

Lawrence Berkeley National Laboratory

Recent Work

Title

From Chaos to Order: The Microstar Data Acquisition and Analysis System

Permalink

<https://escholarship.org/uc/item/1j6222f3>

Author

Rathbun, W.

Publication Date

1991-03-01



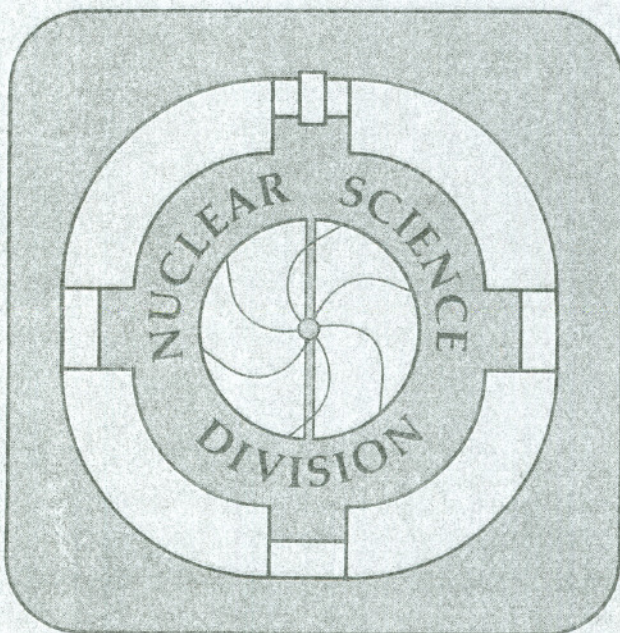
Lawrence Berkeley Laboratory

UNIVERSITY OF CALIFORNIA

From Chaos to Order: The Microstar Data Acquisition and Analysis System

W. Rathbun

March 1991



6

DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. Neither the United States Government nor any agency thereof, nor The Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or The Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or The Regents of the University of California and shall not be used for advertising or product endorsement purposes.

This report has been reproduced directly
from the best available copy.

Available to DOE and DOE Contractors
from the Office of Scientific and Technical Information
P.O. Box 62, Oak Ridge, TN 37831
Prices available from (615) 576-8401, FTS 626-8401

Available to the public from the
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road, Springfield, VA 22161

Lawrence Berkeley Laboratory is an equal opportunity employer.

From Chaos to Order

The MicroStar Data Acquisition and Analysis System

William Rathbun,
Staff Scientist, Lawrence Berkeley Laboratory
Berkeley, California

March, 1991

This work was supported by the Director, Office of Energy Research, Division of Nuclear Physics, Office of High Energy and Nuclear Physics of the U.S. Department of Energy under Contract DE-AC03-76SF00098.

VAX, MicroVAX, VAXstation, VMS, RT-11, PDP, PDP-11, RT-11, J11, Qbus and DEC are trademarks of Digital Equipment Corporation, Maynard, Mass.

Starburst is a trademark of Creative Electronic Systems (CES), Geneva, Switzerland.

CC-LSI and CC-Qbus are products of DSP Technology Corporation, Fremont, California.

DCQ-1000 is a product of The Logical Company, Cottage Grove, Oregon

Document Revision March 8, 1991

I.	Introduction and Historical Overview	1
II.	Learning the System	3
III.	Data Acquisition: Commands and Examples	5
IV.	Data Analysis: Options and Commands	11
V.	Interactive Display and Analysis	27
	1-D Display and Analysis	27
	2-D Display	37
VI.	Using <i>Starburst</i>	41
	Front-end Programming and Preprocessing	42
	The Acquisition Hardware Setup	46
VII.	Monitoring and Control	47
	High-voltage Control	47
	Scaler Monitoring	47
Appendix 1	MicroStar Data Format	49
Appendix 2	Software Installation and System Notes	55

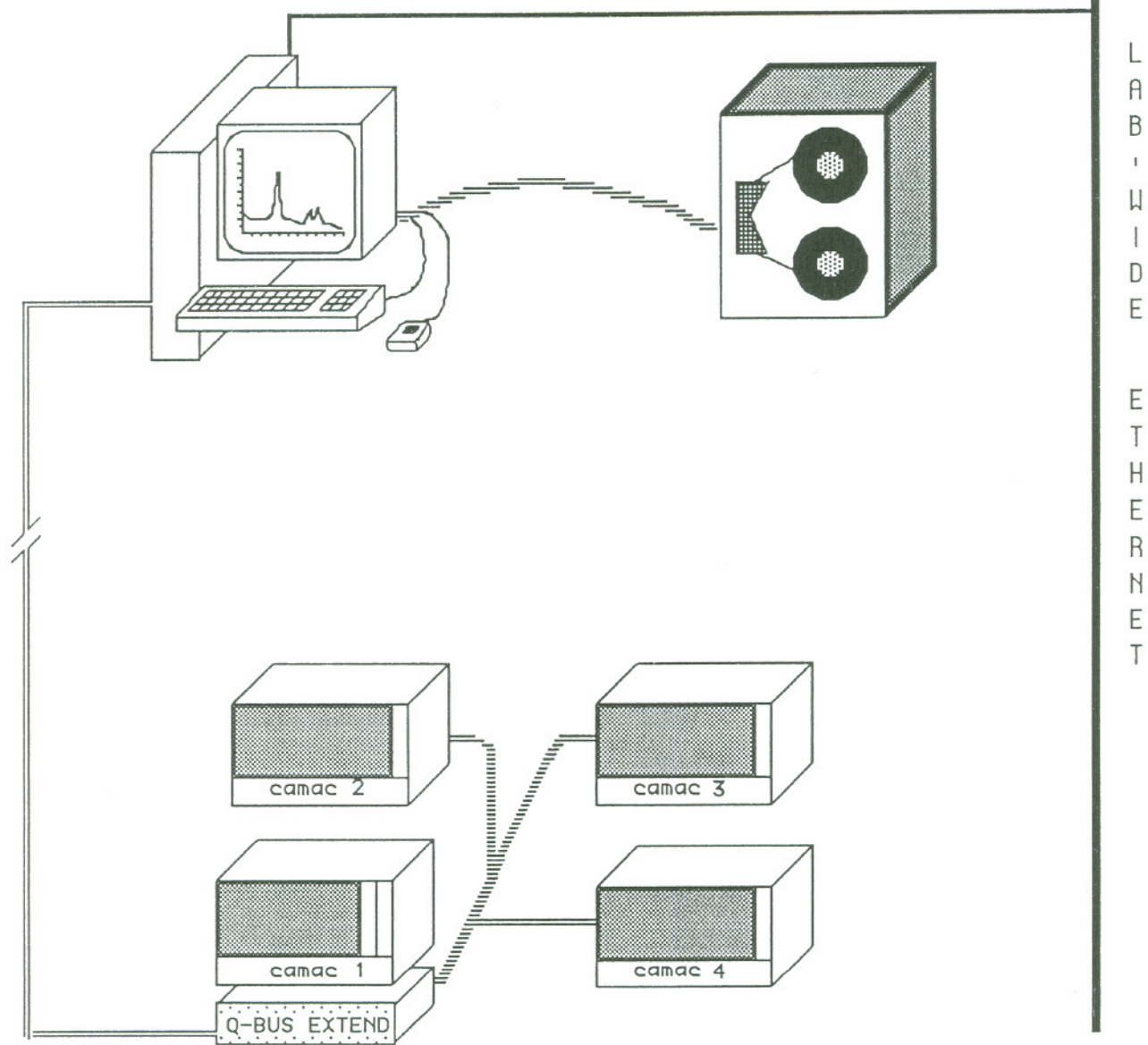


Figure 1. The MicroStar acquisition and analysis system interfaces up to four CAMAC crates to a MicroVAX workstation. Network resources are available through the labwide ethernet.

I. INTRODUCTION AND HISTORICAL OVERVIEW

MicroStar is the latest evolution of a general-purpose, CAMAC-based multi-parameter interactive data acquisition and analysis system developed at the Lawrence Berkeley Laboratory and currently in use at several research laboratories and educational institutions around the world. The roots of the system reach back to an early interactive display program, *DERTAG*, developed in the early 1970's by Dr. Creve Maples of LBL. *DERTAG* was conceived to do what many considered impossible: to provide a common program that could be easily used by the wide variety of experimenters, from atomic- to particle-physicists and nuclear chemists, who use the diverse facilities of the Lawrence Berkeley Laboratory. One early skeptic kept saying *That'll be the day!*; hence when the first early version of the program was completed its name became *Der Tag* (The Day).

The original program ran on an SCC-660 computer with 32 K-words (by 24-bit) memory. As experiments grew more complex, the limits of the SCC-660 were soon realized, and the search for more-modern, high-speed real-time computers led to the purchase of several ModComp-IV computers. These machines offered superior real-time performance, but were somewhat limited in programming tools and did not offer the large virtual-memory addressing capabilities or the large user-base of the DEC VAX 11/780 computers. But real-time performance was of highest importance; as software was developed to take advantage of the hardware capabilities of the ModComps the *DERTAG* system evolved into *MINUS3*, meaning 'an inversion of the Third Law of Thermodynamics, namely, bringing Order out of Chaos'. For short, the system became known as *CHAOS*. This system utilized LBL-developed fast multiplexed ADC's, but was limited in its portability or usefulness to other institutions because of its customized hardware and lack of a CAMAC data acquisition interface. In 1980 LBL Staff Scientist Bill Rathbun assumed responsibility for the continued development of the system, and added CAMAC acquisition as an option in the system. CAMAC soon became the preferred mode of data acquisition. Later, a CAMAC-based front-end pre-processor, the *Starburst*, (which included a PDP-11/73 computer-on-a-chip) was incorporated into the system to allow front-end zero-suppression and data compression, providing greater throughput. The incorporation of the front-end *Starburst*, itself a real-time computer, obviated the need for real-time performance in the host computer. By the mid 1980's the memory limits of the ModComps had become burdensome, and the long-awaited MicroVAX was soon to be introduced. The cost/performance ratio of the MicroVAX was so much improved over the previous generations of computers that a laborious software conversion effort was undertaken. Within a year the MicroVAX/*Starburst* (*MicroStar*) system was online.

The *MicroStar* data acquisition and analysis software consists of several independent processes, although to the user it looks like a single program. These programs handle such functions as data acquisition (if any), data analysis, interactive display and data manipulation, and process monitoring and control. An overview of these processes and functions can be found in the following paragraphs. A more detailed description and users' guide can be found in the subsequent chapters.

Data Acquisition: Beginning with Chaos

Event-mode data are acquired by the front-end *Starburst* auxiliary crate controller according to instructions downloaded by the host Microvax workstation. The event readout is triggered by one or two external signals provided by the user. (See *Acquisition Hardware Setup*, page 46). By default, raw data are zero-suppressed, formatted and blocked into 4096 x 16-bit word buffers before being transferred to the Microvax for recording and/or analysis. The default format for this data is described in Appendix 1. By user option, specialized code can be written and downloaded into the front-end *Starburst* to provide such features as: customized data formatting; selective data readout based on CAMAC bit registers; software data suppression or gating; or the setup and acquisition of FASTBUS data through a CAMAC-FASTBUS interface. In this mode, any format of data can be acquired so long as the raw event size does not exceed 512 words. Blocks of acquired raw data are transferred to the Microvax over a parallel DRV11 (or equivalent) interface.

Using standard DRV11-wa interfaces, the Microvax and the CAMAC crate(s) can be up to 50 feet apart; using optical DRV11 equivalents, the distance can be extended to a kilometer. The data are acquired and logged to tape (or disk, or just to memory to be analyzed) at elevated priorities, so that no on-line analysis can interfere with live data collection. Under typical conditions, data rates up to 40,000 words/second can be sustained.

Data Analysis: Getting Order out of Chaos

The default data analysis consists of building histograms of every parameter collected, and providing generic parameter labels. Optionally, the user can: define parameter labels, spectra sizes and parameter resolutions; create mathematical transformations of one or several raw parameters, creating *pseudo-parameters*; create a customized subroutine (written in any DEC-supported language and dynamically loaded into the running system) to perform calculations or transformations not possible with a simple mathematical expression; create a subroutine to unpack custom-formatted data; specify two-dimensional spectra to build and display; perform simple or complex gating on real or pseudo-parameters; perform complex free-form two-dimensional gating to create additional 1-D or 2-D spectra (these free-form gates can be logically combined with other 1-D or 2-D gates, or can be standalone). Optionally, free-form gates can be used as an analysis prefilter to speed-up data analysis. Additional analysis, such as multi-parameter peak fitting, can be accomplished using the interactive display program.

Interactive Display: The Visualization of Data

The interactive display programs provide on-line and off-line tools for the display, analysis and manipulation of experimental data. Analysis conditions, such as the setting of correlation conditions through simple and complex 1- and 2-dimensional gating, can be created or altered dynamically and interactively. Histogram spectra can be saved and recalled easily.

The 1-D interactive display program, Multi_D, is automatically started by the Chaos program. The 2-D display program, Space, is started by Multi_D when the user "clicks on" the "View 2-D" icon. Similarly, clicking on the "View 1-D" icon while viewing 2-D data will return the user to the 1-D display.

If for some reason the display program(s) appear to crash or lock-up, Multi_D can be restarted by typing "LOOk" to the "Enter Command" prompt.

II. LEARNING THE SYSTEM

The best way to become familiar with the software is to use it from the beginning.

The MicroStar software, once installed on a Vaxstation system, can be executed by typing **USTAR**

in response to the DCL prompt. (See Appendix 2 for information on software installation). MicroStar must be run on a Vaxstation; the software cannot be run on a VAX without a video graphics subsystem.

The software creates several processes; each process will create and utilize one or several *windows* on the screen. While using the software, the user must be aware of which window is currently accepting input from the keyboard. The user can select a window to be connected to the keyboard by "clicking" on it-- i.e., by moving the mouse pointer to the outline of the window and clicking the leftmost mouse button.

When activated, the program will first prompt:

Enter Number of Parameters per Event.

Enter the number of raw parameters in an event. If an event is of variable size, enter the *maximum* event size. The software can currently handle up to 350 parameters. By default, each parameter will have a spectrum created for it. Additional spectra can be created for mathematically transformed spectra (pseudo-parameters) and gated spectra (spectra created from a base parameter which is in coincidence or anti-coincidence with user-selected regions of one or several other spectra) by commands to the sorting program.

The program will next prompt,

Amend CAMAC Setup? (Yes,No,Change,Delete,Insert,List,Program,Vme,User)

This is where the data acquisition setup is defined. Enter the appropriate single-letter response to the prompt. The system automatically saves the last-defined data acquisition setup. The setup is simply a list of CAMAC commands to be used to read out an event. Several options are available. The default is to execute a user-specified list of CAMAC commands on every event trigger. The software will automatically zero-suppress and format the data into the *MicroStar* data format. (Appendix 1) This setup can be created and edited by the **Yes, No, Change, Delete** and **Insert** commands. Optionally, the user can program the front-end *Starburst* processor (type **P** or **U**) using the *LSB* downloading tool. This is a limited cross-assembler which can be used to perform CAMAC operations, selective data readouts and preprocessing at interrupt level in the *Starburst*. Refer to the section *Using LSB* (page 42) for information on front-end processing. Typing **P** will select *program mode* which will automatically zero-suppress and format all data read out by the user's front-end program. Typing **U** will select *user mode* which will allow the data to be transferred without zero-suppression and formatting. In this mode, the user must supply a special *user_unpack* subroutine to the analysis program.

To analyze data being acquired by LBL's VME-based data acquisition system, type **V**. To analyze data stored on tape or disk, and to avoid down-loading the *Starburst*, type **V**; this will tell the program not to download any CAMAC commands but to enter analysis mode; this command is also used to analyze data from a network data source if live data collection is turned on. (This is applicable only in conjunction with LBL's VME-based data acquisition system in use at the Bevatron)

If default CAMAC-based data acquisition is desired, the addresses of the CAMAC modules to read out must be specified here. To list the current acquisition setup (if any) type **L**. To append to an existing setup, type **Y**; to insert CAMAC commands into the current list, type **I** followed by the number of the command to insert after; e.g., **I 0** would insert a command at the very beginning of the list. To delete the old setup and create a new one, type **D 1-999** or **D 1-n** where **n** is the total number of commands in the list; after all the old commands are deleted, type either **Y** or **I 0** to begin inserting new commands from the beginning. When the acquisition setup list is complete, type **<control>-Z** to terminate the list.

Next, the software will list any *startup* or *shutdown* commands currently defined. *Startup* and *shutdown* commands are simply lists of CAMAC commands to be executed each time the data acquisition is started up (startup) or stopped (shutdown). Such commands are useful for gating on or off devices such as NIM integrators (via CAMAC output registers) or for setting gain/offset parameters on some ADC's. The software will query, **Do you want to change startup/shutdown commands?**. Answer Y (yes) or N (no). If you answer Yes, then you will be asked to enter the Crate, Number, Address, Function and Data for each CAMAC command. Always enter five numbers per command; if the command does not require a data word, enter a zero anyway. Up to sixteen commands are allowed for *startup* and sixteen for *shutdown* commands. Terminate each list with a **<control>-Z**. If standard CAMAC acquisition has been selected, the system will now attempt to download the CAMAC-based *Starburst* processor with the appropriate CAMAC event-readout instructions. At this point, the *Starburst* must already be running the USTAR program. (As normally configured, the *Starburst* will automatically start the USTAR program each time it is bootstrapped). See the section *Using Starburst* (page 41) for more information on bootstrapping the *Starburst*.

At this point the software will enter the analysis setup mode. Analysis setup commands are described in detail in Chapter IV, *Data Analysis: Options and Commands* (page 11). By default, the software will assume all data consist of 11-bits, and will create a 512-channel spectrum for each parameter. If default conditions are sufficient, type N (no more changes) to the prompt, **Select type of display information**. Otherwise, refer to Chapter IV to learn how to create 2-dimensional spectra, pseudo-parameters, gated spectra and more. When the analysis setup is complete, type N to return to the main control part of the program.

At this time there will be several windows and icons on the screen. At the lower part of the screen there will be a small box marked *VIEW LIVE DATA*. Clicking on this box with the leftmost mouse button (if the workstation is setup for a right-handed mouse-- refer to the Vaxstation manual for setting up a right- or left-handed mouse) will activate the interactive data display. When activated, you will see a large spectra-display window, with a small *control panel* beneath it. Spend some time getting familiar with this part of the software. Initially, spectrum # 1 will be displayed on the screen. There may or may not be any data in the spectrum; if there are, they are leftover from the previous use of the program.

