

PLC SUPPORT SOFTWARE AT JEFFERSON LAB*

P. Chevtsov, S. Higgins, S. Schaffner, D. Seidman
Jefferson Lab, Newport News, VA 23606, USA

Abstract

Several Automation Direct (DirectNet) Programmable Logic Controllers (PLCs) have been integrated into the accelerator control system at Jefferson Lab. The integration is based on new software that consists of three main parts: a PLC driver with a state machine control block, a device support module, and a common serial driver. The components of new software and experience gained with the use of this software for beam dump systems at Jefferson Lab are presented.

1 INTRODUCTION

PLCs were developed in the late 1960's to eliminate the large cost involved in changing complicated relay based machine control systems. These systems were inflexible. Their major rewiring or complete replacement was necessary every time when the production requirements changed and control sequences had to be modified. It became very expensive when changes were frequent. The idea to replace relay systems with flexible microprocessor based PLCs was a revolutionary step in the industrial control business.

PLCs have been used for many years at Jefferson Lab. They are the main control elements of several subsystems which require relatively slow data monitoring: the radiation control, personal safety, fire and smoke alarm. All these subsystems are very cost effective and reliable. Adding new equipment to control or changing operating conditions to implement new control features can easily be done by the skilled technician staff, without any help from the Controls Software Group.

Recent evolution towards fast processors and a wide variety of compatible I/O modules make the PLCs very attractive for new control applications at Jefferson Lab. The detailed analysis of the PLC market based on cost, performance, data network solutions, and availability of different I/O modules, has allowed us to select Direct Logic products from Automation Direct [1] as the basic PLC line for accelerator control developments. The Direct Logic family of products supports the DirectNet data network that is based on point-to-point (RS-232) and multi-drop (RS-422) serial interface types. DirectNet uses the DirectNet communication protocol and operates asynchronously at speeds up to 38.4 K baud. Therefore, RS-232 is the best choice for applications moving large amounts of data or requiring fast update times. The network is controlled by a master station that issues network commands to individual slave stations.

2 OLD ("TRADITIONAL") PLC CONTROL AT JEFFERSON LAB

The accelerator control system at Jefferson Lab is based on the EPICS toolkit [2]. The EPICS standard model divides control into three distinct layers: the driver support that interfaces directly with hardware and is independent of EPICS, the record support that handles the communications with Channel Access (CA), the EPICS network protocol, and has no knowledge of the control device, and the device support that handles the interface between the driver and record support layers.

Until recently, PLCs have been connected to the serial ports on the Input-Output Controller (IOC) transition modules. Since it is relatively easy to access these serial ports using the standard terminal driver supplied with the real-time operating system running on IOCs, serial ports have been treated as file I/O and data communication has been provided by custom EPICS subroutine records and State Notation Language (SNL) sequencers with the use of simple read/write commands [3]. This way to handle device control bypasses the standard EPICS model. It moves all the processing on the top layer and makes it difficult to identify and maintain PLC control applications since hardware identification in EPICS is handled through the device layer.

To fit the EPICS standard model and the application development environment at Jefferson Lab, new PLC control software has been created.

3 NEW PLC SUPPORT SOFTWARE

Our serial software development tool and some ideas of A. Johnson's device support for DirectNet PLCs via MPF [4] became the basis of a new PLC control software library at Jefferson Lab. The software acts as a DirectNet master through any RS-232 port and consists of three main parts: a PLC driver with a state machine control block, a device support module, and a common serial driver (Fig. 1).

3.1 PLC Device Support Module

The device support module is initialized during the IOC startup procedure. Standard INP and OUT fields of EPICS records contain the name of the PLC to communicate with. This name must be unique for each PLC connected to the IOC. It is used by software to spawn PLC Data Client tasks (one task per PLC) which deal with the control device data requests.

All input records share a common data cache to reduce

*Supported by DOE Contract #DE-AC05-84ER40150.

the amount of DirectNet traffic that could be generated by multiple reads of the same or nearby PLC memory locations. The data cache keeps the data together with its time stamp and is handled by a Data Cache Manager (DCM). When an input record is processed, the DCM examines the cached data for the relevant location. It compares the age of the data with the scan period of the record. If this data is new (i.e. was received less than half a scan period ago) then the cached data is returned immediately finishing the record processing. Otherwise the request is sent to the PLC for new data through the corresponding PLC Data Client.

Output records have access to only the PLC general data storage area. This restriction ensures that the memory words written to the PLC output space by the ladder logic cannot be changed by output records. The IOC maintains its own buffer of values in each output location. This allows several records to point to different parts within the same memory location.

