Fermi National Accelerator Laboratory

TM-1458
2320.000

List Processing Software for the LeCroy 1821 Segment Manager Interface*

T. Dorries, C. Moore, and R. Pordes
Data Acquisition Software Group
Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Illinois 60510

May 1987

*Presented at the Fifth IEEE Conference on Real-Time Computer Applications in Nuclear, Particle, and Plasma Physics, San Francisco, California, May 12-14, 1987.

Operated by Universities Research Association Inc. under contract with the United States Department of Energy

LIST PROCESSING SOFTWARE FOR THE LECROY 1821 SEGMENT MANAGER INTERFACE

Terry Dorries, Carmenita Moore, Ruth Pordes
Data Acquisition Software Group*
Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Il 60510

Setting the Scene

The LeCroy 1821 Segment Manager is a FASTBUS host interface that is used in experiments at Fermilab at two levels of the event data readout.[1] At the lowest level the 1821s are employed in front-end crates to read TDC and ADC modules. In some cases 1821s operating at this level read the data directly into a host computer; in other cases they store the data in intermediate buffer memories such as the LeCroy 1892 multiple record buffer memory.

At the second level the 1821s are used in the (single) host FASTBUS crate at the apex of the data readout. Here the 1821s are used to read data into the data acquisition computer from the buffer memories in which many hundreds of previously gathered events have been stored.

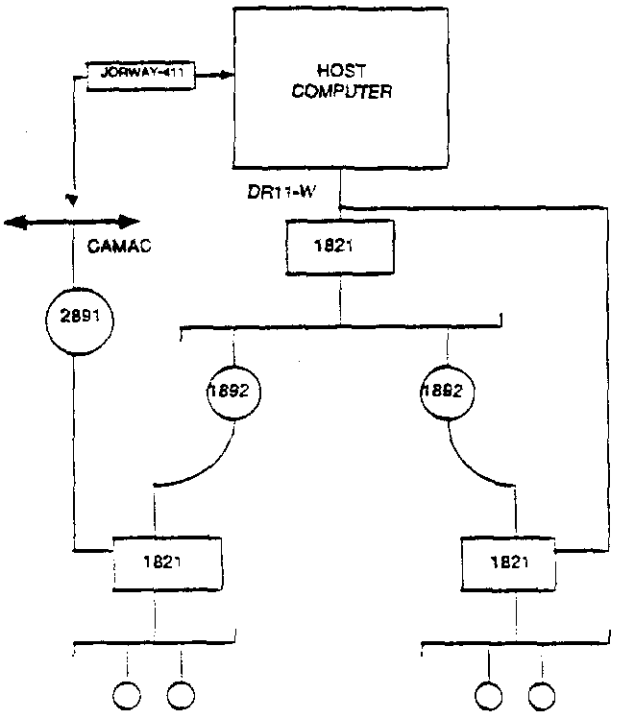

Figure 1: Topology of Experiments Using 1821

The Data Acquisition Software Group in the Computing Department at Fermilab provides support software for the 1821 to run on VAX/VMS, PDP-11/RSX-11M and PDP-11/RT-11 systems. The software has been developed in two phases.

The first phase provides a package of low level access routines which allow the user to immediately execute any one of many 1821 specific functions. The second phase, which is only available on RSX-11M and VMS systems, provides for the definition and execution of lists of 1821 operations. By executing lists of 1821 operations many 1821 functions can be performed in a single I/O request to the device driver thus significantly reducing the software overheads associated with the use of the 1821. This paper will concentrate on the list processor implementations. Since the low level routine package is used by general user level programs, it will be briefly discussed.

Connection to the 1821s

A LeCroy 1821 Segment Manager can be controlled by a host computer through one of two connection paths. The LeCroy 2891 Interface Module is used to interface the 1821 to CAMAC allowing the host computer access to the 1821 through CAMAC. Alternately the 1821 can be connected to a UNIBUS or QBUS through a DR11-W DMA interface.[2] In the latter case, if several 1821s are chained together, at least one 1821 in the chain must be connected through the auxiliary port to the DR11-W using a LeCroy 1821/DEC interface card. Other 1821's may be chained to this primary, either through connections to their front panel or, with a slight modification of the standard 1821/DEC card, through a private 1821/DEC connected to their auxiliary port.