The interactive display programs interface to the user primarily through the mouse. The mouse is used to turn various options on and off, and to mark regions of interest in a spectrum. There are three mouse buttons; the software uses the leftmost button (of a right-handed mouse) as an intermittent switch-- ON when pressed, OFF when released. The middle button is used only to turn ON some option-- releasing the button will have no effect; the rightmost button is used only to turn options OFF-- clicking this button on an option that is currently OFF will have no effect. Try a few things to get used to these buttons. Click the VIEW ALL icon with the middle button and see what happens. Then click-on the VIEW ONE icon. Options that are currently available are outlined in red on a color system. Options that are currently active are also outlined in yellow. Many options are available, both through icons and keyboard input. Click the OTHER OPTIONS icon in the center of the control panel. Notice what additional options become available. Click the KEYBOARD OPTIONS in the center of the new control panel; a window will pop-up allowing you to enter additional options. Before typing anything into this window, make sure the window is *selected* by clicking on the upper border of the keyboard-options window. See Chapter V, *Interactive Analysis and Display* (page 27) for further information on keyboard options. But for now, type X to exit the interactive display mode. The spectra display will shrink to an icon, and the master control window should be waiting for input with the prompt, **Enter Command**. For safety, always click-on the border of a keyboard-input window before typing anything into it-- be sure the window is selected for input before inputting anything to it. The program attempts to automatically select the proper window for you, but sometimes that is not possible.

III. DATA ACQUISITION: COMMANDS AND EXAMPLES

Acquisition and Playback Command List

These commands are entered in response to *** ENTER COMMAND.

A complete description of these commands is given in the pages following this table.

CONTROL OF DATA ACQUISITION

(In response to *** ENTER COMMAND) type:

- NR - New run: to begin acquiring data.
Note: Runs entered with a negative run number will not be recorded on tape.
(Useful for test runs). Stop the run by clicking-on the *Shut Off Data* box.
- CP - To change the number of parameters/event, to redefine event readout & download Starburst. Use this command whenever power has been cycled on the CAMAC crate, after the *Starburst* processor has been rebooted.
- NZR - No zero: Prevents the automatic zeroing of display buffers when processing new data. Effectively adds different runs together. This is a *toggle* command; that is, typing **NZ** again will select zero mode.
Note: All display buffers will be zeroed when the ADC is enabled or a new run is read unless the **NZR** command is used.

TAPE CONTROL AND DATA REPLAY COMMANDS

- AS - Assigns the mag tape or disk file for storage or retrieval of event recorded data.
- NT - To initialize a new tape for writing data.
*** Warning *** Previous data destroyed
- RD,N - To read the next N runs from tape.
(Default: N=1); The analysis results are cumulative.
- AU - To automatically sort and sum all runs from current tape position to end-of-tape.
- LC,N - To locate and read a specific run (N is optional).
- AR,N - To advance N runs and return (default N = 1).
- BR,N - To back up N runs and return (default N = 1).
- RW - To rewind tape.
- ET - To position tape at the logical end-of-tape.
- WET - To write an emergency logical end-of-tape mark.
*** Warning *** Only necessary if the tape writing is abnormally terminated.
The tape is assumed to be in the proper position.
- DM - Dismount currently assigned tape; use before changing tapes.

OTHER COMMANDS

- SET - To alter data analysis or display conditions.
This command invokes the analysis task **Order**.
(For a listing of Analysis Commands see Chapter IV, page 11)
- LOOK - Restarts the display program *Multi_D* in case of lockup or crash.
- END - To terminate *MicroStar* and all related tasks.

NR - Start New Run

The **New Run (NR)** command is used to start the acquisition of a new data set. Although data may be acquired with or without recording on magnetic tape, event-mode data generally are recorded on tape. If tape recording is to be employed, the tape unit must have been previously specified using the **ASsign** command (see above). Furthermore, the tape must first be 'initialized' using the **NT (New Tape)** command described on the following page.

If you wish to collect data without recording on magnetic tape, (e.g., for testing either your analysis setup or front-end hardware), **use a negative #** in response to the question 'Enter Run #'; otherwise, the tape must be positioned at the end of the last run on the tape, or must be initialized and positioned at beginning-of-tape.

*** ENTER COMMAND (?? FOR LIST)

NR

*** ENTER TITLE (32 CH. MAX.)

TEST

*** ENTER RUN NUMBER

- 1

--TAPE RECORDING DISABLED-- ENABLE ADC? (Y/N)

Y

--ACQUISITION STARTED - 18-SEP-1990 11:33:26

Note: For protection, the program prohibits writing on a tape unless the tape is positioned at the logical end-of-tape (**ET** command) or the tape is first initialized (**NT** command). The tape will always be positioned properly when a run is stopped normally and no other tape commands are used.

Once a data acquisition run is started using the **NR** command, the acquisition program will accept no further commands from you until the run is stopped. Data acquisition can be stopped by clicking on the *SHUT OFF DATA* icon in the *Monitor* window. This can also be used to pause or stop the analysis of data played back from tape or disk.

CP - Change number of parameters per event

This command is similar to restarting the program; it should be used when the physical event is to be redefined, or after the front-end *Starburst* processor has been rebooted, such as after a power-cycle on the CAMAC crate(s). This will cause the front-end *Starburst* processor to be downloaded with the necessary event readout commands.

NZR - No Zero

The **NZR** command effectively allows successive runs to be summed. Normally, all display buffers are zeroed whenever a new run is started (**NR**) or sorted from tape (**RD**). If one desires to sum histograms over several runs, the **NZR** command should be invoked to inhibit the clearing of all display memory.

AS - Assign Tape Unit Number

This command is used to assign the device to be used for the storage and retrieval of event-mode data. There are no default assignments. The tape or disk unit to be read or written is defined by this command.

AS

Select DEVICE ASSIGNMENTS for EVENT data I/O:

1. . . For 1600/6250 BPI mag tape MSA0:
2. . . For 8mm tape
3. . . For ModComp data on msa0:
4. . . For ModComp data on 8mm
5. . . For ModComp data, 1000 words/rec on disk
6. . . For data acquired by USTAR on disk, 4k words/rec
7. . . For ModComp data on disk, 512 bytes/rec
8. . . For uSTAR data copied from tape to disk, 4k words/rec

Type 2 to select the 8mm Exabyte tape drives.

The software will query **Enter device name (include colon)**. Typically, the 8mm tape drives are units MUA0: and MUA1:. Type in the appropriate name. Note: before the tape is assigned, be sure it is physically mounted in the drive; when using 8mm tape units, always be certain that the green ready light is on before issuing any tape commands. Also never use the eject button on the Exabyte tape drives when using the MicroStar software. To dismount and eject an 8mm tape, use the **DM** (dismount) command in response to Enter command.

If acquisition to or playback from disk is desired, select device assignment 6. The program will query **Enter complete file name**. Enter the file name; if necessary, include the complete file specification, using the standard DCL file specification syntax:

NODE::DEVICE:[DIRECTORY]filename

NT- Initializing a Magnetic Tape

In order to minimize the possibility of a good tape being written over through user-error, the program attempts to read the currently-assigned tape before allowing a new run to be stored on that tape. Tape recording will not be allowed unless the assigned tape is currently positioned at the logical-end-of-tape. Whenever a run is normally terminated, the tape will be positioned at the logical-end-of-tape. However, to begin recording on a tape, a 'logical-end-of-tape mark' must be written onto the beginning of the tape. This is accomplished with the 'New Tape' command. The program will subsequently write over this end-of-tape mark, and write a new end-of-tape mark after the last run recorded.

*** ENTER COMMAND (?? TO LIST)

NT

*** INITIALIZE TAPE? (Y/N)

WARNING - PREVIOUS DATA DESTROYED

Y

. . . MUA0: INITIALIZED

*** ENTER COMMAND

Note: File mark operations are very slow on the 8mm tapes (approximately 1 minute to write an end-of-tape). When using these tapes the program will 'remember' the last tape operation performed, and will avoid reading the logical- end-of-tape mark. Thus if a user takes data to the 8mm tapes, and between runs does some other tape operations, such as re-sorting a previous run, the program will recognize that the tape has been repositioned, and will refuse to write onto the tape until the tape has been explicitly positioned to the end, (**ET** command) or is re-initialized.

RD - Read New Run from Tape

The **RD** command is used to read and sort event-mode data. The tape drive must be properly assigned (see **AS** command), and the tape positioned at the start of an event-mode run. The **RD** command will cause the run's header record to be read and verified; if the tape is positioned at the start of an event-mode run, the message

SORT RUN? (Yes/No/Auto_skip_bad/Skip_records)

will be typed. If the tape is not properly positioned, the message

DATA INCONSISTENT, OR TAPE NOT PROPERLY POSITIONED

will appear, and the tape will be repositioned to the beginning of the current file.

To begin data sorting, at the appearance of the message:

SORT RUN? (Yes/No/Auto_skip_bad/Skip_records) type
Y

The other options are: **A** (auto-skip bad records) used to automatically skip unreadable tape records (useful when trying to analyze data from a tape that has developed parity errors); and **S** (skip records), used when one wants to continue analysis from a known location on a tape.

AU - Automatic Read and Sort to End of Tape

The **AU** command allows all runs to be sorted and summed from the current tape position to the end of tape.

LC - Locate Run on Tape

The **LC** command allows the currently-assigned mag tape to be positioned to the beginning of a specified run for playback and analysis (through the **RD** command). The tape will be searched from the current location to the logical-end-of-tape mark.

AR - Advance Run(s) on Tape

The **AR** command advances the currently-assigned tape the specified number of runs (or 'file marks') from the current position. (E.g., **AR 4** advances to the fourth run past the current position).

BR - Backup Run(s) on Tape

The **BR** command backs up the currently-assigned tape the specified number of runs from the current position.

RW - Rewind Tape

The **RW** command positions the currently assigned tape to the beginning-of-tape mark.

ET - Position to End of Tape

The **ET** command positions the currently-assigned tape to the logical-end-of-tape mark. This is most useful to skip past previously-recorded runs in order to store a new run on the same tape.

WET - Write End-of-Tape mark

This command is only used in the case of an emergency, such as a restart after a computer crash during acquisition.

Warning: if it is necessary to use this command, be sure the proper tape drive is assigned, and the tape properly positioned.

The program assumes the tape is properly positioned, and no safety checks are made.

*** ENTER COMMAND (?? TO LIST)

WET

WRITE EMERGENCY END OF TAPE? (Y/N)

Y

DM - Dismount Tape

The **DM** command dismounts the currently assigned tape. This command must be used when changing tapes. **Warning:** this command **must** be used to dismount one tape before mounting (via **ASign**) a new tape. Starting with VMS version 5.0, the tape unit will hang forever in a 'mount verify' loop if you manually remove one tape and load another without first explicitly dismounting the first tape.

SET - Change Display/Analysis Setup

The **SET** command transfers control to the analysis/display setup part of the software.

LOOK - Re-execute Interactive Display Program

This command is used to manually activate the interactive display program **Multi_D** if it crashes or otherwise locks up. This is a 'last resort' command.

END - Terminate the Program

The **END** command terminates *MicroStar* and all related tasks

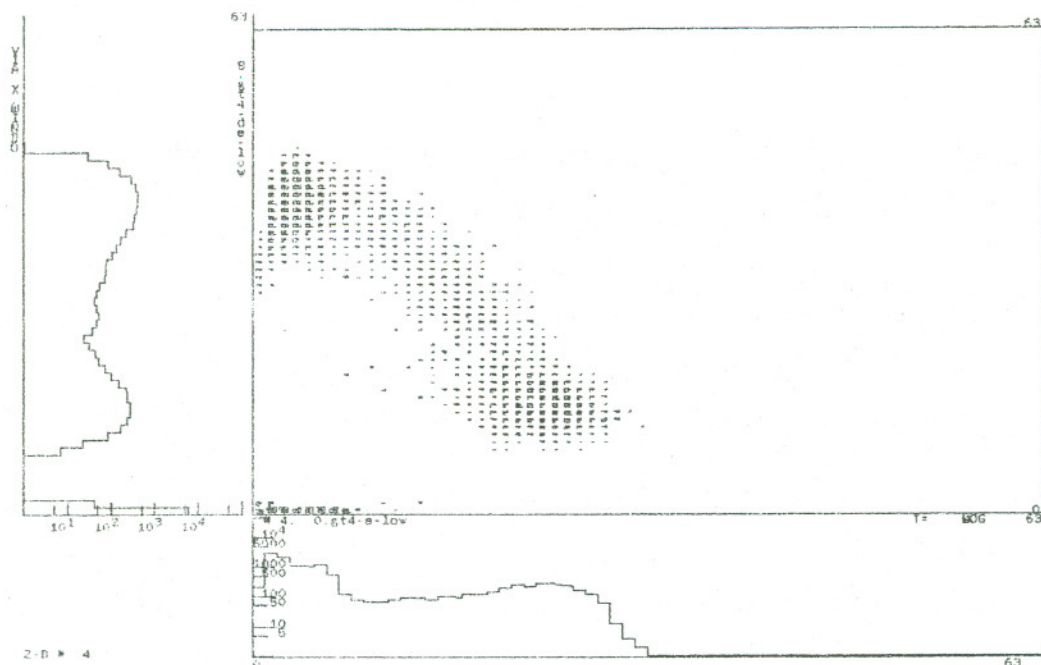
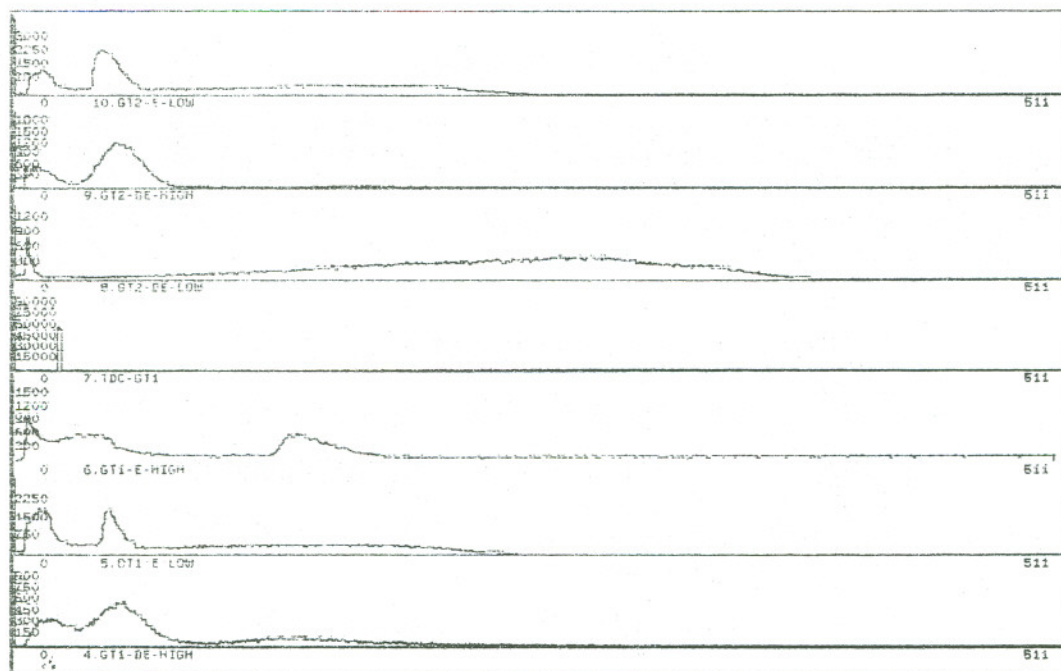


Figure 2. Data-analysis commands not only specify analysis conditions and algorithms, but also define the spectra to be created and displayed. Above, a typical 2-D spectrum viewed with its X- and Y-projections; below, a typical multi-spectra 1-D display.



IV. DATA ANALYSIS: OPTIONS AND COMMANDS

Data analysis and display setup commands are entered in response to the prompt, **Select type of Display Information**. These commands can be entered during initial setup, or whenever the acquisition or playback is stopped. In response to the prompt from the acquisition control part of the program, **ENTER COMMAND**, type **SET**. The program will then respond **Select type of Display Information**. Select from the following list the analysis and display options wanted.

COMMAND LIST FOR SORTING TASK

Analysis and Display Setup Commands

(In response to **Select Type of Display Information**)

The following commands may be used to select specific sorting categories for review or change.

- ALL** - Review all display information.
- PS** - Select pseudo-parameters.
- SN** - Select singles spectra information.
- TD** - Select two-dimensional spectra.
- GS** - Select gated singles information.
- FS** - Select freeform-gated singles information.
- GT** - Select gated two-dimensional spectra.
- FT** - Select freeform-gated two-dimensional spectra.
- LA** - Select spectrum labels.
- SR** - Set resolution; define # of bits of data in a raw parameter. Defaults to 11 bits.
- FI** - Select prefiltering data through free-form gates.
- L** - List current setup on console; optionally also list to printer.
- N** - No change desired, setup complete, exit.
- DF** - Default, current setup erased and default conditions restored.
- M** - List current available memory.
- RS** - Restore last complete setup (useful after a restart of program or a **DF** command).
- ARC** - Archive; store current setup conditions (including gates) on disk.
- RT** - Retrieve a particular setup from the disk.
- LD** - List all setup files in the current directory.

Edit Commands

These commands may be used to alter or review any specific information requested by one of the basic display setup commands in the list above.

- Y or Y#** - Yes or Yes, spectrum number or pseudo-parameter type.
- #** - Either spectrum number or pseudo-parameter type. (See also **A**.)
(For the advanced, a # entry may be used in place of **A**.)
- A** - All, may be used in place of a # entry.
- N** - No, no change.
- L or L#** - List or list number.
- C or C#** - Change or Change Number.
- D or D#** - Delete or Delete Number.

GENERAL INFORMATION

Additions, changes, deletions, etc., may be made in any order to any sorting category.

Changes can generally be made only when data acquisition or playback is stopped.

All setup information (except default), including gate locations, is automatically saved on disk when exiting the setup routine (using the **N** command). Thus the last complete setup may always be restored by using the **RS** command. However, since gate locations are set interactively by the display program and may be changed anytime, the user should always re-enter the setup mode (type **SET** in response to *****ENTER COMMAND**) *after all final gate locations have been set*; this will force the current gate settings to be saved with the setup.

Setup conditions can also be saved on a more permanent basis by utilizing the **AR**chive command.

When one of the options in the **Display Setup Commands** list is selected, e.g., **GS**, the program responds ***** Gated Singles Spectra? (Y,N,C,D,L)**, at which point you can make the appropriate modifications which might be required, using the **Edit Commands**.

Samples the usage of these commands are given on the following pages.