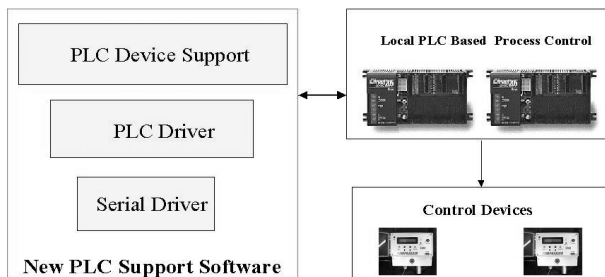


Figure 1: New PLC support layout.

3.2 Common Serial Driver

Serial driver software is activated with the use of two library calls. The first call initializes the serial communication hardware. The second call sets the basic communication parameters for each serial port provided by the underlying hardware: the baud rate, number of stop bits, parity, etc. Each serial port is served by a separate control task. The task reads and writes the data into and out of the serial port and handles the hardware operation timeouts. It also has an internal data parsing procedure and provides the interface to it for any application that needs to access serial port data streams.

3.3 PLC Driver

The PLC driver is the main integrating part of the PLC control software. It communicates with the serial driver and device support module, handles the DirectNet protocol, and provides the information on the PLC status. For each used PLC the driver is activated by only one library call. The arguments of this call are the serial port hardware parameters and the name of the PLC connected to this port. They are used to spawn a PLC driver control task (one per PLC) with the references to the proper serial driver and PLC device support module.

The PLC driver control task has three main components: a PLC Data Server, a State Machine Control Block, and a Data Communication Diagnostic Block.

The PLC Data Server receives data requests from the corresponding PLC Data Client described above and directs them to the appropriate serial port control task. When the data from the PLC is received, the Data Server sends them to the Data Cache Manager, a component of the PLC device support module mentioned above. DCM puts this data into the data cache and finishes the EPICS record processing procedure.

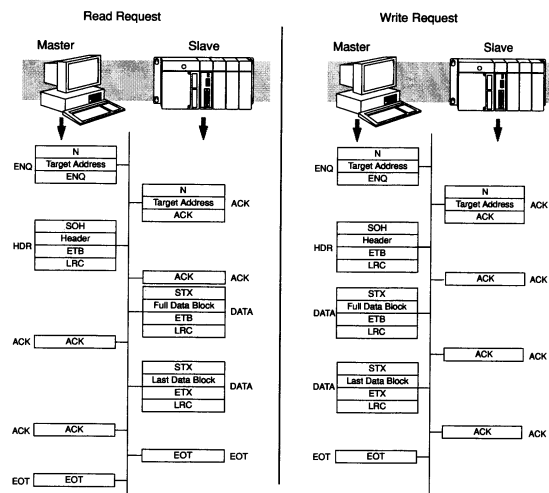


Figure 2: DirectNet communication protocol details.

The DirectNet protocol defines a set of rules for the dialogs between a host computer and a PLC. It requires providing a lot of service and diagnostic information such as the end of transmission (EOT) flag and data check sum (LRC – the longitudinal redundancy check) as well as data receipt acknowledges (ACK) from both sides for each transferred data block (see Fig. 2). As a result, the scenario of each dialog depends on the requested data type.

The State Machine Control Block completely automates the IOC part of the dialog. It uses the interface to the data parsing procedure of the serial driver described in the previous section to take care of all data receipt acknowledges immediately after they are provided or requested by the PLC. This allows the software to keep the highest data exchange rate for all DirectNet network connections.

The Data Communication Diagnostic Block collects various status information about PLCs: the total number of data requests, the number of hardware and software failures, data communication time statistics, etc. All this information is provided for the PLC device support that makes it available for the whole control system.

4 PLC BASED BEAM DUMP CONTROL

The beam dump systems are very important components of the experimental end stations at Jefferson Lab. They provide controlled absorption of the accelerator beam power. In order to safely dissipate the high power continuous electron beam (up to 1 MW, 6 GeV, 200 μ A) the dumps are equipped with a number of sensors for measuring such critical control parameters as the hydrogen production, cooling water flow, temperature, pressure, etc.

In a new system design, each beam dump is completely handled by one DirectNet PLC. All control functions have been developed and implemented with the use of the Ladder Logic programming software package DirectSOFT 32.