Note that for a host computer with a UNIBUS connection, the 1821 may be connected using a DEC DR11-W. When the connection is to a QBUS, the DEC DRV11-W has been found not to work. (This is because the 1821/DEC relies on the GO strobe to initiate its operations, and the DEC DRV11-W does not provide this strobe to the user device.) The MDB Systems MLSI-DR11-W interface, however, does work.[3]

Low Level 1821 Access Routines and Support Programs

The low level access routines are a Fortran routine package that provide a user interface to the 1821. The package contains routines for reading and writing 1821 host I/O registers, downloading and executing 1821 microcode programs, downloading pedestals, and reading out the 1821's internal memory.

Two diagnostic tools, one which tests the internal integrity and functions of an 1821, and one which provides an interpretive environment for a user to specify 1821 operations to be performed, have been developed based on the low level routines.[4,5]

Overview of the List Processors

For a user's program to access the 1821 the host's operating system must place an I/O request to the 1821 device driver. The time required for the system to issue such a request is on the order of 2 to 4 milliseconds. Several 1821 functions must be done

*Operated by the URA, Inc., under contract with the U.S. DOE.

for the 1821 to perform any useful function. For example, using the 1821 microcode provided by LeCroy, an operation to read a FASTBUS data word (if it is to follow the FASTBUS Specification of an operation) will require between 4 and 6 functions performed on or by the 1821. Clearly the software overhead imposed by accessing the 1821 on a per function basis is too restrictive in a real time data acquisition environment.

In order to minimize the number of I/O requests issued by the system, list processing drivers have been developed. Such drivers exist for the RSX-11M and VMS systems for 1821s connected through a DR11-W to the host. These drivers allow the user to perform a list of 1821 and FASTBUS operations through a single I/O request. The drivers are a significant extension to the original drivers provided by LeCroy (VMS) and Ohio State University (RSX).[6,7]

In addition to the device drivers, the low level Fortran access routines have been extended to allow building, executing, saving and interrogating the lists and the results of their execution.

List Structure

The list structure developed consists of fixed length 8 byte list elements. Each list element specifies an 1821 control operation, DMA readout of the 1821 data memory, microcode execution, or a control or arithmetic function. The list structure is designed to ease the later implementation of the standard routines for FASTBUS[8] for the 1821. The list element structure is compatible with the so called "F-Code" format, defined by CERN for FASTBUS lists.[9] The available list element opcodes provide enough flexibility and functionality that a whole data acquisition read-out can be specified by a single, or at the most two, lists.

The list structure allows for:

- o definition of control parameters to vary the options and parameters associated with execution of each list element,
- o read/write of individual 1821 I/O registers,
- o execution of 1821 microcode routines with and without data from the host computer,
- o readout of the 1821 data memory within the list,
- o 3 internal variables for manipulation within the list,
- o arithmetic and logical functions to be performed on the data read or written, the status returned by the 1821, or the internal variables,
- o immediate or conditional branching within a list,
- o execution of DO loops within the list,
- o 16 or 32-bit mode data manipulation,
- o separate read and write data buffers,
- o return of status information from the execution of the list,
- o return of status from the execution of each operation within a list,
- o inclusion of an 'Event Header' in the data buffer (for the VMS implementation only).

Timings

The following timings have been measured for the two list processing drivers. A VAX/780 computer was used for the VMS timing tests and the RSX tests were run on a PDP-11/45. The null list times represent the time required for the user's process to invoke the device driver, the driver software to do the setup necessary for list execution, and for the driver to return control to the user's process; that is, the one time overhead per list execution. The overhead per list element varies significantly with the actual operation being performed. The total overhead for a particular list is the sum of the per list element overheads for each list element within the list and the null list overhead.