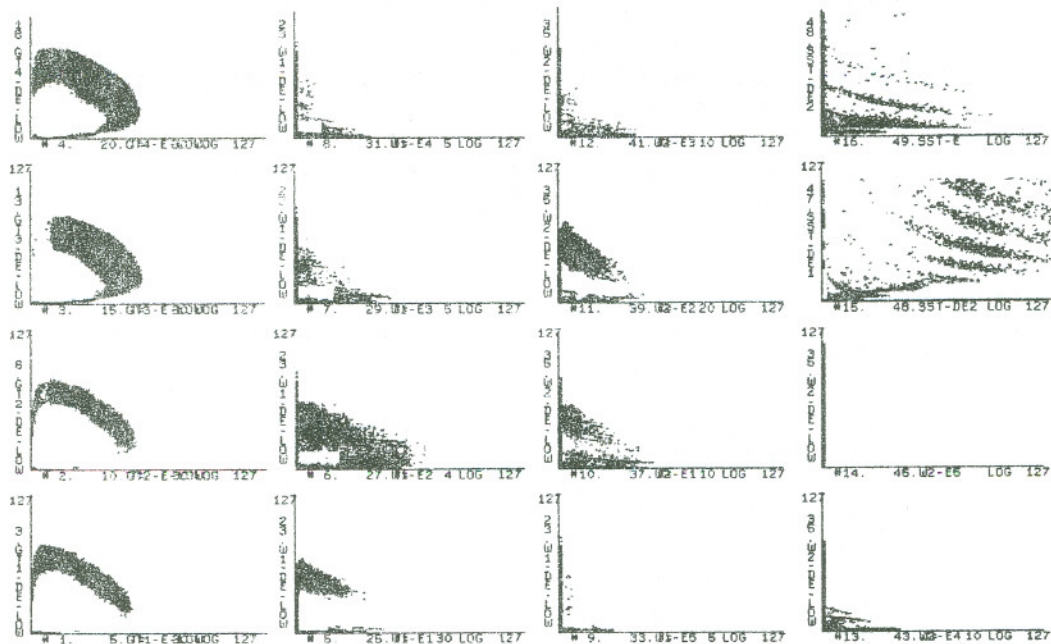


Figure 3. Up to sixteen memory-resident 2-D spectra can be defined using the analysis command **TD**. Additional '2-D virtual spectra' require no memory allocation and can be created 'on the fly'.

Analysis Options

ALL - Review All Display information

By default, the analysis program creates a 512 channel singles spectrum for every real parameter of the event. If you have other requirements, use the commands listed in the command list to create pseudo-parameters, to create two-dimensional spectra for display, to create gated singles or gated 2-D spectra, or to otherwise change the display and analysis setup. By answering 'ALL' to the prompting statement, the analysis program will query you for inclusion of all available options, as illustrated in the following dialogue.

*** SELECT TYPE OF DISPLAY INFORMATION (? FOR LIST)

ALL

*** PSEUDO-PARAMETERS (Y/N/C/D/L/?)

N

(See PS for pseudos.)

... 13 SINGLES SPECTRA

*** CHANGE # CHANNELS/PARAMETER ? (Y/N/L)

N

(See SN for singles changes.)

*** LABEL PARAMETERS? (Y/N/D/L)

N

(See LA for labels)

... 0 GATED SINGLE SPECTRA USED (OF 337 POSSIBLE)

... 512232 WORDS OF STORAGE AVAILABLE FOR SINGLES SPECTRA

*** GATED SINGLES SPECTRA? (Y,N,C,D,L)

N

(See GS for gated singles)

... 512232 WORDS OF STORAGE REMAIN

... 0 2-D DISPLAYS CURRENTLY USED (OF 16 POSSIBLE)

*** AMEND 2-D SPECTRA SETUP? (Y,N,C,D,L)

N

(See TD for 2-D setup)

... 512232 WORDS OF STORAGE REMAIN

... 0 GATED 2-D DISPLAYS CURRENTLY USED (OF 16 POSSIBLE)

*** CHANGE GATED 2-D SETUP? (Y,N,C,D,L)

N

(See GT for gated 2-D's)

*** ENTER COMMAND (?? FOR LIST)

Command Options

SN - Changing Size of Singles Spectra

The following dialogue is an example of what might be specified following the selection of the SN option.

*** SELECT TYPE OF DISPLAY INFORMATION (? FOR LIST)

SN

... 13 SINGLES SPECTRA

*** CHANGE # CHANNELS/PARAMETER ? (Y/N/L)

Y

*** ENTER SPECTRUM # (OR ALL)

3

... PARAMETER 3 = 512 CHANNELS

*** FOR PARAM. 3

1000 (1000 channels,)

*** CHANGE # CHANNELS/PARAMETER (Y/N/L)

N

*** SELECT TYPE OF DISPLAY INFORMATION (? FOR LIST)

GS - Creating Gated-Singles Spectra

The following dialogue is an example of what might be specified following the selection of the GS option to create gated singles spectra.

*** SELECT TYPE OF DISPLAY INFORMATION (? FOR LIST)

GS

... 0 GATED SINGLES SPECTRA USED (OF 337 POSSIBLE)

... 510974 WORDS OF STORAGE AVAILABLE FOR SINGLES SPECTRA

*** GATED SINGLES SPECTRA? (Y,N,C,D,L)

Y

*** ENTER # OF ADDITIONAL GATED SINGLES SPECTRA

2

*** ENTER GATE #, PARAMETER #, NUMBER CHANNELS

*** FOR SPECTRUM # 14 (GATED)

3 10 256

*** FOR SPECTRUM # 15 (GATED)

4,11,512

*** SELECT TYPE OF DISPLAY INFORMATION (? FOR LIST)

GS

... 2 GATED SINGLE SPECTRA USED (OF 19 POSSIBLE)

... 509438 WORDS OF STORAGE AVAILABLE FOR SINGLES SPECTRA

*** GATED SINGLE SPECTRA? (Y,N,C,D,L)

L

SPECT	GATE #	PARAM #	CHANNELS
1	3	10	256
2	4	11	512

*** GATED SINGLES SPECTRA ? (Y,N,C,D,L)

N

*** SELECT TYPE OF DISPLAY INFORMATION (? FOR LIST)

This action has defined two additional spectra to be created. Each new spectrum will be incremented whenever a 'hit' occurs within (or, for *NOT* gates, outside of) the associated gate. These *gates* are essentially logical conditions which can include one or multiple spectra, and can be adjusted dynamically by the user during analysis. The actual setting of these gates is accomplished interactively, and is described in detail in Chapter V, page 36. **Note:** So-called "free-form gates" will work in conjunction with normal 1-D and 2-D gates by simply giving the free-form gate the same gate # as standard gates. For more information on free-form gates, refer to Chapter V, especially the section *Setting Gating Conditions* on page 36.

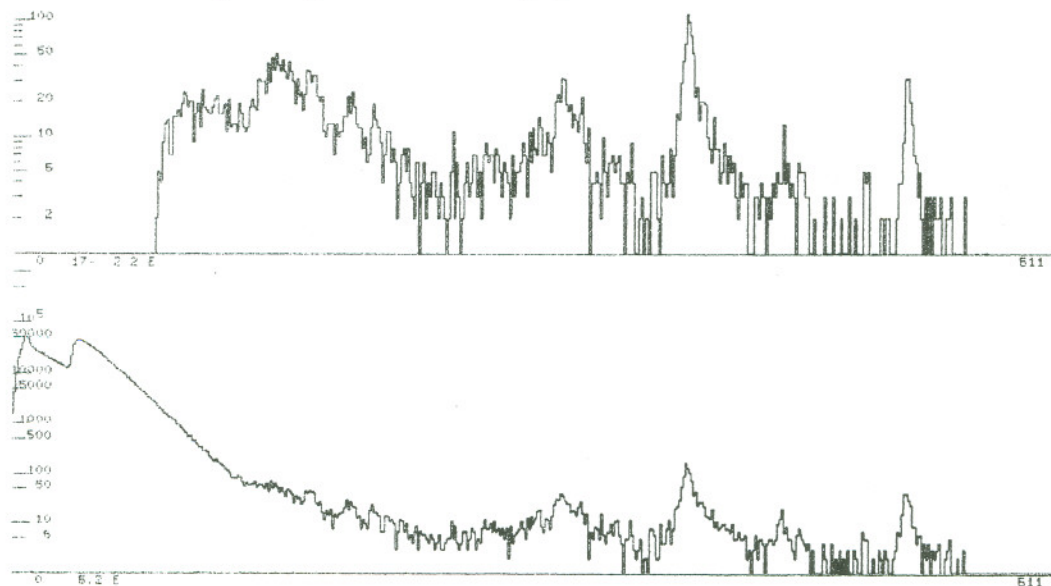


Figure 4. A gated spectrum (top) projected from a base spectrum (bottom). Gating conditions can be used to extract information otherwise obscured by background.

*** FOR 2-D DISPLAY # 2, (GATED, 492842 WORDS LEFT)
6,4,3,100,100

*** SELECT TYPE OF DISPLAY INFORMATION

LA - Labeling Spectra

*** SELECT TYPE OF DISPLAY INFORMATION (? TO LIST)

LA

*** LABEL PARAMETERS (Y,N,D,L)

Y

*** ENTER SPECTRUM # (OR ALL)

1

*** ID FOR 1

ENERGY

*** LABEL PARAMETERS? (Y,N,D,L)

Y

*** ENTER SPECTRUM # (OR ALL)

2

*** ID FOR 2

TIME OF FLIGHT

*** LABEL PARAMETERS? (Y,N,D,L)

N

*** SELECT TYPE OF DISPLAY INFORMATION (? FOR LIST)

ARC - Storage of an Analysis Setup

When the setup is complete, and verified, it can be archived as follows:

*** SELECT TYPE OF DISPLAY INFORMATION (? TO LIST)

ARC

Stores the setup, including gates, spectrum labels, and pseudo-parameters on disk. **ARC** should be used only after the setup is complete and tested (such as through usage).

*** ENTER NAME OF SETUP FILE (DEFAULT EXTENSION = .SET) **JUNERUN**

TD - Creating 2-Dimensional Spectra

The following dialogue is an example of what might be specified following the selection of the TD option to create 2-dimensional spectra.

*** SELECT TYPE OF DISPLAY INFORMATION (? FOR LIST)

TD

... 509438 WORDS OF STORAGE REMAIN
... 0 2-D DISPLAYS CURRENTLY USED (OF 16 POSSIBLE)
*** AMEND 2-D SPECTRA SETUP? (Y,N,C,D,L)

Y

*** ENTER # OF ADDITIONAL 2-D SPECTRA

2

ENTER THE X AND Y PARAMETER NUMBER AND
THE NUMBER OF CHANNELS FOR X AND Y
*** FOR 2-D DISPLAY # 1 (509438 WORDS REMAIN)

2,4,100,100

*** FOR 2-D DISPLAY # 2 (499438 WORDS REMAIN)

3 5 64 64

*** SELECT TYPE OF DISPLAY INFORMATION (? FOR LIST)

GT - Creating Gated 2-D Spectra

Gated two-dimensional displays can be defined in a way analogous to that for regular 2-D's, as described in the following dialogue.

*** SELECT TYPE OF DISPLAY INFORMATION (? FOR LIST)

GT

... 502842 WORDS REMAIN
... 0 GATED 2-D DISPLAYS CURRENTLY USED (OF 16 POSSIBLE)
*** CHANGE GATED 2-D SETUP (Y,N,C,D,L)

Y

*** ENTER NUMBER OF ADDITIONAL GATED 2-D SPECTRA

2

ENTER GATE #, X AND Y PARAM. # AND
NUM. OF CHANNELS FOR X AND Y
*** FOR 2-D DISPLAY # 1, (GATED, 502842 WORDS LEFT)

5,2,1,100,100

*** SELECT TYPE OF DISPLAY INFORMATION (? TO LIST)

N

(No change)

*** ENTER COMMAND

Note:

The command **RS** is used to recover the current setup conditions. It is used after restart of the program or after the use of a **DF** command. If the number of parameters per event is redefined, (made larger or smaller) a new setup must be created. In this case, use **DF** (default) to reset to default conditions; then use setup commands to define a new setup. For this reason, be sure to initially define the event to be large enough to handle all parameters you might want to take, especially if your experiment is large and complex.

RT - Retrieval of an Analysis Setup

... DEFAULT CONDITIONS IN EFFECT.

... CURRENTLY UNALLOCATED MEMORY = 510974 WORDS
(AVAILABLE FOR SINGLES = 510974 WORDS)

*** SELECT TYPE OF DISPLAY INFORMATION (? TO LIST)

LD

Listing of Setups:

JUNERUN.SET	19-JUNE-1990 14:43:22
JUNERUN1.SET	20-JUNE-1990 09:53:47
LAST_ORDER_SETUP.SET	10-MAY-1990 23:56:32

*** SELECT TYPE OF DISPLAY INFORMATION (? TO LIST)

RT

*** ENTER NAME OF SETUP FILE (DEFAULT EXTENSION = .SET) **JUNERUN**

*** RESTORE THIS SETUP? (Y/N/L)

L

(List - let's see for sure what it has in it.)

ID	NO. PARAM.	SNGLS. SPEC.	GATED SNGLS.	2-D SPEC.	GATED 2-D	PSEUDO PARAM.
JUNERUN.SET	23	23	9	3	6	0

*** RESTORE THIS SETUP? (Y/N/L)

Y

... SETUP RESTORED

*** SELECT TYPE OF DISPLAY INFORMATION (? TO LIST)

N

(Setup complete; no change desired; exit)

*** ENTER COMMAND

CREATION OF PSEUDO-PARAMETERS - PS

The use of artificial or 'pseudo' parameters is a particularly important tool in analysis. Several options are available to achieve transformations.

The following types of pseudo-parameters are available:

1. Summation: $DE1/A + DE2/B + E$
2. Particle ID: $(DE1/A + E)**P - E**P$
3. Time-of-Flight (Mass): $(E + K)*(T - T0)**2$
4. Time-of-Flight (Energy): $E*(1/(T - T0)**2)$
(The presence of parameter **E** is optional.)
5. Position: $E1/(A * E1 + B * E2)$
6. Series 2: $A(X + B * Y + C * (X * Y)**P)$
7. Series 1: $A + B * X + C * X**2 + D * X**3 + E * X**4 + F * X**5$
8. Multiply: $A + B * X + C * Y + D * X * Y$
9. Square Root: $SQRT(X)$
Calculator Pad: Evaluates a one line FORTRAN expression **DEACTIVATED**
11. PLOT XY - Creates a quasi-2D plot of parameter X vs parameter Y
12. User Subroutine; creates spectra based on a user-supplied subroutine.

Notes:

All data are normalized to 4096-space at the time of event-unpacking and before any pseudo-parameter calculations are performed. All pseudo-parameter calculations must return data in 4096-space through appropriate normalizations.

The variables DE1, E1, E, X, etc. refer to existing parameters; B, C, P etc., are constants.

For purposes of speed, the calculations for pseudo-parameter types 1 through 9 are carried out using integer arithmetic or table lookup techniques. If tables are required, the necessary memory is obtained from the user's current display buffer.

If the basic list does not provide the desired transformation, the list may be chained to produce more complex results. A new parameter, say number 15, could be created using equation (1). This value could then be used as an input to another equation, such as E of equation (3). Further, if the parameter 15 is by itself not interesting, the final result of the second transformation could also be stored as parameter 15. If the necessary transformation still cannot be obtained, the user can create a dynamically-loaded subroutine to perform the calculation. (pseudo type 12)

Pseudo-Parameter Option List:

- Y** - Yes, add a new pseudo-parameter.
- N** - No further pseudo-parameters; no changes.
- C** - Change conditions of pseudo-parameter number #.
- D** - Delete pseudo-parameter number #.
- DA** - Delete all current pseudo-parameters.
- LA** - List all current pseudo-parameter setups.

Example of Use of Pseudo-Parameters

*** SELECT TYPE OF DISPLAY INFORMATION (? FOR LIST)

PS

*** PSEUDO-PARAMETERS (Y/N/C/D/L/?)

?

