NUREG/CR-1668
RR

Advanced Mobile Multi-Processor Gamma-Ray Acquisition and Analysis System

Manuscript Completed: July 1980
Date Published: September 1980

Prepared by
E. W. Killian, R. A. Morneau

EG&G Idaho, Inc.
Idaho National Engineering Laboratory
Idaho Falls, ID 83415

Prepared for
Division of Safeguards, Fuel Cycle and Environmental Research
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, D.C. 20555
NRC FIN No. A6075

ABSTRACT

This report describes a new Gamma-Ray Acquisition and Analysis system which has been developed for the In-Plant Source Term Measurement Program. A new computer was added to the system described previously in Reference 5, "Procedures, Source Term Measurement Program," TREE-1178. One computer is now used to acquire the data and the other (new computer) is used to analyze the resulting data. The throughput of the system has been dramatically improved. Data analysis times have been reduced by about a factor of 10. Moreover, the analysis procedure is much more complex and provides results which can be directly reported with minimal operator interpretation. The information contained in this report supersedes the description of the analysis package given in Appendix B of the procedures manual (NUREG-0384).

CONTENTS

CONTENTS (cont'd)

FIGURES

TABLES

PREFACE

This report describes a new Gamma-Ray Acquisition and Analysis System which has been developed for the In-Plant Source Term Measurement Program. This work is being performed by the Idaho National Engineering Laboratory for the Office of Nuclear Regulatory Research in support of requirements of the Effluent Treatment Systems Branch of the Office of Nuclear Reactor Regulation.

The In-Plant Source Term Measurement Program is a joint effort of the Physics Division of EG&G Idaho, Inc., and the Exxon Nuclear Idaho Company, Inc. The primary objective of the program is to study the behavior of radionuclide transport in commercial operating pressurized water reactors (PWR's) and provide the NRC with data that can be used in the evaluation of plant designs for liquid and gaseous waste treatment systems. This evaluation requires a knowledge of the sources and quantities of radioactive waste materials generated at a nuclear power reactor during normal operation including anticipated operational occurrences, during shutdown and refueling, and during normal restart operations. The study is designed to measure how sources change with plant design, radionuclide inventory in various components, transports through the system and performance of the radioactive waste treatment systems.

The measurement program requires that samples of process fluids, both liquid and gaseous be taken for measurement. Two types of samples are taken, liquids from the liquid stream and sampling cartridges from the gaseous (ventilation) streams. The liquid samples are taken in 50 ml or 450 ml glass bottles and acidified with concentrated hydrochloric acid (approximately 2 ml acid per 100 ml sample) to minimize plateout and settling problems. The gaseous streams are sampled for isotopes of iodine (mainly ^{131}I) using a special sampling system that will selectively strip a stream of particulate iodine, elemental iodine (I_2), hypoidous acid (HOI), and organic iodine (e.g., CH_3I). The collection media are mounted in series and air from the ventilation duct pulled through the media, and the various iodine species are collected. Following the collection of samples they are analyzed on the advanced mobile multi-processor gamma-ray acquisition and analysis system.

The advanced mobile multi-process gamma-ray acquisition and analysis system is housed in a mobile laboratory which is moved to a commercial reactor plant for the duration of a measurement period. The gamma-ray system consists of two lithium drifted germanium [Ge(Li)] crystals, preamplifiers and associated electronics. The Ge(Li) crystal gamma-ray detectors are computer controlled for the collection of spectra by the NOVA-1200 computer. A panel is provided for input of system variables. Following the collection of the spectra, it is transferred to an Eclipse S230 computer for spectra analysis using the advanced analysis computer program. This program searches the spectrum for peaks, integrates the

peaks using a Gaussian fitting faction, tags the peaks with possible radionuclide identification, corrects for background radiation, performs interference decontaminations for known interfering gamma rays, and resolves the results for multiple gamma rays for a single radionuclide to provide a single weighted average. Parent-daughter corrections are made as appropriate. The program then prints a summary table of the standard radionuclides and a second table of found non-standard radionuclides. The results are listed as microcuries/milliliter. In the case of a standard radionuclide that is not found to be present, a detection limit is listed.

ACKNOWLEDGEMENTS

The authors wish to thank the following people for their contributions in the design and successful completion of this spectrometer and gamma-ray analysis system.