	VMS	RSX
Null list:	6.7 Msec	4.1 Msec
Overhead per list element:	160-360 usec	400-850 usec
Overhead for single operation using non-list processing device driver:	4 Msec	2 Msec

User Subroutine Library

The Fortran subroutine library provided allows a user program to dynamically build and execute lists. Subroutines are available to construct list elements, execute, save or restore a list, change options and parameters associated with list execution, and interrogate the status and data returned from execution of the list.[10]

List Generation and Execution Tools

Programs have been developed for both RSX-11M and VMS systems to provide interactive features for performing list operations through the 1821. These programs allow the user to interactively define, edit and execute the 1821 lists. The programs can either create lists dynamically or read in "source code" list files (ASCII text files containing list element definitions). Once a list has been created or read in from a disk file it is compiled into an executable form. The compiled lists can then be executed and/or saved on disk for later execution. If the user chooses to execute the list then the programs can be used to interrogate the results - both data and status information.[11,12]

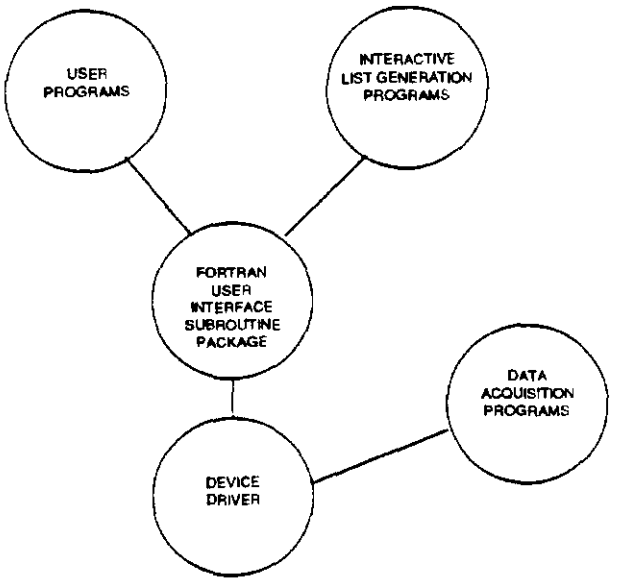

Figure 2: Overview of the 1821 Software

Data Acquisition Applications

Execution of 1821 lists on receipt of an event interrupt has been incorporated into the Fermilab standard RSX-11M data acquisition program[13] and is in use at two experiments. Data acquisition through the VMS 1821 list processing driver has been incorporated into the VAXONLINE Event Builder,[14] and will be used in a fixed target experiment during the upcoming run. It is interesting to note that this last experiment uses a list of some 100 operations to provide sophisticated control of which modules should be read out, and what 1821 microcode routines are to be executed.

In both data acquisition systems, the 1821 lists are constructed 'offline' using the list generation tools. They may be executed for test purposes using these tools, and then saved on disk for later execution by the data acquisition programs themselves.

Summary

Many fixed target experiments at Fermilab now include some FASTBUS electronics in their data readout. The software reported in this paper, provides general support for the LeCroy 1821 interface. The list processing device drivers allow FASTBUS data to be read out efficiently into the Fermilab Computing Department supported data acquisition systems.

Acknowledgments

As usual many people have contributed to the work reported in this paper. All members of the Data Acquisition Software group contribute to any project being developed in its midst. For the projects reported in this paper, in particular, many experimenters at Fermilab have contributed to the detailed understanding of the LeCroy hardware, and the testing of software for the 1821.

References

1. LeCroy Corporation, 700 So. Main Street, New York 10977.
2. DR11-W Direct Memory Interface Module, Digital Equipment Corporation.
3. MDB Systems, Inc., Orange, CA 92665.
4. T1821, Lecroy-1821 Diagnostic Program, Fermilab, PBN-040.
5. LeCroy 1821 SM/I Diagnostic Tool (LDT), Fermilab, FBN-041.
6. NYCB Computing, Inc., Lecroy 1821 Device Driver.
7. John W. Heimaster, Ohio State University.
8. Proposed IEEE Standard Routines for FASTBUS, FASTBUS Software Working Group, February 1987, Fermilab FBN-051.
9. CERN's Approach to FASTBUS Software, E.M. Rimmer, IEEE Transactions on Nuclear Science, 3968, Vol. NS-30, October 1983.
10. Low Level Routines for the Lecroy 1821, Fermilab FBN-039.
11. FLIST, An Interactive Program for Generating Lecroy 1821 Lists for RSX-11M, Fermilab FBN-0048.
12. Interactive List Generation Program FLMENU, Fermilab FBN-0047.
13. A High Speed CAMAC Data Acquisition System for PDP-11, D.M. Berg, et al., IEEE Transactions on Nuclear Science, 1368, Vol. NS-32.
14. See companion paper to this conference, The VAXONLINE Software System at Fermilab, V. White, et al.