... OUTPUT ON PRINTER

*** PSEUDO-PARAMETERS? (Y/N/C/D/L/?)

Y (There are shortcuts in this dialog for *expert* users, as shown later in this example)

*** ENTER TYPE OF PSEUDO-PARAMETER

1

1 ... SUM = <DE1/A + DE2/B + E> * NORM

*** ENTER CONSTANTS A AND B (2 DEC. PLACES)

2., 0. (NOTE: using 0. for B will cause 2nd term (de2/B) to to be dropped)

*** ENTER PARAMETER # FOR DE1 AND E

1, 2

*** ENTER NUMBER OF NEW PARAMETER (DEFAULT = 14)

14

*** ENTER NORMALIZATION (DEFAULT = 1.00)

1.0

1 ... P(14) = < DE1(1) / 2. + E(2) > * 1.00

*** PSEUDO-PARAMETERS? (Y/N/C/D/L/?)

Y 2 (Combining 2 answers skips one prompt)

2 ... SIG = (DE1/A+E)**P - E**P

(THIS PSEUDO REQUIRES 2048 WORDS OF MEMORY FOR TABLES)

*** ENTER CONSTANTS A AND P (2 DEC. PLACES)

10.4, 1.73

*** ENTER PARAMETER # FOR DE1 AND E

1, 2

*** ENTER # OF NEW PARAMETER (DEFAULT = 15)

15

*** ENTER NORMALIZATION (DEFAULT = 1.0)

6.0

2 ... P(15) = <(DE1(1)/10.40 + E(2))** 1.77 - E(2)** 1.73> * 6.0

*** PSEUDO-PARAMETERS? (Y/N/C/D/L/?)
3 (or simply giving the desired pseudo-parameter type # is sufficient)

3 ... TFM = < ((E+K)*(T - T0)**2 / 4096) * NORM
*** ENTER INTEGER CONSTANTS K AND T0 (4096 SPACE)
0,2000

*** ENTER PARAMETER # FOR E AND T
2,5
*** ENTER # OF NEW PARAMETER (DEFAULT = 16)
16

*** ENTER NORMALIZATION (DEFAULT = 1.0)
1.

3 ... P(16) = < (E(2) + 0)*(T(5)-2000)**2 > * 1.0

*** PSEUDO-PARAMETERS? (Y/N/C/D/L/?)
4

4 ... TFE = 4096*E / ((T - T0) / A)**2 * NORM
*** ENTER CONSTANTS T0 (INTEGER, 4096 SPACE) AND A
100,2

*** ENTER PARAMETER # FOR T AND E
(Note: ENTER 0 FOR E IF PARAMETER NOT DESIRED)
5,1

*** ENTER # OF NEW PARAMETER (DEFAULT = 17)
17

*** ENTER NORMALIZATION (DEFAULT = 1.0)
1

4 ... P(17) = < 50**2* P(1) / ((T(5)-100) / 2.00)**2 > 1.0

*** PSEUDO-PARAMETERS? (Y/N/C/D/L/?)
Y 5 (or type just 5 or just Y)

5 ... POS = < E1*4096 / (A*E1 + B*E2) > * NORM
*** ENTER CONSTANTS A AND B
.5,.5

*** ENTER PARAMETER # FOR E1 AND E2
8,9

*** ENTER # OF NEW PARAMETER (DEFAULT = 17)
17

*** ENTER NORMALIZATION (DEFAULT = 1.0)
1.

5 ... P(17) = < E1(8) / (0.5*E1(8) + 0.5*E2(9) > * 1.0

```

*** PSEUDO-PARAMETERS? (Y/N/C/D/L/?)
6
# 6 ... SER = <A*E1 + B*E2 + C*(E1*E2)**0.5>*NORM
      (THIS PSEUDO REQUIRES 2048 WORDS OF MEMORY FOR TABLES)
*** ENTER CONSTANTS A,B AND C (2 DEC. PLACES MAX.)
.5,1.5,0.
*** ENTER PARAMETER # FOR E1 AND E2
3,5
*** ENTER # OF NEW PARAMETER (DEFAULT = 18)
5
... PARAMETER 5 REPLACED

*** ENTER NORMALIZATION (DEFAULT = 1.0)
.5
... P(5) = <E1(3)* 0.50 + e2( 5)* 1.5 + 0.00*(E1(3)*E5(5)**0.50 *0.50>

*** PSEUDO-PARAMETERS? (Y/N/C/D/L/?)
7
# 7 ... SR1 = <A + B*X + C*X**2 + D*X**3 + E*X**4 + F*X**5>*NORM
      (THIS PSEUDO REQUIRES 2048 WORDS OF MEMORY FOR TABLES)
*** ENTER CONSTANTS A,B,C,D,E, AND F

1.97,.56,.0034,1.9E-5,0.,0.
*** ENTER PARAMETER # FOR X
15
*** ENTER # OF NEW PARAMETER (DEFAULT = 19)
19

*** NORMALIZE CALCULATION TO 4096 SPACE? (Y/N)
N
*** ENTER NORMALIZATION (DEFAULT = 1.0)
1.

--- WARNING! MAX. AND/OR MIN. VALUES OF 3.1 AND *008.5
      EXCEED TABLE CAPACITY.
      ERRORS MAY RESULT FROM USAGE.
      (This message informs you of a potential problem if
      X (parm. 15) varies through its full range (0-4096)
# 7 ... P(19)=< 1.97 + 0.560*X(15) + 0.340E-02*X**2 + 0.190E-4*X**3 + 0.000*X**4 +
      0.000*X**5>* 1.00
*** PSEUDO-PARAMETERS? (Y/N/C/D/L/?)
8
# 8 ... MUL = <A + B*X + C*Y + (D/4096)*X*Y> / NORM
*** ENTER CONSTANTS A (INT) ,B,C,D (2 DEC. PLACES)
125, .5, .5, 0.
*** ENTER PARAMETER # FOR X AND Y
6,12
*** ENTER # OF NEW PARAMETER (DEFAULT = 20)
6
... PARAMETER 6 REPLACED
*** ENTER NORMALIZATION (DEFAULT = 1.0)
1
# 8 ... P( 6) = <125 + 0.50*X ( 6) + 0.50*Y (12) + (0.00/4096)*X*Y> / 1.0

```



```

*** PSEUDO-PARAMETERS (Y/N/C/D/L/?)
9
# 9 ... Sqrt = < Sqrt (X) > * NORM
                (EXPECTS 0 =< X =<4096, RETURNS 10*Sqrt(X)
*** ENTER PARAMETER # FOR X
8
*** ENTER # OF NEW PARAMETER (DEFAULT = 20)
20
*** ENTER NORMALIZATION (DEFAULT = 1.0)
1.67
# 9 ... P(20) = < Sqrt(X (8) ) > * 1.67

*** PSEUDO-PARAMETERS (Y/N/C/D/L/?)
11
#10 ... PLOT PX VS PY
*** ENTER PARAMETER #'S FOR Px AND Py RESPECTIVELY
15,1
*** ENTER # OF NEW PARAMETER (DEFAULT = 21)
21
# 10 ... P (21) = PLOT OF PX (15) VS PY (1)

*** PSEUDO-PARAMETERS (Y/N/C/D/L/?)
12
#11 ... USER-SUPPLIED SUBROUTINE
ENTER COEFFICIENT # TO DEFINE;
ENTER 0 or <ctrl>-Z TO QUIT
1
ENTER VALUE FOR COEFFICIENT # 1
125
ENTER COEFFICIENT # TO DEFINE;
ENTER 0 or <ctrl>-Z TO QUIT
2
ENTER VALUE FOR COEFFICIENT # 2
.4
ENTER COEFFICIENT # TO DEFINE;
ENTER 0 or <ctrl>-Z TO QUIT
0
***ENTER # OF NEW PARAMETER (DEFAULT = 22)
22
ENTER NAME OF USER SUBROUTINE < user > ( <CR> if no change)
RTE_PSEUDO
#11 ... User-supplied subroutine RTE_PSEUDO

*** PSEUDO-PARAMETERS (Y/N/C/D/L/?)

```

Use of the User-supplied Pseudo-parameter

Pseudo-parameter type 12 provides the user a mechanism to perform transformations or calculations not possible with the built-in pseudo-parameters. This subroutine can be written in any DEC-supported higher-level language. (Lower-level languages such as Macro can also be used as long as the argument-passing rules for FORTRAN-callable subroutines are followed.) This user-supplied subroutine has the form

```
SUBROUTINE any_name(UNPACKED_EVENT, PARAMETER_NUMBER, USER_CONSTANTS)
```

where

UNPACKED_EVENT is an INTEGER*2 array containing the unpacked event, with all data values normalized to 4096-space; word N of this array is parameter N.

PARAMETER_NUMBER is an INTEGER*2 value specifying the parameter number for which this calculation is to be performed (note: the user may supply only one subroutine to perform all pseudo-parameter-type 12 calculations; when different calculations are needed for different pseudo-parameters, appropriate branching logic must be provided in the user's subroutine).

USER_CONSTANTS is an array of REAL*4 values which can be specified by the user and accessed by the user's subroutine at run-time.

Notes and restrictions:

The user-supplied subroutine is dynamically linked into the analysis system at run-time. The subroutine must be compiled prior to run-time. If several subroutines exist on the file specified by the user, the linker will create an output file with the executable routines ordered ALPHABETICALLY. Thus, if the user subroutine calls other subroutines which are located on the same file, the user subroutine must be named something alphabetically ahead of any other routine on the file. This can be accomplished, if necessary, by adding "AAA" to the beginning of the subroutine name, e.g.,

Subroutine AAA_user(unpacked_event,pseudo_number,user_constants)

Also, the dynamically-loaded subroutine cannot access any common blocks except those local to the routines located on the user-specified file.

An example of the usage of this pseudo-parameter is provided by the FORTRAN example on the following page.

SUBROUTINE USER(IBF,NP,A)

```

C
C SUM PSEUDO with offsets: performs  $P=(A_{n1}*P_m + A_{n2}*P_n + A_{n3})*A_{n4}$ 
C
C NOTE: in this example this subroutine is called 15 times for each event,
C       creating 15 new parameters. The user's data contain 39 real parameters
C       per event. These pseudo-parameters comprise parameters 40 through 54
C       (assuming that these 15 user-defined pseudo-parameters were the first
C       15 pseudo-parameters defined by the user during setup time.)
C
C IBF - Buffer of raw, unpacked event
C NPRAM - # of raw parameters per event
C NP - Spectrum # of current pseudo parameter being calculated
C      (NOTE: Pseudo #1 is spectrum # NPRAM+1 )
C A - Array of user-defined constants; defined during pseudo-setup
C
      INTEGER*2 IBF(80),NP
      DIMENSION A(256)
      DATA NPRAM /39/          ! Number of real parameters; pseudo 1 = parameter 40
C
      NX=NP-NPRAM
      IBF(NP)=0
C
      go to (10,20,30,40,50,60,70,80,90,100,110,120,130,140,150) nx
10      L= ( A(1)*IBF(2)+ A(2)*IBF(6) + A(3)) * A(4)
      GO TO 160
20      L= ( A(5)*IBF(2)+ A(6)*IBF(10) + A(7)) * A(8)
      GO TO 160
30      L= ( A(9)*IBF(2)+ A(10)*IBF(14) + A(11)) * A(12)
      GO TO 160
40      L= ( A(13)*IBF(2)+ A(14)*IBF(18) + A(15)) * A(16)
      GO TO 160
50      L= ( A(17)*IBF(2)+ A(18)*IBF(22) + A(19)) * A(20)
      GO TO 160
60      L= ( A(21)*IBF(6)+ A(22)*IBF(10) + A(23)) * A(24)
      GO TO 160
70      L= ( A(25)*IBF(6)+ A(26)*IBF(14) + A(27)) * A(28)
      GO TO 160
80      L= ( A(29)*IBF(6)+ A(30)*IBF(18) + A(31)) * A(32)
      GO TO 160
90      L= ( A(33)*IBF(6)+ A(34)*IBF(22) + A(35)) * A(36)
      GO TO 160
100     L= ( A(37)*IBF(10)+ A(38)*IBF(14) + A(39)) * A(40)
      GO TO 160
110     L= ( A(41)*IBF(10)+ A(42)*IBF(18) + A(43)) * A(44)
      GO TO 160
120     L= ( A(45)*IBF(10)+ A(46)*IBF(22) + A(47)) * A(48)
      GO TO 160
130     L= ( A(49)*IBF(14)+ A(50)*IBF(18) + A(51)) * A(52)
      GO TO 160
140     L= ( A(53)*IBF(14)+ A(54)*IBF(22) + A(55)) * A(56)
      GO TO 160
150     L= ( A(57)*IBF(18)+ A(58)*IBF(22) + A(59)) * A(60)
C
160     IF(L.LE.4096) IBF(NP)=L
      RETURN
C
      END

```

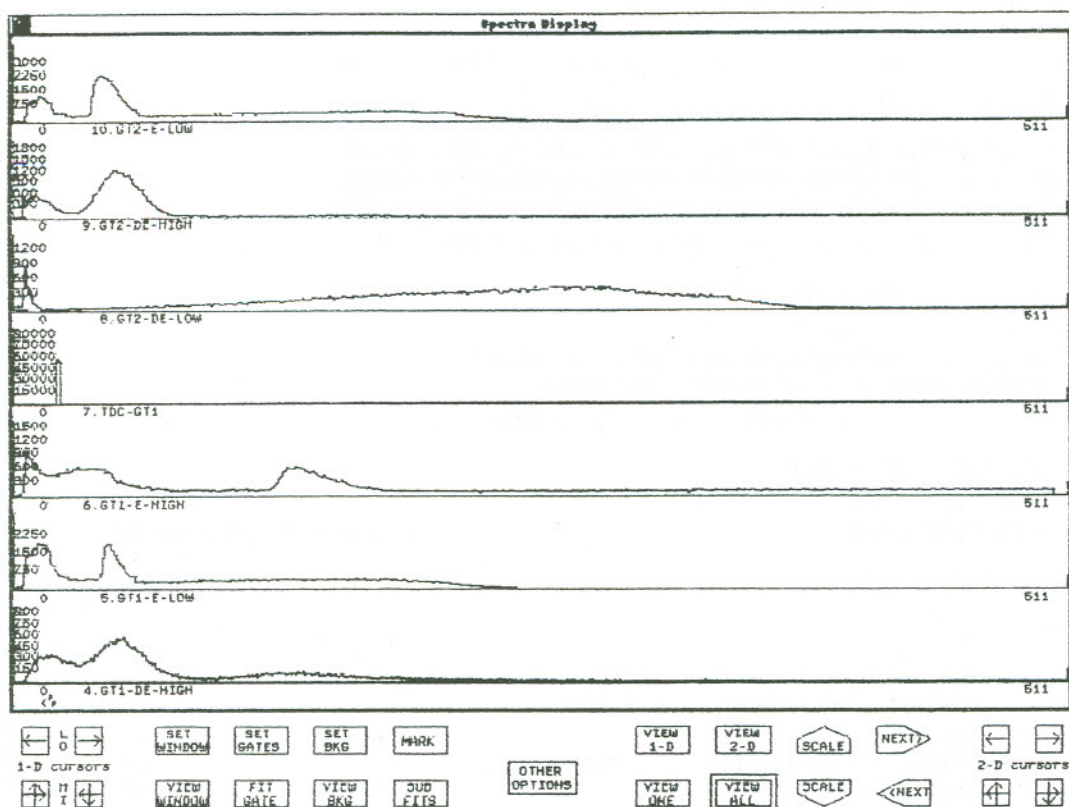
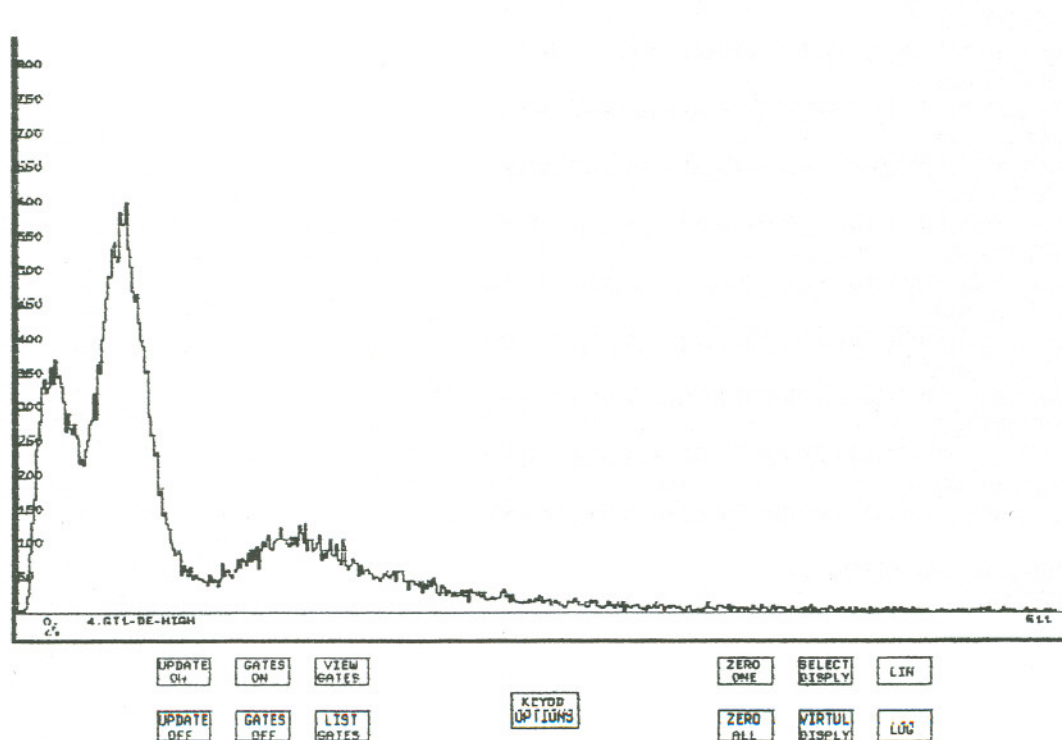


Figure 5 Interactive analysis is facilitated by utilizing graphic user interfaces rather than hard-to-remember commands. Above, the *primary control panel* is shown beneath a display of several 1D spectra. Additional options are available from the *secondary control panel*, shown beneath the display of a single delta-E spectrum (below).



V. INTERACTIVE ANALYSIS AND DISPLAY

The interactive display program *Multi-D* is automatically started by the acquisition software. When first activated, the program will display a short window at the bottom of the screen indicating choices available. These choices are *VIEW LIVE DATA*, *VIEW TAPE DATA*, *ACCESS CAMAC CRATE*, *REVIEW LAST DISPLAY*, and *END DISPLAY*. These choices are selected by clicking on the appropriate box.

To view live data acquired by the acquisition program, click inside the box marked *VIEW LIVE DATA*. Note: live data cannot be viewed while setup conditions are changing; i.e., if you are changing the setup by adding pseudo-parameters or additional gated or 2-D spectra (in response to **Select type of Display Information**) the software will not let you view spectra until all setup changes are complete.

To view 1-D or 2-D histograms previously saved on disk or tape, click-on the box *VIEW TAPE DATA*. A window will then pop up indicating additional commands available. To use auxiliary programs utilizing the CAMAC crate(s), click-on the box *ACCESS CAMAC CRATE*. These programs include utilities to 1) control high-voltage power supplies using the LeCroy 4032A high-voltage system (HV_CONTROL); 2) periodically read, save and print-out contents of specified CAMAC scalers (SCALERS); 3) setup and read-out CAMAC-based histogramming memories (HISTO). These utility programs might not be implemented on your particular system.

To view the last data set you were looking at (if any), whether live event data, saved histogram data, or data from external devices, click inside the box marked *REVIEW LAST DISPLAY*.

To terminate the display program for any reason, click inside the box *END DISPLAY*. Note: you normally don't terminate the display this way. *MicroStar* and all related processes will be terminated by the **END** command given to the acquisition program in response to the **Enter Command** prompt.

VIEW LIVE DATA

Once the analysis and display setup is complete, click-on the box marked *VIEW LIVE DATA*.

The *spectra display* window will then appear; below this window will be a *control panel* consisting of several small boxes labeled with various options. The leftmost four boxes and rightmost four boxes are labeled with arrows; these are *cursor controls* and become operational only when other options requiring cursor information are activated.

Learning to use the control panel *set window / view window*

From the right, the next boxes are labeled *set window* and *view window*. The *set window* box is used to define a region of interest in the currently-selected spectrum. To set a window and expand a region of the spectrum, first click-on the *set window* box using the middle button of the mouse (optionally, click with the leftmost button and hold it down while dragging the mouse pointer out from the *set window* box). Move the mouse pointer to someplace near the lower-energy boundary (i.e., left side) of the region of interest and click with either the middle or leftmost mouse buttons. Notice what happens. The flashing vertical bar on the screen is the low-energy cursor. Move the mouse pointer someplace else and click. The cursor will jump to the new location. Get used to positioning the cursor with the mouse pointer. Now move the pointer to the leftmost boxes of the control panel at the bottom of the screen (the *1-D cursor* boxes denoted by arrows inside). Position the pointer inside the upper box with the right-pointing arrow. Click-on this box with the leftmost mouse button. Click and hold: the cursor will move to the right; release, and the cursor will stop. Similarly, try the upper box with the left-pointing arrow. The cursor will move to the left. These two boxes control the *1-D low-energy cursors*. Once either of these boxes (<-- or -->) is clicked on, the low-energy cursor is selected, and can be rapidly positioned just by moving the pointer to the spot of interest and clicking with the left or middle mouse buttons. Make sure you don't accidentally lock-on the cursor movement, such as by clicking on the arrow with the middle button; if you do, you can click-off the option using the rightmost button, or you

can move the mouse pointer back into the *Spectra Display* window and click the cursor on the desired location. Notice that options that are active have their corresponding boxes highlighted (on color systems, outlined in yellow).