Two new PLC based beam dump control systems (for experimental Halls A and C) have successfully been working at Jefferson Lab since April 2002. The software has been running very smoothly without any failures. Together with such powerful EPICS tools as the Alarm Handler and Data Archiver it has been providing an excellent control of beam dump systems.

The general structure of dumps as well as critical process parameters and their statuses are provided on the main dump control screens. They have been designed by the Controls Software Group together with instrumentation engineers. The information on the screens (Fig. 3) is very easy to understand and extremely important for the beam dump system diagnostics.

5 CONCLUSIONS

New PLC support software significantly simplifies the process of the creation of PLC control applications. First, it does not require software coding. Secondly, it is completely activated with the use of only a very limited number of library calls. All necessary activation calls for each control computer can be put into one configuration file that is downloaded into this computer from a standard directory. This makes it much easier to support existing and add new hardware components throughout the whole control system.

6 ACKNOWLEDGEMENTS

The authors are grateful to A. Johnson for very fruitful discussions and assistance in this software project.

7 REFERENCES

- [1] "DirectNet Manual", DA-DNET-M, Automation Direct, 1998.
- [2] L.R.Dalesio et al., "The Experimental Physics and Industrial Control System Architecture: past, present and future," NIM, A 352 (1994).
- [3] P. Chevtsov et al., "Information-Control Software for Handling Serial Devices in an EPICS Environment", ICALEPCS-2001, San Jose, CA, USA.
- [4] A. Johnson, "Device Support for DirectNet via MPF", www.aps.anl.gov/asd/people/anj/directNet/dnMpf.html

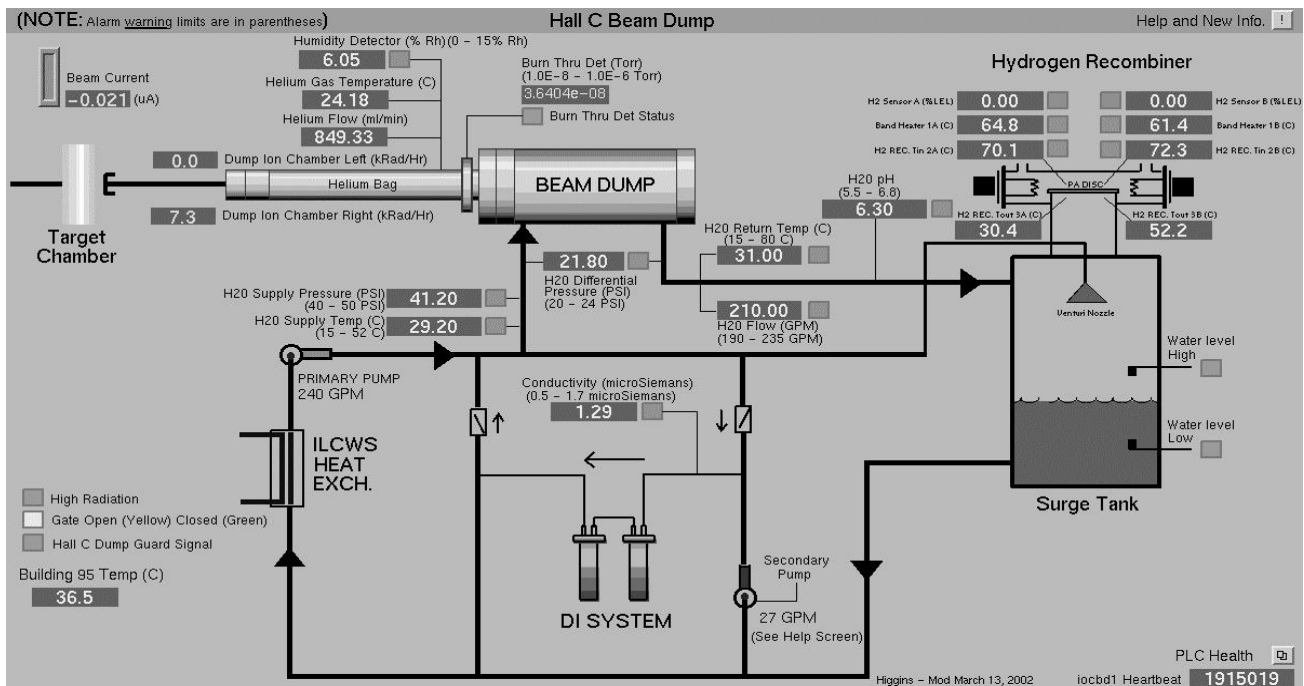


Figure 3: Main control screen for the Hall C beam dump system at Jefferson Lab.