J. W. Mandler - Source Term project manager

R. L. Kynaston - Digital electronics

J. K. Hartwell - Data analysis algorithms

L. O. Johnson - Analog electronics

R. A. Coates - Digital electronics

D. R. Underwood - Electronics technician

We would also like to thank R. G. Helmer for his help with the data analysis algorithms, W. R. Myers for his managerial support, and all of the Source Term Project people whose actual hands-on experience with the system lead to many helpful suggestions.

I. SOURCE TERM GAMMA-RAY SPECTROMETER SYSTEM

The source term gamma-ray spectrometer system has been developed for the Nuclear Regulatory Commission (NRC) in support of the Source Term Project. The system is installed in the NRC Mobile Laboratory (a fifth-wheel trailer) which is moved to various reactor sites where gamma-ray pulse-height measurements are made. The system has been designed to give the operators complete gamma-ray spectrometry capability, from data collection to analysis and report generation. The major features of the system are:

- High-rate data collection capabilities. Up to 100,000 counts/second without peak-shape distortion.
- Automatic energy calibration and system monitoring capabilities using a precision pulse generator.
- Unattended data collection using a computer controlled sample changer.
- Timely and detailed data analysis.
- Automated sample logging and bookkeeping.

Figure 1 is a block diagram of the complete spectrometer system. The system consists of detectors and front-end electronics, a NOVA-1200 data acquisition computer, an ECLIPSE data reduction computer and a sample changer.

I.1 NOVA-1200 COMPUTER SYSTEM

The NOVA-1200 computer system consists of the following major components:

- A NOVA-1200 with 32 K words of core memory.
- An in-house fabricated control panel used to control the collection of data.
- A console teletype.
- A dual-drive floppy disk for program storage, data archives and sample information.
- A computer-controlled sample changer.
- A CRT display monitor.
- A serial communications link with the Eclipse computer.
- ADC and precision pulse generator interface.

The NOVA-1200 computer system can collect two simultaneous spectra of 4096 channels with a maximum of 16,777,215 (2^{24}-1) counts per channel.