By default, the low-energy cursor is selected whenever an option requiring cursors is first activated. The high-energy cursor can be selected by momentarily clicking on one of the *1-D high energy* cursors, (left side of *control panel*, lower two boxes) and then using the mouse to mark the position of the rightmost part of the region of interest. Once the region of interest has been marked, adjusting the lower cursor will automatically cause the upper cursor to be adjusted accordingly, keeping the width of the selected region constant. Adjusting the upper (high-energy) cursor has no such effect. When initially setting a window, always set the low-energy cursor first, then set the high-energy cursor. To adjust the position of a window that is already set, and to keep the width constant, select the low-energy cursor and position the window to the desired location. Once the region of interest has been selected, click-off the *set window* option, and click-on the *view window* box (use the middle button, or click and drag with the left button). The selected region of the spectrum will be displayed. Click off the *view window* option (use the right or left button) and the full spectrum will be displayed. A separate region of interest (window) can be set for every spectrum.

WARNING: Never manually *pop* any window to the foreground while *set gates* or *set windows* is active. This will cause extremely high operating system overhead, and will appear to hang the system. If this ever happens, click-off the *set gate* or *set window* option using the rightmost mouse button, making repeated fast clicks until the option finally turns off. Note that in this case data-taking will still proceed if active, but the user's interface will appear hung.

set gates / fit gate

The next options on the *control panel* are marked *set gates* and *fit gate*. Clicking on the *set gates* box will activate a pair of cursors in a manner similar to setting a window, except that in this mode the setting of each end point is independent of the other. *Set gates* includes several functions and options. At the simplest, *set gates* can be used to determine specific channels and counts in a spectrum. Activating the *set gates* option will display the current high- and low-gate settings for the current spectrum. By default, these settings are initially the lowest and highest channels in the spectrum. Clicking on any channel in the current spectrum will move the selected cursor (initial default is the low-energy cursor) to that channel; above the cursor, two numbers will be displayed: the upper number indicates the channel on which the cursor is currently positioned; the lower number indicates the number of counts in that channel. This is most useful for determining the energy of points-of-interest in the spectrum.

By default, activating the *fit gate* option while the *set gates* mode is active will cause both the integral and centroid of the region, including end points, to be displayed on the screen. If any of various fitting options has been specified, this action will initiate the peak-fitting procedure. See the section *Peak Fitting* (page 35) for more information on multi-parameter peak fitting.

If any gated spectra have been defined, the actual gates are defined (and can be dynamically changed any time) while in the *set gates* mode by simply clicking the high and low boundaries of the gate, and then clicking the *mark* option. The *Keyboard Options* window will then pop up, asking the user to enter a gate number. Enter the gate number that was used to define the appropriate gated spectrum; if several gates are to be defined for the identical region, enter the several gate numbers, either individually or as a list, or both-- e.g., you could enter 5 to define part of gate 5, or you could enter 5, 7-20, 25, 30, 35-50 to define several gates. If a *NOT* gate is desired, enter an *N* followed by the number(s), e.g., N1-5 or N1 N2 N3 N3 N5 (note that spaces and commas are equivalent separators). If you mix *AND* gates with *NOT* gates in the same line, be sure to specify all the *AND* gates first. For example, to specify gates 1-5 as *NOT* gates and 6-10 as *AND* gates, type 6-10,N1-5. Typing N1-5,6-10 is the same as typing N1-10. For more information on gating, see *Setting Gating Conditions* (page 36).

When you have finished setting gates, always be sure to click-off the *set gates* option.

WARNING: Never manually *pop* any window to the foreground while *set gates* or *set windows* is active. This will cause extremely high operating system overhead, and will appear to hang the system. If this ever happens, click-off the *set gate* or *set window* option using the rightmost mouse button, making repeated fast clicks until the option finally turns off.

set background / view background *autoback = SET GATE + VIEW BKG*

The next two boxes to the right are *set background* and *view background*. The *set background* option is used to interactively mark points to be used for fitting 2nd- or 3rd-degree polynomials to regions of the spectrum. A separate background can be created for every spectrum.

Note: At the time of this writing, the interactive user interface for this option has not been completed.

The *view background* option is used display the background curve superimposed on the current spectrum. When used in conjunction with *set gates* and *fit gate* this option will cause the background to be subtracted channel-by-channel before the fit or integral is computed. If no interactive background has been set, engaging this option while the *set gates* option is active will cause a linear background to be displayed between the two end-points of the gate. This linear background will be subtracted before any fit or integral is computed.

mark

The *mark* box is used in conjunction with other options to mark regions of the currently-selected spectrum, for use in analysis. For example, in peak fitting you must *mark* approximate locations of the convoluted peaks; in gate setting, you select the gate positions you want using the cursor controls and the mouse, and click *mark* when satisfied that the gate is in the right position; when setting free-form gates, *mark* is used to lock-in the desired vertices of the free-form gates.

subtract fits

Sub fits is used in conjunction with peak fitting to subtract the calculated fit from the actual data and to display the residuals. This option will have no effect if fits have not been performed on the currently-selected spectrum. See the section *Peak Fitting* (page 35).

other options

In the center of the *control panel* is the box marked *other options*. Clicking on this box has two effects: a second *control panel* will overlay the bottom of the screen indicating additional options to select; and the program will enter an *idle loop* while waiting for you to make a selection. Normally, the display program will spend all available computer time updating the screen, drawing cursors, etc. This interactive computation will compete for computing resources with the analysis process, resulting either in increasing analysis time considerably or decreasing the number of events analyzed on-line. Thus **when not utilizing the interactive features of the program, it is best to put the display program in idle mode by clicking the *other options* icon.** The additional options available from this second control panel are *update on/off*, *gating on/off*, *view gates/list gates*, *keyboard options*, *zero one/zero all*, *select display*, *virtual display*, and *linear/log*. These options will be discussed in more detail following this section on the primary control panel. To escape from the idle loop, click any option on the secondary control panel, e.g., click *update on*. By default, update mode is on unless turned off.

view 1-D / view 2-D

If any 2-D spectra have been defined in the analysis setup, clicking on the *view 2-D* box will cause the 2-D display program *Space* to become active; if no 2-D spectra have been defined, you will be so informed, and the option will have no effect. The 2-D display program operates in a manner completely analogous to the 1-D program, and is discussed further later (page 37). Clicking on the *view 1-D* box will have no effect if you are already viewing 1-D data; if you are viewing 2-D data, clicking on the 1-D box will return you to the 1-D display program.

view one / view all

One or many 1-D spectra can be viewed at a time. When the interactive display is first activated the first spectrum will fill the entire spectra display window. Clicking on the *view all* box will cause all singles spectra (or a selected list of spectra - see *select display* option of the secondary control panel) to be simultaneously displayed. If you have defined a large number of spectra, it may be meaningless to attempt to display all simultaneously on the screen. It is best to use the *select display* option to select a subset of spectra to display.

scale up / scale down

The *scale* boxes are used to redraw the currently-selected spectrum with a lower or higher y-scale. The box that points upward is used to blow up a spectrum, i.e. lower the y-axis full scale (makes data bigger). The downward pointing box is used increase the y-axis full scale (makes data smaller). As long as you have the option clicked-on, the scale will be recomputed; when you click it off, the spectrum will be redrawn with the new y-scale. For this reason, **warning-- do not leave the scale option continuously on; all other options are ignored during rescaling** and you may think the program is locked-up. Use the left mouse button (intermittent on) when adjusting the scale, so that when you release your finger the option will be turned off.

next

The *next* option is used to select the next spectrum to display. The right-pointing box is used to select the next spectrum in ascending order; the left-pointing box is used to select the next spectrum in descending order. This is most useful when the *view one* option is active. If *view all* is selected, the currently selected spectrum is shown by a red marker on the x-axis. If *update mode* is also on, the currently-selected spectrum will be continuously changing, as indicated by the red line underneath the currently-updating spectrum.

2-D cursors

The rightmost four boxes are controls for the 2-D cursors; they are used for setting 2-D gates and windows, and have no purpose while in the 1-D display mode (note they are outlined in white, not red, hence are not available for use in the 1-D display program).

More on other options

Clicking on the *other options* box will bring up a secondary control panel. This will also cause the display program to enter an idle loop while waiting for you to make a selection from the secondary control panel. These selections are described below. If the display program is allowed to idle, more CPU time is available for the analysis process, or for other users of the system. The analysis process will automatically idle itself when it has no work to do.

update on / off

By default the program will continuously update all spectra as data are acquired. There are times when it is useful to turn off this updating; for example, if several spectra are simultaneously displayed (*view all* option) and you want to mark a gate or window in one spectrum at a time, the continuous updating should be turned off. This will keep the display program from moving from spectrum to spectrum, and will allow you to manually select the spectrum of interest using the *next* control box. Update mode is controlled by clicking the *update on* or *update off* box on the secondary control panel. Once a selection is made, the program will return to interactive mode, and the primary control panel will again be visible.

gates on / off

By default, *gating* is turned off. That is, even if you have defined gated spectra and have set the gates, the gating logic will not be activated until the *gates on* box is clicked. If a setup is saved or archived with gating mode on, then gating mode will automatically be set on when the setup is restored or retrieved. Note: gating is only applicable when analyzing event-mode data, either on-line or off-line. Gating is not relevant when viewing previously-saved histogram data. Gates

cannot be set when viewing previously-saved histogram data. When no gate is set (default) and gating is turned on, zero is a valid coincidence, i.e., all events will pass through the gate; whereas a gate set wide-open requires non-zero data for coincidence. Gating is turned off or on by clicking the mouse pointer inside the appropriate box.

view gates / list gates

The *view gates* option is used to display all gates which have been set on all spectra currently being viewed. Gates are displayed with the corresponding gate numbers drawn above the gated regions. If a large number of gates have been set, it may be difficult to check any one particular gate visually; the *list gates* option is used to list all gate locations to the system printing device. Note: printed output is *spooled* to an intermediate file, and is generally not printed until the file is closed, such as by ending the program. To force the spooled output to the printer, get into *keyboard options* and type **Dump**.

keyboard options

Many options and commands are available that utilize keyboard input. These options can be invoked by clicking on the *kybd options* box. This will cause the *Keyboard Options* window to pop up. As with all keyboard input windows, always first click-on the window with the leftmost mouse button to be certain that the keyboard is currently attached to that window. If the window is attached to the keyboard, the upper right corner of the window will show a high-lighted **KB** marker. The program will prompt **ENTER OPTION (? FOR LIST)**. Keyboard options are described in more detail in the *1-D Keyboard Options Table*.

zero one / zero all

This option is used to erase one or all 1-D spectra. This is useful in setup or testing, to erase displayed data without actually stopping and restarting a run. If *zero one* is selected, the currently selected spectrum will be zeroed; if *zero all* is selected, you will be asked if you really want to zero all spectra. Answer **Yes** or **No**. Note: only 1-D spectra will be zeroed using this option; any 2-D spectra will be unaffected, and must be zeroed from the 2-D display or by beginning a new run (or tape sort). By default, all spectra will be zeroed when a new run is begun or is sorted from tape (override with the NZR command).

select display

The *select display* option is used to select spectra to view simultaneously. Clicking this box will bring up the *Keyboard Options* window with the prompt, *Display which spectra?* Enter the desired numbers of the spectra to view (first click-on the window to be sure the keyboard is attached to it); use either explicit numbers, or one or more lists; e.g., **1,2,3,4,5,10,12,13,14,15** or, equivalently, type **1-5,10,12-15**. The selected subset of spectra can be viewed by clicking the *view all* box of the primary control panel. Note: once a subset of spectra has been selected, *view all* will only display the selected subset. To once again see all spectra, click-on *select display* and type **ALL** to the prompt *Display which spectra?*

virtual display

The *virtual display* option is used only in the 2-D display program for the creation and display of 2-D scatter plots (virtual displays). It has no function in the 1-D display program except to exit idle mode and to return to the primary control panel.

linear / log

The *lin / log* options are used to select a linear or logarithmic spectra display.

1-D Keyboard Options Table

CM	-	Compresses the current spectrum; use one of following: cm0 or cm1 cancels compression cm2 does a 2-channel compression cm3 does a 3-channel compression cm4 does a 4-channel compression cm5 does a 5-channel compression
CP	-	Change projection; permits the basic parameter being projected in the currently-selected gated spectrum to be changed.
DU	-	Dump; causes spooled output to be flushed to the printer.
EXE	-	Executes any DCL command and returns to user.
F	-	Folds channels together and redistributes the counts; use: f0 or f1 cancels the fold (default) f2 displays data as 2-channel sums f3 displays data as 3-channel sums f4 displays data as 4-channel sums
FI	-	Fix; replaces the original raw spectrum with the results of virtual transformations, e.g., minus background, plus another spectrum, etc. Note: The S Hi ft, M U lti pl y, P L us, and M I nus options are <i>virtual</i> transformations; the transformations are displayed, but the raw data are not touched. The F I x command is used to make these virtual transformations permanent.
LA	-	Changes the label of the current spectrum on the display only.
MAP	-	Prints channel contents of current spectrum; also dumps ascii to disk file.
MI	-	Minus; virtually subtracts the specified spectrum from the current spectrum
MU	-	Multiplies the current spectrum by a given factor
N	-	Normalizes the display to include negative points. 2nd entry resets option
PH	-	Print hits; list events passing selected free-form gates to FFGHITS.DAT.
PL	-	Plus; virtually adds specified spectrum to current spectrum Note: The spectra combined with M U lti pl y, P L us and M I nus must be the same size. Only the raw data in the specified spectrum will be combined with the current spectrum. However, multiple spectra can be combined; if spectrum #3 is added to #2, and #2 is added to #1, then spectrum #1 will contain the sum #1 + #2 + #3.
RU	-	Changes the current run # on the display only.
SAV	-	Saves all currently displayed spectra on disk or tape.
SC	-	Scale; permits the full-scale Y value to be manually selected.
SH n	-	Shift the current spectrum N channels. (negative # for left shift)
SM	-	Smooths the current spectrum; use one of following: sm0 cancels smoothing sm1 uses a 3-point function sm2 uses a 5-point function sm3 uses a 7-point function sm4 uses a 9-point function
TI	-	Changes the title of the current spectrum on the display only.
X	-	Exit; returns user to previous master menu, or to Tape Control menu

List of Fitting Options

- | | | |
|----|---|---|
| 1 | - | Return integrated peak and centroid (default) |
| 2 | - | Fit amplitude and position - width fixed |
| 3 | - | Fit amplitude and position and width |
| 4 | - | Fit amplitude and position width and tail |
| 5 | - | Fit amplitude and position - width and tail fixed |
| 6 | - | Fit two 3-parameter gaussians |
| 7 | - | Same as 6, except widths are equal and fixed |
| 8 | - | Fit with two 4-parameter gaussians |
| 9 | - | Fit with three 3-parameter gaussians |
| 10 | - | |
| 11 | - | Same as 9, except widths are equal and fixed |
| 12 | - | Fit with four 3-parameter gaussians |
| 13 | - | Same as 12, except widths are equal and fixed |
| | | |
| W | - | Enter FWHM of peak |
| Y | - | Enter exponential tail factor |
| P | - | Enter fitting parameters for peak |
| I | - | Identify; displays ID number for all fits to each peak currently displayed.
(a check indicates the fit is contained in the current fit spectrum. |
| D | - | Delete; entering Dn will remove fit # n from the fit spectrum but not from memory |
| A | - | Add a peak to fit spectrum. |
| E | - | Eliminate peak from memory. |
| Z | - | Zero fitted spectrum. |
| V | - | View; displays a fitted peak. |
| Q | - | Quick print; omits detailed listing of fit points; (2nd entry resets option) |
| S | - | Summarizes information on peaks in fit spectrum. |

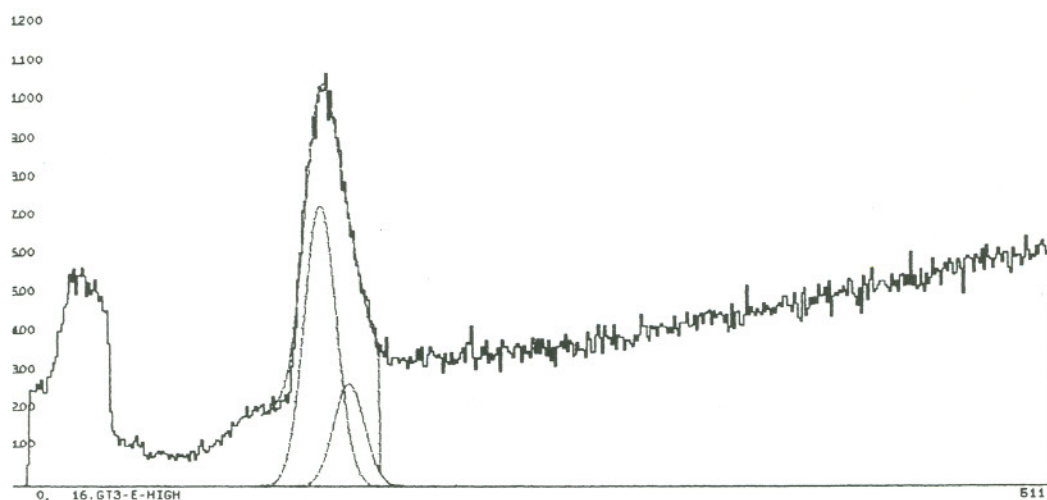


Figure 6. An example of multi-parameter peak fitting. Here the computed results are superimposed on the raw data, and the component peaks are displayed.

