, there could exist a "Quench" button, which would activate an energy dump from the magnet (at half-field, of course!). Since these priorities are lower than the local loop priorities, the triggers are easy to disable intentionally at the hardware level.

At the lowest priority is the passing of reports up to the operator console, which displays them. The requirement here is for an update rate of perhaps once per second for preformatted reports. This should be slow enough that the PLC's have no problem keeping up with requests from the slow control system, even though they are executing their own set of instructions in the local loops. The slow control software handles the initiation of requests and processes the replies from the PLC's. No request for news ever slows down the automatic control system, since these requests are both of the lowest priority, and

Figure 18.4.2: The Slow Control LogicControl loops indicate the automatic controls. They have top priority. Next in priority are the triggers, which can be locally disabled, which the enable line indicates. Last in priority are the news transmissions. The connection to DAQ is for completeness only.

are travelling on the ethernet, not the Data Highway. The only mechanism for slowing any control functions will be the processing of requests by the PLC's, or the operation of triggers. If the news report from the PLC exists in a predetermined format, of constant length, the delay caused by its transmission becomes a bookkeeping problem. This just means that it becomes part of the local loop, simply another task for the PLC.

Chapter 19.
Offline Analysis

Revised April 1994

19.1. Introduction

The goal of the offline group is to provide the g-2 collaboration with a set of tools for doing analysis, including easy access to the data and a common framework for developing managing and distributing analysis code. The offline code will not be as complicated as that required for the typical high energy physics experiment because of the limited number of physics goals and the relative simplicity of the cuts needed to form each dataset. g-2 is a minimum bias experiment in the truest sense. In order to conserve resources, however, it is imperative that as much of the offline package as possible be obtained from somewhere else, in particular from CERN. While not perfect, CERN code is coherent, well supported and runs on all the platforms needed by g-2: HP-UX, SGI IRIS, SUN SPARC DECstations and VAX/VMS. Furthermore, CERN software packages exist for nearly all our major software requirements - most of the offline work should involve the integration of this contributed code.

Listed below are the principal tasks of the offline group.

1 Set coding standards
2 Define data bank structure
3 Define analysis framework
4 Establish code management
5 Provide code distribution
6 Provide standard user interface
7 Set up database (geometric and calibration)
8 Provide histogramming and display capabillity: PAW/HBOOK/HPLOT/HIGZ etc.

The first item includes choosing the languages we will use and the coding standards we hope to enforce, selecting a databank manager for input and output, and setting up a code verification strategy. The analysis framework is a general means for integrating user code. With input and output standardized in terms of well-defined data banks (item 2) the user instructs the analysis manager to execute a set of routines on each data unit (presumably