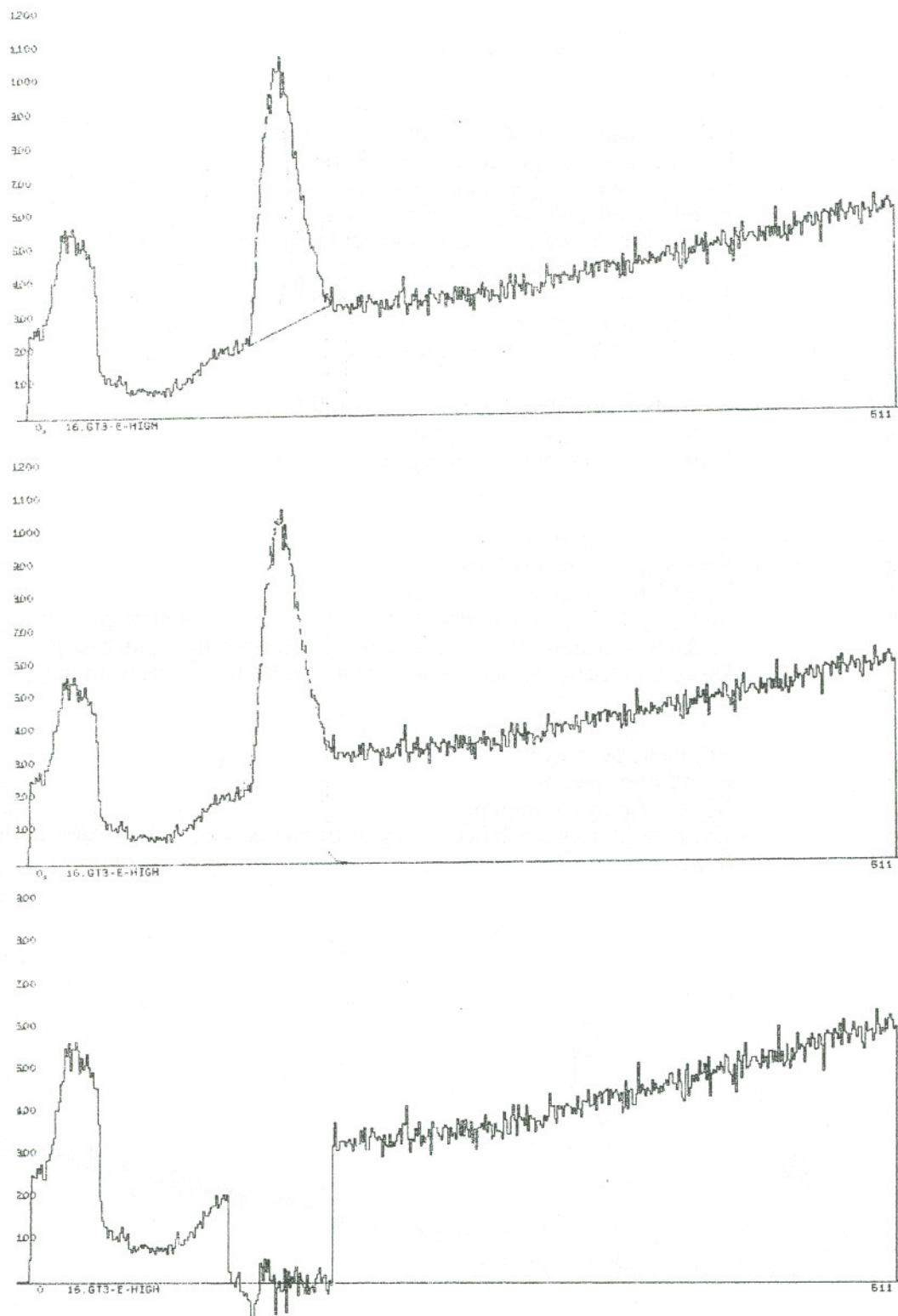


Figure 7. Multi-parameter peak fitting is one of several analysis tools available. Here, component gaussians are viewed individually (top), summed and superimposed on raw data (center), and stripped from the raw data (bottom). MicroStar can deconvolute up to four gaussians in a single peak.

Peak Fitting

There is currently a limitation in the program that prevents peak fitting in channels higher than 1023. If you must fit peaks in these higher channels, first use the SHift command to shift the higher channels into the 0-1023 range; after the fit has been performed add the shift offset to the computed centroids to determine the actual centroids.

To invoke gaussian peak fitting you must first give the program a reasonable starting value for the width (FWHM) of the peak(s) to be fitted. This can be done most easily by using *set gates* and *fit gate* to integrate a peak or region of the approximate width of a single peak. Alternatively, you can use fitting option *W*: from primary control panel, click *other options* then *keyboard option*, then type *W*. When prompted, enter the approximate Full Width at Half Maximum (FWHM) of the peak, in channels. This will determine a starting value for the multi-parameter peak fitting.

By default, invoking *fit gate* while *set gates* is active will yield the integral and centroid of the region defined by the cursors. This is option *1* of the fitting options. (See *List of Fitting Options* on previous page.) If you have already invoked other fitting options (2- 13) you must invoke fitting option *1* to be able to integrate peaks and regions.

Once a starting value has been established for the FWHM, peak fitting can be invoked by first selecting the appropriate fitting option from the *List of Fitting Options*. For example, to deconvolute two gaussian peaks in a single region, select option *6* (fit two 3-parameter gaussians--amplitude, position and width) by typing *6* to the prompt *Enter Option*. Then click-on the *set gates* option and position the cursors around the convoluted peak. If you want to subtract a linear background from the region, click-on the *view background* box. If you are satisfied with the selected region and background, click-on the *fit gate* box. For each peak, click-on the approximate centroid; a small asterisk (*) will appear above the data on the channel marked. If you are satisfied with this approximation, click *mark*. Do the same for the second peak. After the second peak has been *marked* the fitting procedure will begin. It may take several seconds for the fitting to complete. The fitting procedure can be canceled anytime by clicking off either the *fit gate* or the *set gate* box. When the fit is complete the computed fit-spectrum will be displayed superimposed in color on the raw data, and the calculated parameters will be displayed in the *Keyboard Options* window, and printed on the system printer. *Note: printed data are spooled to the disk before being printed. The actual printing usually does not occur until the program has terminated. To force spooled data to print, get into Keyboard Options and type DUMP.*

To subtract the fit spectrum from the raw data and display the residuals, click-on the *sub fits* box. To normalize the display of residuals to include negative points, get into *keyboard options* and type *N* in response to the *Enter Option* prompt.

Be sure to click-off all engaged options when peak fitting is complete.

Comments and Restrictions:

Peak fitting cannot be performed on data above channel 1023; if spectra larger than 1024-channels are defined, and peaks in the higher channels must be fit, use the SHift keyboard option to bring the high channels into the 0-1023 range.

Information on a maximum of **50 peaks** may be stored concurrently.

It is permissible to include both zero and negative points in a fit.

It is advisable to have at least **three more data points than free parameters**.

It is **not advisable to use fitting option 4 on peaks unresolved on the low-energy side.**

Peak position indicators must be moved after each fit.

The fitting of **multiple peaks is restricted to a maximum of 300 channels.**

There is no restriction on the size of individual peaks which may be fitted. However, some truncation of the initial display will occur if the peak is wider than 100 channels.

Setting Gating Conditions

Simple gates can be set only if gated spectra have been defined. See Chapter IV, *Data Analysis*, (page 11) for information on defining gated spectra. Free-form gates can be set anytime two dimensional spectra have been defined. Up to 64 standard gates can be defined, and each gate can be set on one, several, or all spectra. Additionally, 99 free-form gates can be defined; a free-form gate is a gate of any shape drawn on one or several 2-dimensional spectra. If a free-form gate is given the same gate number as a simple gate, the resulting spectrum will be projected from the *AND* of the standard and free-form gate. Since there can be 99 free-form gates and only 64 standard gates, gates numbered 65 through 99 are by definition stand-alone free-form gates.

Once gated spectra have been defined, the gating conditions can be set by first selecting the spectrum on which to set the gate. To define simple gates, click-on the *set gate* box; position the cursors to the desired locations using the cursor controls and the mouse pointer; when the gate is in the desired location, click-on the *mark* option. When prompted, enter the appropriate gate number or numbers. For logical *NOT* gates, precede the number with an *N*. If *NOT* gates and *AND* gates are defined on the same line, enter all the *AND* gates first. To delete a simple gate set on this spectrum, enter the negative gate number, e.g., -3. To completely delete a simple gate, you must delete it from every spectrum on which it is set. Gates can be set on one or several spectra, including pseudo-parameters and other gated spectra, as long as the gate definitions are logically consistent. Gates set on gated spectra will cause the corresponding component gates to be set. The program will prevent logically-inconsistent gate settings.

Free-form gating conditions are set in the 2-D display program by invoking the keyboard option *FG*. To set the free-form gate, position the mouse pointer to the desired location on the currently-selected 2-D spectrum and click with the left mouse button. If you are satisfied with this position, click *mark*. Otherwise reposition the pointer and click. Click *mark* to define the vertex point of the gate. Now move the pointer to another point to define another vertex of the gate, and click; a temporary line will be drawn between the vertices; if you are satisfied with this line segment, click *mark*; otherwise reposition the mouse pointer and click with the left button. Click *mark* to define the line segment. To optimize analysis, use as few vertices as necessary to define the gate. Continue until you have defined all but the last line segment of the free-form gate; to define the last line segment and force closure of the gate, click-on the *set gate* box. The software will query you for the gate number(s). Enter a single number or a list; do not enter an *N*; there are no *NOT* free-form gates. To view free-form gates, click-on the *view gates* option of the secondary control panel of the 2-D display program. To list gates to the system printing device, click *list gates*. To delete a free-form gate, enter keyboard option *DG* from the 2-D program; enter the gate number to delete a specific gate, or a -1 to delete all free-form gates. Note that a free-form gate can span several spectra. *Note: there is currently an unresolved problem in deleting free-form gates that span several spectra. If you must delete such a gate, currently you must delete all free-form gates and redraw them. To test the integrity of all defined free-form gates, list the setup by typing L to the sorting program's prompt Select type of Display Information. All free-form gates will be listed and tested for integrity.*

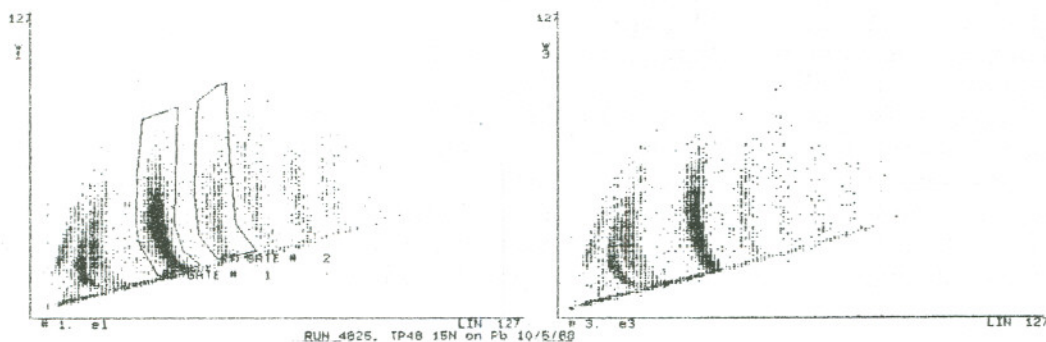


Figure 8. Free-form gates of arbitrary shape can be applied to one or many spectra to specify complex analysis conditions interactively.

Interactive 2-D Display

If you defined any 2-dimensional spectra during analysis setup, they can be viewed by clicking-on the *view 2-D* box. (Previously-saved 1 or 2-D histograms can be reviewed by selecting *View Tape Data* from the master menu, and selecting *RD (read)* from the list of choices). Up to sixteen 2-D spectra can be created in memory; an additional sixteen *virtual 2-D* displays (scatter plots) can be defined or redefined dynamically. Use of the 2-D display program is completely analogous to the 1-D program. Control panel functions and program options are described below.

set window / view window

In 2-D mode, the *set window* option is used to define regions to project from the x-domain and y-domain of the 2-D spectra. The low-energy x-cursor is selected and moved by clicking-on one of the two leftmost, upper boxes of the control panel (marked by <--- and --->). Additional cursor adjustments can then be made by clicking the mouse pointer on any desired channel. The low-energy y-cursors are controlled similarly by the leftmost lower two boxes of the control panel. The high-energy x-cursors are controlled by the rightmost upper two boxes of the control panel (marked <--- and --->). The high-energy y-cursors are selected and controlled by the lower rightmost two boxes. Once the desired x- and y-windows have been selected, click-off the *set window* option and click-on the *view window* option to display projections. Projections will be created and displayed dynamically as the data are analyzed. Click-off the *view window* option to return to the regular 2-D display.

set gates / fit gate

The *set gates* box is used to activate cursors for identifying channels, for integrating regions or for setting simple gates. Setting a simple gate in 2-D mode is equivalent to setting the same gates on the component 1-D spectra, and is accomplished in a similar manner: determine the gated region by manipulating the x- and y-cursors, and click *mark* when the gates are set to the desired position. When queried as to the gate number(s), enter either a number (e.g., 5), or a list of numbers (e.g., 5,6,7,12-20). For *NOT*-gates, precede the number or list with an *N* (e.g., N5-10). If *NOT*-gates and *AND*-gates are to be defined on the same line, enter all the *AND*-gates first. Entering a negative number will delete that gate from the current spectrum.

set background / view background

In 2-D mode, the *set background* option is used to set a threshold for the currently-selected spectrum. This threshold is simply a constant that will be subtracted from every cell of the 2-D display when the *view background* option is selected. Note this is a *virtual* subtraction; i.e., the actual data are not changed in memory; the background is subtracted only from the displayed data. Clicking-on the *set background* box will bring up the prompt Enter Z-axis Threshold; click the *Keyboard Options* window to be sure it is selected, then type in the appropriate number to be subtracted from the selected spectrum. To display the background-subtracted spectra, click-on the *view background* box. It will be necessary to force a screen-erase to display the background-subtracted data; do this by toggling between the *view one* and *view all* options.

mark

Mark is used in the 2-D mode only for marking the vertices of free-form gates. Free-form gates are arbitrarily shaped regions drawn on one or more 2-D spectra for gating purposes; see the section *Setting Gating Conditions*, page 36.

sub fits

The *sub fits* option is not used in 2-D mode.

other options

Clicking-on the *other options* box will bring up the secondary control panel, and will put the 2-D display program in idle mode; options of the secondary control panel are discussed immediately following this section.

view 1-D / view 2-D

This option is used to switch between the 1-D display program and the 2-D program.

view one / view all

These options are used to display either a single 2-D spectrum or all defined 2-D spectra. Currently it is not possible to display selected subsets of all spectra. When viewing all 2-D spectra simultaneously, the currently-selected spectrum is indicated by a colored bar along the y-axis.

scale up / scale down

These boxes are used to adjust the z-scale up or down for the currently-selected spectrum.

next

The *next* option is used for selecting and displaying either the next higher-numbered (--->) or lower-numbered (<---) spectrum.

more on other options

Clicking-on the *other options* box will bring up the secondary control panel, and the display program will relinquish CPU time until you have made a selection. The functions of these other options are completely analogous to their usage in the 1-D display program, except that *select display* has no function except to return you to the primary control panel; additionally, the *virtual display* is available in the 2-D display program.

virtual display

A *virtual display* consists of one to sixteen additional 2-D displays that can be created and changed dynamically by the user without going back to setup mode. These displays are *scatter plots* -- they are drawn on the screen but they are not stored in memory. No z-scale information is available, and virtual displays cannot be integrated or background-subtracted.

To create or view virtual displays, click-on the *virtual display* box. If you have already defined virtual displays, or if they were defined in any setup that you restored (RS) or retrieved (RT), the *Virtual Display* screen will pop up, and, if data are being acquired or analyzed, you will see the spectra begin to take form. If you have not yet defined any virtual 2-D's, you will be asked how many virtual 2-D's you want to create (16 maximum); then you will be asked to specify 3 numbers for each virtual 2-D display: the x-parameter #, the y-parameter #, and a gate # if you want the spectrum to be projected via gating conditions. If you do not want the spectrum to be gated, specify a zero for the gate number. When you have specified the three numbers for all defined spectra, the *Virtual Display* screen will pop up and the spectra will be drawn. While in *virtual display* mode, notice that the *next* control panel option has been replaced with an *erase* option. Clicking-on the *erase* box will erase all accumulated virtual displays. To escape virtual display mode and return to the regular 2-D display, click-on *other options*. If you want to archive the virtual display with the rest of the analysis setup, stop any data taking or analysis and type SET to the ENTER COMMAND prompt; then you can either archive the setup (AR command) or just type N to cause all current setup conditions to be written to the file LAST_ORDER_SETUP.SET.

If you want to redefine the virtual displays, click-on *other options* then click *keyboard options* and, when prompted with ENTER OPTION, type SV (setup virtuals). You will then be asked to define the virtual displays, as described above.

2-D Keyboard Options Table

CON	-	Create contour map superimposed on current spectrum.
CUT	-	Cutoff; allows control of color contour regions for 2-D displays. On color systems, z-intensity is represented by 8 colors. By default, these color levels are autoscaled to the data (cutoff=0); specifying a non-zero number will determine a region through which the eight colors will be distributed. By default, z-intensity is also represented by a spiral within each pixel for compatibility with monochrome systems. On color systems this can cause some undesirable interference patterns when viewing some data. On these systems, the cutoff value should be specified as a negative number to suppress the drawing of the spiral pixels.
CXY	-	Changes the X & Y parameters associated with the current display; subsequent data sorted will display new parameters.
DG	-	Delete free-form gate #; -1 deletes all free-form gates
FG	-	Setup free-form gates on current spectrum
GRD	-	Overlay current 2-D display with a grid
MAP	-	Channel printout of current 2-D display
SAV	-	Save all currently displayed spectra on disk.
SV	-	Setup virtual displays (up to 16 max)

The following commands are still available, but have been superseded by mouse options:

L	-	Linear / log display toggle
U	-	Update mode on/off toggle
VG	-	View gates; displays all standard & free-form gates.
LG	-	List gates to printing device (spooled)
GAT	-	Gating mode on / off toggle
VIR	-	Virtual display; draw scatter plots
Z #	-	Zero spectrum. ZA to zero all 2-D spectra. (1-D not affected)



Figure 9. Virtual displays (scatter plots) can be created 'on the fly' to monitor the experiment. In this example, several charge states are clearly visible.

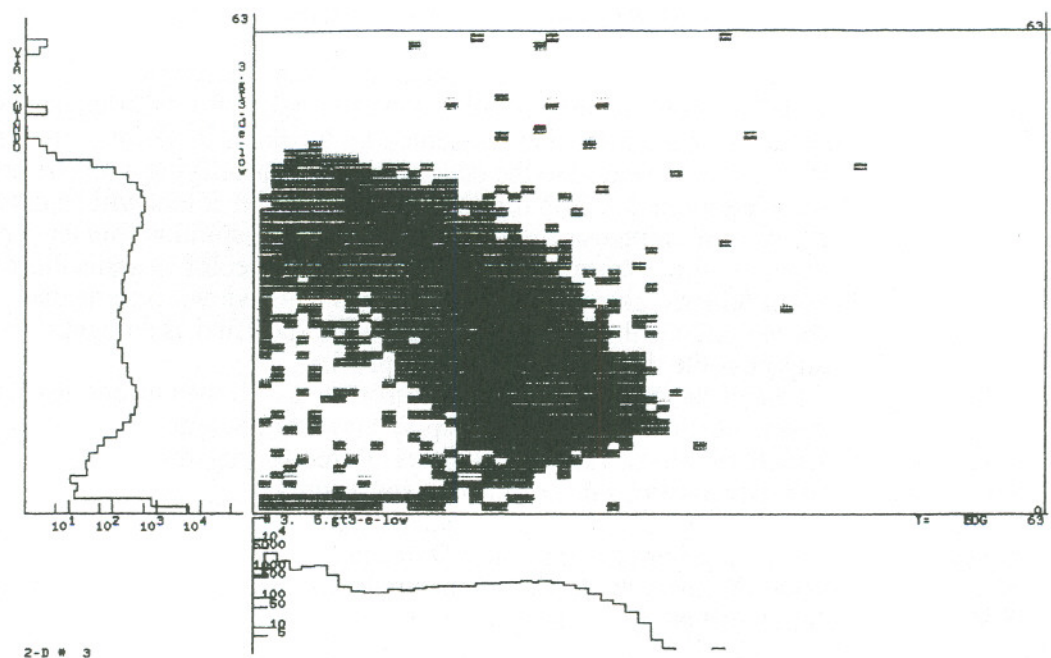
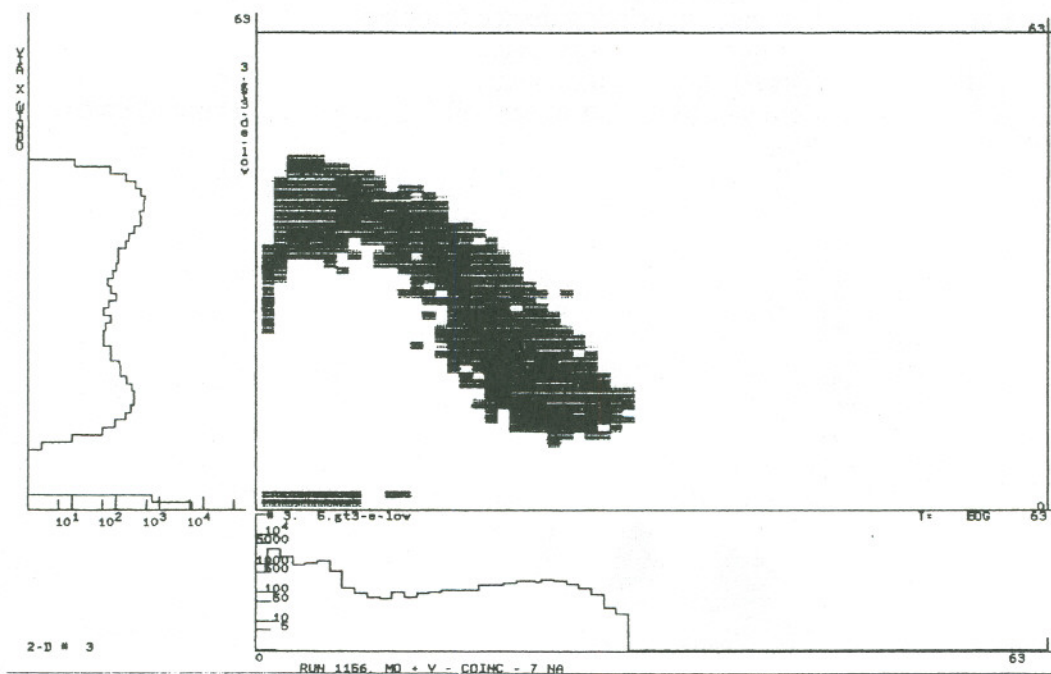


Figure 10. (Top) A 2-D spectrum with its X- and Y-parameters projected (by engaging the *view window* option while in 2-D mode). Below, the same spectrum projected after interactive background subtraction.



VI. USING STARBURST

Starburst is an auxiliary CAMAC crate controller manufactured by Creative Electronic Systems (CES) of Geneva, Switzerland. This device includes a DEC PDP-11/73 computer on a chip (the J11), and operates in conjunction with a standard A-2 CAMAC crate controller as a real-time computer and a CAMAC interface. When connected to a Q-bus and appropriate devices, the *Starburst* can run any PDP-11 operating system; DEC's RT-11 operating system is utilized for its speed and simplicity.

The *Starburst* resides in slot 23 of the first CAMAC crate, immediately next to the A-2 crate controller. The Auxiliary Control Bus (ACB) cable must be connected from the rear of the *Starburst* to the rear of the A-2 controller. Similarly, short LEMO cables must be connected from REQUEST to GRANT IN on the crate controller, and from GRANT OUT on the crate controller to GRANT IN on the *Starburst*. The Q-bus ribbon cables P3 and P4 should be connected, red side up, to connectors J3 and J4 on the front of the *Starburst*. These cables should be routed through the back of the CAMAC crate and connected to the corresponding connectors on the paddle board in the first slot of the Q-bus expansion box. The *Starburst* operator's terminal should be connected to the small I/O connector on the front of the *Starburst*. (Optionally, one can use a window in the VAXstation running KERMIT in place of a regular terminal.)

If the A-2 crate controller does not have a CAMAC Branch Highway cable connected to it, the A-2 controller must be switched OFF-LINE for the *Starburst* to address the CAMAC dataway.

Next, make sure the connections between the microVAX and the expansion box of the *Starburst* are intact. If DRV11-WA interfaces are used, make sure the ribbon cable is connected such that J1 on the *Starburst* end is connected to J2 on the microVAX end, and J2 on the *Starburst* end is connected to J1 on the microVAX. If a DCQ-1000 fiber optic interface is used, make sure the fiber optic cable marked Rx is connected to receptacle Rx, and Tx to Tx. Additionally, make sure the RS232 connection is intact; the DLV-11 or equivalent in the *Starburst*'s expansion box must be connected to device TTA1: on the microVAX; usually, a null-modem is required on one end of this serial line.

If additional CAMAC crates are to be connected, the software currently requires that additional crates be interfaced through the CC-LSI or CC-Qbus controllers, manufactured by DSP Technologies, Inc., of Fremont, California, USA. If additional crates are used, they should be set to base address 762000 (crate 2), 770000 (crate 3) and 774000 (crate 4). All attached crates, even if not used, must be powered-on for the system to run.

Since *Starburst* resides in the CAMAC crate, and the CAMAC crate(s) must be powered off while adding or removing modules during the experimental setup, starting up the uSTAR software in the *Starburst* is usually one of the last things you do before actually taking test data. Any time the power is cycled on the CAMAC crate the *Starburst* must be rebooted. Once the *Starburst* has been started up, the user-specified event readout code must be downloaded from the VAXstation. This is accomplished either by starting up the *MicroStar* software in the VAXstation, or issuing the CP (Change Parameters) command to the already-running acquisition program.

To bootstrap the *Starburst*, first turn on the rack's master power switch, if any. Make sure power is applied to the CAMAC crate(s), the disk drive, and the Q-bus expansion box. Make sure the disk drive is spun-up and ready. The console terminal or window should be alive with an ampersand (@) prompt. Engage CAPS LOCK on the console to set upper case characters. (The bootstrap PROM program will not recognize lower-case letters)

Type **173000G**

The system should respond **BOOT>**

Type in the name of the system device, using only capital letters, e.g.,

DLA0

Note that no carriage returns are necessary or desirable on the above two commands. Once the system is booted, all commands **must** be terminated with a carriage return. That's all there is to it. The system will automatically startup the uSTAR software, and the message *Starburst waiting for Host* should appear on the console.

Anytime the power is cycled on the CAMAC crate, the *Starburst* must be rebooted, and then must be downloaded with the appropriate readout commands.

LSB LANGUAGE DESCRIPTION

Using LSB

LSB is a language used to specify operations to perform at every event trigger. *LSB* gives the user the flexibility to perform selective event readout, custom formatting, etc. If more than one CAMAC crate is used, currently *LSB* must be used to program the multi-crate readout. Similarly, if the 2-interrupt capability of the *Starburst* is desired, *LSB* provides the tools needed to specify the machine instructions to execute at each interrupt service routine. The *LSB* language is described below. Following that are sample *LSB* programs from actual runs at LBL.

Starburst commands are assembler-like operators which are read from a user-specified file. These commands are translated into PDP-11 machine code and downloaded to the *Starburst*. Commands are free-form; either commas or blanks can be used as separators. Do not use TABs. Commands which are the destination for branch or jump commands must contain a label. The part of each line following the command may be used for comments. Lines which begin with * are also treated as comments.

An *LSB* readout program in general will read data from CAMAC and move some or all the data into an output buffer to be passed to the microVAX. By user-option at the time of downloading the *Starburst*, the front-end may or may not zero-suppress all data in the buffer before transmission to the microVAX. If the data are to be zero-suppressed (default) the user's front-end program must put all selected parameters in the corresponding location in this buffer; the user must explicitly SKIP locations in the buffer not to contain any data (one could STORE zeroes into these locations, but the buffer is passed to the users' interrupt routine pre-zeroed, so SKIPPING is much faster). In zero-suppress mode, parameter N must be in the Nth word of the buffer; all zeroes in the buffer will be removed during event formatting. If the user specifies NO zero-skip (option User) then the buffer will be shipped intact to the microVAX with no formatting.

A label is a string consisting of a colon followed by 1 to 4 characters. Labels must begin in column 1. **END** is a predefined label used to exit from the interrupt.

In the examples below, constants are shown as **CONST**. Constant values are normally decimal numbers. To designate a hexadecimal constant, the value must be preceded by a # symbol. The term **VAR** means a variable value which is defined by a labeled **WORD** command. **ADDRESS** is a decimal or hexadecimal constant denoting an absolute address in the *Starburst* memory.

WARNING: The command translator remembers the last function code used in each crate, and loads the function registers only when necessary. This is desirable because 3 extra memory cycles are used each time a function register is loaded. When a program branch or jump occurs, it is possible to reach a section of code with the wrong value in a function register. To avoid this problem, the user must carefully analyze all function code usage and insert **FUNCTION** commands where necessary.

Commands recognized by the Starburst translator:

READ	C,N,A,F	Read dataway into event buffer
GET	C,N,A,F	Read dataway to Accumulator
WRITE	C,N,A,F	Write Accumulator to dataway
PUT		Put Accumulator into event buffer
PUT	CONST	Put CONST into event buffer
SKIP	CONST	Skip CONST words in event (Same as PUTting zeroes, but faster)
LOAD	CONSTVAR	Load CONST or VAR into Accumulator
ADD	CONSTVAR	Add CONST or VAR to Accumulator
SUB	CONSTVAR	Subtract CONST or VAR from Accumulator
STO	VAR	Store Accumulator in variable
LDM	ADDRESS	Load Accumulator from ADDRESS
STM	ADDRESS	Store Accumulator at ADDRESS
TST	ADDRESS	Test value at ADDRESS
BIT	CONSTVAR	And CONST or VAR with Accumulator
CMP	CONSTVAR	Compare Accumulator with CONST or VAR
BEQ	LABEL	Branch to LABEL if result .eq. 0
BNE	LABEL	Branch to LABEL if result .ne. 0
BHI	LABEL	Branch to LABEL if result .gt. 0
BMI	LABEL	Branch to LABEL if result .lt. 0
BPL	LABEL	Branch to LABEL if result .gt. 0
JUMP	LABEL	Jump to LABEL
:NAME	WORD	Define a variable or constant
	FUNCTION	CRATE Force reload of function code in CRATE

SAMPLE STARBURST PROGRAMS

The keywords **\$S1** and **\$S2** identify the beginning of each of the code sequences to be loaded into the Starburst interrupt routines. The numbers following the keywords specify the address of the interrupt code in the Starburst and must correspond to certain locations in the version of the Starburst program which is being used.

If an interrupt level is unused, no program is needed for that level.

INTERRUPT LEVEL 1 PROGRAM

```

$S1,297,2297      Address of 1st Interrupt Service Routine in Starburst
*
*   This program will save a 6-word event only when a flag bit is
*   set and the value of the *   first parameter is greater than 100.
*
      GET      1,2,0,0      READ BIT REGISTER IN SLOT 2, SUBADDR 0
      BIT      #0001      TEST BIT 0
      BEQ      BAD        BRANCH IF BIT 0 IS ZERO
      STO      TEMP        SAVE VALUE FOR LATER
      GET      1,1,1,0      READ FIRST PARAM FROM SLOT 1, SUBADDR 1
      CMP      100        COMPARE PARAMETER TO CONSTANT= 100
      BHI      GOOD        BRANCH IF PARAMETER VALUE.GT. 100
:BAD  WRITE    1,1,7,11    RESET THE ADC
      JUMP     END        'END' IS PREDEFINED LABEL
:TEMP WORD     0          BIT REGISTER VALUE IS SAVED HERE
*
:GOOD PUT
      READ     1,1,2,0      READ 2ND PARAMETER
      READ     1,1,3,0      READ 3RD PARAMETER
      WRITE    1,1,7,11    RESET THE ADC
*   Here bit 14 of the bit register is checked to see if we want to
*   read parameters 4, 5,6
      LOAD     TEMP        GET BIT REGISTER VALUE AGAIN
      BIT      #0002      TEST BIT 1
      BNE      G2         BRANCH IF BIT 1 NOT ZERO
      SKIP     3          PUT 3 ZEROES IN EVENT
      JUMP     DONE
:G2  READ     1,1,4,0      GET PARAMETER 4
      READ     1,1,5,0      GET PARAMETER 5
      READ     1,1,6,0      GET PARAMETER 6
:DONE WRITE    1,1,7,11    RESET ADC
$ (Any line beginning with $ indicates end of interrupt program)

```

INTERRUPT LEVEL 2 PROGRAM

```

$S2,2358,4358      Define address of interrupt routine in Starburst
*
*   This program resets one module whenever a bit is found set in
*   another module.
*
      GET      1,2,1,0      READ BIT REGISTER IN SLOT 2, SUBADDR 1
      BIT      #0004      TEST BIT 2
      BEQ      END        DO NOTHING IF NOT SET
      WRITE    2,17,0,9    ELSE RESET MODULE 17 IN CRATE 2

```


Front-end code to read and clear an 8-channel ADC

```
$S1 297 2297
  READ 1 12 0 2
  READ 1 12 1 2
  READ 1 12 2 2
  READ 1 12 3 2
  READ 1 12 4 2
  READ 1 12 5 2
  READ 1 12 6 2
  READ 1 12 7 2
$END OF FILE
```

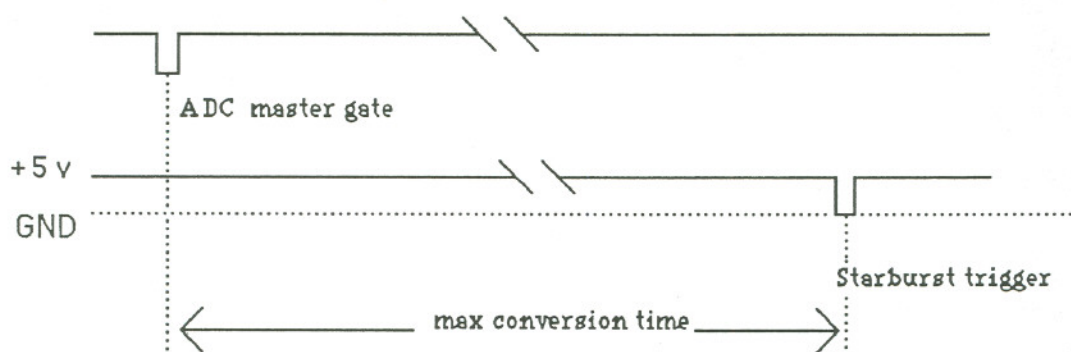
Front-end code to read data and suppress pedestals

```
$S1 297 2297
  LOAD 2                                set computer busy latch
  WRITE 1 9 0 17
  GET 1 1 0 0
  CMP 15
  BHI N1
  SKIP 1
  JUMP N1X
:N1  PUT
:N1X GET 1 1 1 0
  CMP 15
  BHI N2
  SKIP 1
  JUMP N2X
:N2  PUT
  . . . et cetera
:N31 PUT
:N31X GET 1 4 7 0
  CMP 15
  BHI N32
  SKIP 1
  JUMP N32X
:N32 PUT
:N32X READ 1 6 0 0
  READ 1 6 1 0
  READ 1 7 1 0
  READ 1 7 2 0
  WRITE 1 6 0 9
  WRITE 1 7 0 9
  WRITE 1 19 0 9
  WRITE 1 19 1 9
  LOAD 4
  WRITE 1 9 0 17
  LOAD 8
  WRITE 1 9 0 17                reset computer-busy latch
$END OF FILE
```

ACQUISTION HARDWARE SETUP

The event readout is triggered by one (or optionally two) front panel interrupts provided by the user. Each of these interrupts consists of an inverse-TTL signal, normally HIGH (2.8- 5 volts), going momentarily to ground to trigger the readout. The width of this signal is not critical, but should be in the range of 500ns - 5 μ s. A Gate & Delay Generator used in conjunction with a Level Adapter can be used to supply the signal. This interrupt should be timed such that all CAMAC modules have had time to convert before the trigger. For a typical AD811 ADC, the conversion time is 80 microseconds, thus if an event consisted only of parameters from this ADC, the readout trigger should be delayed 80 μ secs after the ADC strobe. But if an AD811 is used with, say, a LeCroy 2228A TDC (conversion time up to 104 μ secs) the *Starburst* interrupt should be extended to 104 microseconds. If a mixture of modules is used, the delay time should, in general, be the longest conversion time of any of the modules.

There is a latency of 18-20 microseconds from the time the interrupt is applied until the time the users' interrupt program is entered. Where high data rates are essential, one can subtract this latency time from the interrupt delay time to optimize throughput. Similarly, there is a 5-10 μ sec delay from the time the last instruction in the users's readout code is executed until the front end is ready to accept another event. Thus if hardware blocking logic is to be employed, the blocking should begin with the master gate and end 10 μ secs after the end of the users' readout code. The end of the users' readout code can be marked by including a CAMAC write to an output register as the last operation performed.



VII. MONITORING AND CONTROL

Two additional utilities are available for general scaler monitoring and recording, and for the setup and monitoring of LeCroy 4032(A) high voltage power supplies through the LeCroy 2132 CAMAC interface. These programs can be run through a separate login.

High voltage monitoring and control

To run the high-voltage monitoring and control program, define a symbol to execute the program:

HVC ::= "RUN BEV_SHARE:HV_CONTROL"

Then type **HVC** to execute the program.

The program will prompt

Enter CAMAC slot # for 2132

Enter the appropriate slot for the LeCroy 2132; note there is a minor software restriction that the 2132 reside in Crate # 1.

The software will then attempt to initialize the 2132, emptying its FIFO. If error messages appear at this point, check that the Starburst is up and running the USTAR program, and that the RS232 connection between the Starburst and the microVAX is intact..

The software will then prompt

HV CONTROL enter **CLEAR**, **READ**, **ON**, **OFF**, **EDIT**, **CHANGE**, **MONITOR**, **EXIT**

All desired high-voltage setting are stored on a file called HVSETUP.DAT. This file simply contains the desired voltage for a given channel, one entry per line. To protect some hardware, the software will not allow you to set a voltage higher than 1500 volts. (This can be changed by editing HV_CONTROL.FOR and changing the *DATA MAXVOLT /1500/* to the appropriate value.

The commands are self-explanatory, but are elaborated on below.

CLear -	Initializes the high-voltage system; clears out 2132's FIFO.
REad -	Reads the 4032(A) and lists and prints desired settings and actual readings.
ON -	Turns on the high-voltage supplies, setting the voltages specified in the file HVSETUP.DAT, set by the user in the EDit mode.
Off -	Turn off the high-voltage supplies.
EDit -	Edit or create the file HVSETUP.DAT. This file contains one entry per line, an integer specifying the desired setting for the corresponding HV channel.
CHange	Change the setting for a single channel.
MONitor	Monitor the high-voltage system; the software will ask for a monitor interval in minutes, and an acceptable deviation from desired settings. At every interval the voltage reading will be dumped to the screen, and a beeping will sound if any channels are found to deviate too far from desired settings.
EXit -	To terminate the program. Note: once in MONitor mode the program must be stopped with a <control>-C.

Scaler monitoring

If scaler monitoring and recording are required, define a symbol to execute the SCALERS program:

SCALERS ::= RUN BEV_SHARE:SCALERS

Then to run the program, type **SCALERS**. The program will respond

SCALERS> Enter **EDit**, **ECldump**, **DUp**, **CLear**, **ZErocl**, **Auto**, **EXit**

The program is set up to read both LeCroy ECL scalers and regular scalers. The location of the scalers is specified in a file SCALER_LOCS.DAT. This file can be created or changed by the **EDit**

command. This file should contain, on each line, the Crate, Number, Address and Function code for each scaler. A brief description of the commands for the SCALER program follows:

EDit	-	Spawns default editor for creating or modifying scaler location file.
ECldump		Performs special readout of ECL scalers.
DUmp		Read and print all defined scalers, showing date & time.
CLear		Clears all defined scalers (except ECL scalers).
ZEROecl		Zeroes ECL scalers.
AUto		Automatically reads and dumps scalers periodically. The program will ask for a delta time (in minutes) between scaler dumps.
EXit		Terminate the program. Note that you must abort with a <control>-C or <control>-Y when in AUto mode.

APPENDIX 1

Tape and Data Formats

Event data consist of one or several 4096 word (8192 byte) header records containing information such as run number, data format, time and date, run title, and parameter labels, followed by several 8192-byte data records. The data records consist of events which may be of variable length and usually span tape records. The events are generally separated from each other by an *event separator*, which is always FFFF hexadecimal (integer -1), unless one selects user-formatted data. The last buffer of a run is padded with zeroes after the last acquired data. The structure of the header record for event type data is shown below. If more than 155 parameters are acquired, additional header records will be written to accomodate the additional labels. There will be one additional record for every 162 additional parameters/event beyond 155. The number of *additional* header records is contained in word 20 of the first header. Versions of MicroStar before 3.0 write only one header record.

WORD	DESCRIPTION
1	The run number.
2	The number of parameters per event
3	Month (1-12)
4	Day (1-31)
5	Year
6	Hour (0-23)
7	Minute (0-59)
8	Second (0-59)
9	Number of words per record in the subsequent data.
10-13	0 for event data.
14-18	RESERVED
19	Event data format: NOTE: acquired data always 5,6 or 7
	0 => old LBL multiplexer data
	1 => old LBL 16-channel multiplexer zero-skip data.
	2 => old LBL chained-multiplexer zero-skip data.
	3 => old LBL CAMAC data, non zero-skip.
	4 => old LBL CAMAC high-resolution data
	5 => Starburst zero-skip CAMAC data.
	6 => HERA zero-skip CAMAC data
	7 => USER formatted MicroStar data
20	Number of additional header records
21-99	Unused, should be all 0s.
100	Start of run title (16 words or 32 char max).
.	.
200	Start of 1st parameter label (12 words or 24 char max).
.	.
225	Start of 2nd parameter label (12 words or 24 char max).
.	.
250	Start of 3rd parameter label (12 words or 24 char max).
.	.
275	Start of 4th parameter label (12 words or 24 char max).
.	.
.	Etc.

STARBURST ZERO-SKIP CAMAC FORMAT

This is the default format for data acquired by MicroStar. All event-separators, hit-maps, and data are 16 bit words. An event-separator, consisting of all bits equal to 1 (FFFF hexadecimal), follows each event. Consecutive parameters within an event are divided into groups of 8, and each group is preceded by a sub-event hit-map. The most significant bit (bit 15) of each sub-event hit-map is set to distinguish it from the data words. Bits 8-14 of each hit-map are encoded with a sub-event number from 0 to 127. This allows the discrete identification of up to 1024 parameter/event data although the MicroStar software can handle a maximum of 512 parameters. The low order byte, bits 0-7, indicates which event parameters are present. Each of the 8 bits corresponds to one parameter; if only parameters 3 and 8 were non-zero, the hit-map would be '10000100'. Finally, non-zero data words are presented exactly as they are acquired from CAMAC modules (15 bits are available for data). A sample of data from this format looks like:

Word	Comment
1000 0000 0000 0111	1st sub-event, parameters 1,2,and 3
0xxx xxxx xxxx xxxx	parameter 1
0xxx xxxx xxxx xxxx	parameter 2
0xxx xxxx xxxx xxxx	parameter 3
1000 0010 0010 1100	3rd sub-event, parameters 3,4,and 6
0xxx xxxx xxxx xxxx	parameter 3 (event parameter 19)
0xxx xxxx xxxx xxxx	parameter 4 (event parameter 20)
0xxx xxxx xxxx xxxx	parameter 6 (event parameter 22)
1111 1111 1111 1111	event-separator
end of event #1	
1000 0111 1100 0000	8th sub-event, parameters 7 and 8
0xxx xxxx xxxx xxxx	parameter 7 (event parameter 63)
0xxx xxxx xxxx xxxx	parameter 8 (event parameter 64)
1111 1111 1111 1111	event separator
end of event #2	

STARBURST NON-ZERO-SKIP CAMAC FORMAT

This event format simply collects all data words readout from CAMAC, appends an event separator (hex FFFF), and passes the data to the microVAX. For 20 parameters per event data, it would look like:

Route	Data	Comments
xxxx xxxx xxxx xxxx		1st data word
xxxx xxxx xxxx xxxx		2nd data word
....
xxxx xxxx xxxx xxxx		20th data word
1111 1111 1111 1111		event separator
end of event #1		
xxxx xxxx xxxx xxxx		1st data word
....
xxxx xxxx xxxx xxxx		10th data word
....
xxxx xxxx xxxx xxxx		20th data word
1111 1111 1111 1111		event separator
end of event #2...		

HERA HARDWARE ZERO-SKIP FORMAT

CAMAC zero-skipping also can be done in hardware using 8-channel multiplexers and Selenia ADCs, as currently done by the HERA group at LBL. Events are separated by an event separator having all bits set (FFFF hexadecimal). Events are broken up into sub-events of 8 parameters with each group preceded by a sub-event header word. This header word has bit 15 set, with the rest of the bits of the most significant byte all zero. Bits 4-7 contain the multiplexer number (16 maximum) and bits 0-3 contain the number of data words plus one (for the header word). To distinguish the individual parameters in a sub-event, the 3 most significant bits of each data word comprise a route code (13 bits are available for data). For example, this format would look like:

Word	Comment
1111 1111 1111 1111	event separator
1000 0000 0001 0011	sub-event header, MUX #1, 2 data words
000x xxxx xxxx xxxx	parameter 1
100x xxxx xxxx xxxx	parameter 5
1000 0000 0101 0010	sub-event header, MUX #5, 1 data word
111x xxxx xxxx xxxx	parameter 8
end of event #1	
1111 1111 1111 1111	event-separator
1000 0000 0010 0101	sub-event header, MUX #2, 4 data words
001x xxxx xxxx xxxx	parameter 2
010x xxxx xxxx xxxx	parameter 3
100x xxxx xxxx xxxx	parameter 5
111x xxxx xxxx xxxx	parameter 8
end of event #2...	

The following commands may be used to copy an event tape on drive 'mua0' to a disk file named 'data.dat':

```
$ MOUNT/FOREIGN/BLOCKSIZE=8192 MUA0:
$ COPY MUA0: DATA.DAT
```

The file will create an unformatted, variable record length file with 8192 byte (4096 words) long records. To read this file from a Fortran-77 file would require the following open statement:

```
OPEN (UNIT=3, FILE='DATA.DAT', FORM='UNFORMATTED',
2 STATUS='OLD', RECORDTYPE='VARIABLE', RECL=2048)
```

The RECORDTYPE='VARIABLE' part of the open statement is very important. Segmented record types are the default for unformatted files in VAX Fortran-77 and trying to read a variable record length file as a segmented file will not work. For unformatted files, the record length RECL is number of long words. Then if there was an integer*2 array of 4096 words called 'BUFFER', the first unformatted read would return the header record and subsequent unformatted reads would return the data records (or additional header records followed by the data records)

```
READ( 3 ) BUFFER          ! gets the header
IHEARRECS = BUFFER(20)    ! # of additional header records
DO I=1, IHEADRECS
    READ( 3 )              ! skip over the header records
END DO
READ( 3, END=999 ) BUFFER ! the first buffer of data
.....
ETC.
```

1-D HISTOGRAM TAPE FORMAT

One or more 1-D spectra may be recorded in a single file on disk (or optionally on tape). The data are written sequentially in blocks of 4000 words/record, (8000 bytes) without breaks between spectra. The last record will be padded with zeroes as required. The header record contains the necessary information to extract the individual spectra. The spectra within a file will be 32 bits/channel; the spectra need not be the same length. At this time spectra can not exceed 4096 channels in length.

1-D HISTOGRAM HEADER STRUCTURE

WORD	DESCRIPTION
1	The run number.
2	0 for histogram data.
3	Month (1-12) ^
4	Day (1-31) ~
5	Year ~
6	Hour (0-23) ~
7	Minute (0-59) ~
8	Second (0-59) -
9	Number of words per record in the subsequent data (always 4000 for uSTAR histograms).
10	number of histograms in the file (32 max).
11	total number of data records to follow header.
12	reserved
13	Type of data: always 12 for uSTAR 1-D histograms
14-17	Unused, should be all 0s.
18-19	total number of 16 bit words written
20	Spec #1: number of channels.
21	" start channel
22	" number of words/channel (always 2 for uSTAR 1-D histograms)
23	Spec #2: number of channels.
24	" start channel
25	" number of words/channel
26	Spec #3: number of channels.
27	" start channel
28	" number of words/channel
.	.
.	Etc.
.	.
180	Start of General title (32 characters max)
200	Start of 1st spectrum label (12 words or 24 char max).
.	.
.	.
225	Start of 2nd spectrum label (12 words or 24 char max).
.	.
.	.
250	Start of 3rd spectrum label (12 words or 24 char max).
.	.
.	Etc.
.	.

2-D HISTOGRAM HEADER STRUCTURE

One or more 2-D histograms may be recorded in a single file on tape. The 2-D histograms are written sequentially, column by column, in blocks of 4000 words/record, without breaks between spectra. The last record will be padded with zeroes as required. The header record contains the necessary information to extract the individual spectra. The data is written as one word per channel, with all 16 bits used (64k counts per channel). Counts between 32k and 64k may be interpreted as negative numbers on some computers since FORTRAN does not usually have unsigned 16 bit integers as a data type.

WORD	DESCRIPTION
1	The run number.
2	0 for histogram data.
3	Month (1-12) ^
4	Day (1-31) ~
5	Year ~
6	Hour (0-23) ~
7	Minute (0-59) ~
8	Second (0-59) ~
9	Number of words per record in the subsequent data (4000 for histograms).
10	number of histograms in the file (16 max).
11	total number of data records to follow header.
12	RESERVED
13	Type of data: always 14 for uSTAR 2-D histograms
14-17	Unused, should be all 0s.
18-19	total number of 16 bit words written.
20	Spec #1: parameter number on X-axis
21	" parameter number on Y axis (
22	" number of channels X-axis.
23	" number of channels Y-axis.
24	" start channel on X-axis
25	" start channel on Y-axis
26	" location in array (=1 for spec #1).
27	" unused.
28	Spec #2: parameter number on X-axis.
29	" parameter number on Y axis.
30	" number of channels X-axis.
31	" number of channels Y-axis.
32	" start channel on X-axis
33	" start channel on Y-axis
34	" location in array
35	" unused.
.	.
.	Etc.
.	.
180	Start of general title (32 characters max)
.	.
200	Start of 1st spectrum or parameter label (12 words or 24 char max)
.	.
225	Start of 2nd spectrum or parameter label (12 words or 24 char max)
.	.
250	Start of 3rd spectrum or parameter label (12 words or 24 char max)
.	.
.	Etc.

APPENDIX 2

Software Installation and System Notes

The *MicroStar* software is supplied as a VMS saveset on magnetic tape. This saveset contains all executable programs, source modules, libraries, and all procedures needed to install, re-install, run, or link all the programs.

To install the software on your system, first log into the SYSTEM account. Create a directory to contain the saveset, and set the default directory protections to allow world read/execute privileges:

```
CREATE / DIR sys$sysdevice:[USTAR] /PRO=(S:RWE,G:RWE,W:RE,O:RWE)
```

Then restore the saveset into the new directory:

```
BACKUP / REWIND tape_name:USTAR sys$sysdevice:[USTAR]*.*
```

When the restore operation is complete, dismount the distribution tape and archive it in a safe place.

Because these programs require various privileges to operate, they are *installed* as privileged system images so that they may be run from unprivileged user accounts. This is accomplished by the INSTALL_USTAR procedure. This procedure must be executed each time the system is rebooted. Copy the INSTALL_USTAR.COM procedure into the system manager's directory:

```
COPY sys$sysdevice:[USTAR]INSTALL_USTAR.COM SYS$MANAGER:*
```

Copy the shared image file SHR_COM.EXE to the system shared image directory:

```
COPY sys$sysdevice:[USTAR]SHR_COM.EXE sys$share:*
```

and set the protections on the shared file to allow world read/write access:

```
SET FILE / PRO=(s:rwed,o:rwe,g:rwe,w:rwe) sys$share:SHR_COM.EXE
```

Modify the system startup procedure sys\$manager:SYSTARTUP_V5.COM or equivalent to install the software at startup time, by adding the lines

```
$ ASSSIGN / SYS / EXEC sys$sysdevice:[USTAR] BEV_SHARE  
$ @sys$manager:INSTALL_USTAR
```

Also, the default VMS values for the UIS windowing system's memory pools must be expanded to accomodate the large MicroStar windows. If these memory pools have not already been expanded, do so by modifying the system startup procedure sys\$manager:SYSTARTUP_V5.COM or equivalent; immediately before the line:

```
$ @STARTVWS
```

add the lines

```
$ DEFINE / SYS / EXEC UIS$QVSS POOL SIZE 1593200  
$ DEFINE / SYS / EXEC UIS$P1_POOL_SIZE 512000
```

NOTE: the memory pool sizes defined above are appropriate for an 8-plane color VAXstation; if you are installing the software on a monochrome workstation, consult the VAXstation Software Installation Guide for appropriate values. Allocate enough memory for 25 windows, even though that many will never be created. If the memory pools are too small, the computer may crash after several hours of operation.

Next, define a symbol to execute the programs (note: only one user can execute the uSTAR programs at a time; the RUN_CHAOS procedure prevents multiple users from attempting to run the program at the same time.) Add the following definition to the sys\$manager:SYLOGIN.COM file, or otherwise define the symbol for all users:

\$ USTAR ::= @BEV_SHARE:RUN_CHAOS

Once all the system startup files are appropriately modified, the system must be rebooted for the installation to be complete.

NOTE: These programs can accept data from a variety of sources, including CAMAC-based Starburst systems, data tapes, or network VME acquisition. If a network connection to a remote VME data acquisition system is desired, the logical name *VME_DATA_SOURCE* must be defined (relevant only to LBL's VME<=>VAX data acquisition system). This can be done by the user prior to invoking the software.

DEFINE VME_DATA_SOURCE NET-BEVAX1-DAQS-JVN

Privileges needed to run the data acquisition system are:

- Process CHAOS needs exquota,pswapm,log_io,altpri,mount,share,readall
- Process ORDER needs exquota,pswapm,share
- Process MULTID needs exquota,pswapm,share,mount,readall
- Process SPACE needs exquota,pswapm,share,mount,readall
- Process CACQ needs exquota,pswapm,log_io,altpri,share
- Process CTRL needs exquota,share
- Process LSB needs logio,share,readall

If you need the high-voltage monitoring and control program utilizing the LeCroy 4032 (a) CAMAC control subsystem, you will also need to install HV_CONTROL as a privileged system image:

Process HV_CONTROL needs logio,share,readall

If you need the scaler monitoring program, it also needs to be installed with privileges:

Process SCALERS needs logio,share,readall

Note that the shared image SHR_COM.EXE must be installed /WRITE/SHARE in the directory SYSS\$SHARE: (which is usually also SYSS\$LIBRARY)

It may be necessary to expand system resources and working set defaults to attain satisfactory operation. Suggested values for a VAXstation 3200 system running VMS v5.1 are:

PAGEDYN = 4,800,000, NPAGEDYN = 1,500,000, PQL_WSDEFAULT=300,
PQL_WSQUOTA=3200, PQL_WSEXTENT=4096, ASTLM=500, MAXDETACH=12

These values can be changed through SYSGEN or through the AUTOGEN procedure.

Make sure the User Authorization File (UAF) allows all potential users sufficient resources to run the programs. Suggested UAF parameters are: PRCLM=10, FILLM=50, BIOLM=40, DIOLM=40, TQELM=40, ENQLM=200, BYTLM=30000, JTQUOTA=2048, WSDEF=256, WSQUOTA=3200, WSEXTENT=4096, PGFLQUO=100000