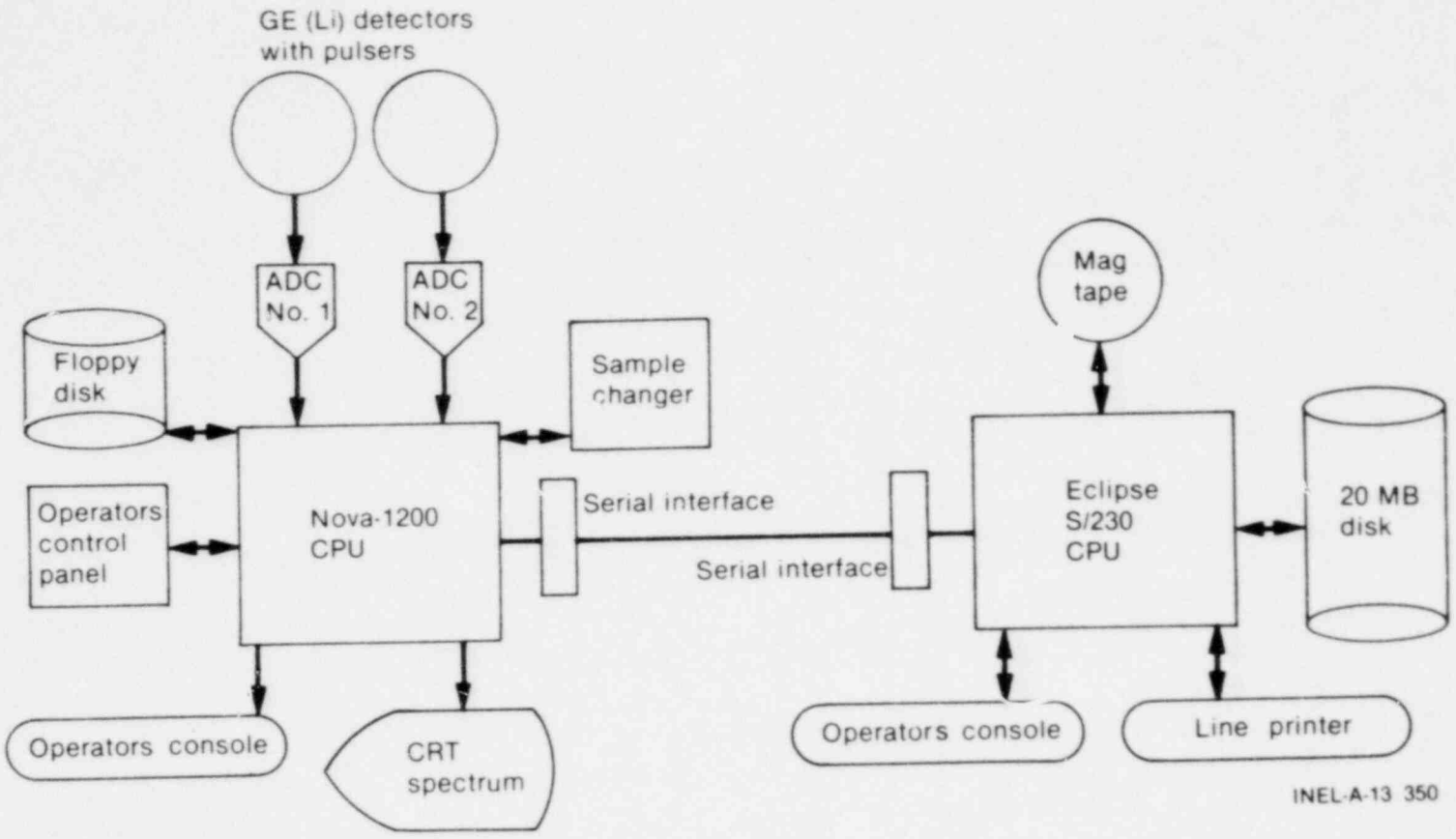


Figure 1. Source Term Trailer Gamma-Ray Spectrometer System

Data is accumulated via a direct memory increment interface. Time to store one event is 1.8 to 2.4 micro-seconds. The system has a two amplitude precision pulse generator as part of each data interface. The pulser is used to monitor operational characteristics of the analog front end such as system noise, gain and zero shifts. The pulse generator operates from a constant current source at a fixed rate, providing two fixed amplitude pulses at the input to the preamplifier of the detector. At the same time the pulse is injected, the NOVA-1200 is notified that the next event to be received is from the pulser. This allows pulser events to be stored independently from the normal gamma-ray data. Pulser data is stored in a window of 17 channels; one window for each of the two amplitudes. The data contained in these windows are then placed in the collected data spectrum when the count is finished. The ECLIPSE computer uses this pulser data to perform an energy calibration and validate the quality of the gamma-ray spectrum.

The NOVA-1200 software is written in assembly language and communicates with the user via a control panel and an operator's console (teletype). The fast front-end electronics are described in Section I.2.

Appendix D contains a summary of the commands available on the NOVA-1200. Appendix E contains the program loading and startup instructions.

In the ensuing sections, the following notation conventions have been followed. The " " symbol is used to indicate operation of the CTRL key on the operator's console. The CTRL key is used with another key and operates similarly to the SHIFT on an ordinary typewriter. In all examples which follow, user input is underlined. If a response to a question or command is terminated with a RETURN key, the symbol "↵" is used. If in the text an ambiguity exists between a numeric 0 and the letter "O", the numeric zero will be slashed (e.g., Ø).

I.1.1 CONTROL PANEL

Figure 2 is a picture of the control panel which was fabricated in-house to provide a good man-machine interface between the computer and the operator. Its primary function is control of the spectrometers. As can be seen from the figure, the panel is divided into three parts. Two of the parts are identical and each controls the operation of one spectrometer. The third part displays information about the gamma-ray spectrum. The spectrum from which the data are being displayed can come from either spectrometer and is selected by pushing an appropriate key on the panel. Spectral data are displayed on the CRT and can be rotated, windowed and displayed in a log or linear format. The display is controlled from the set of keys located in the center of the panel. All of the information being displayed is kept current by frequent computer update.

Besides controlling the spectrometer and displaying vital information, the panel is used to input to the computer information about the sample currently being counted. This information includes: (1) the sample number, (2) counting distance, (3) efficiency table to use for this sample, and (4) the spectral disk slot to use when transfering the data to the floppy disk.

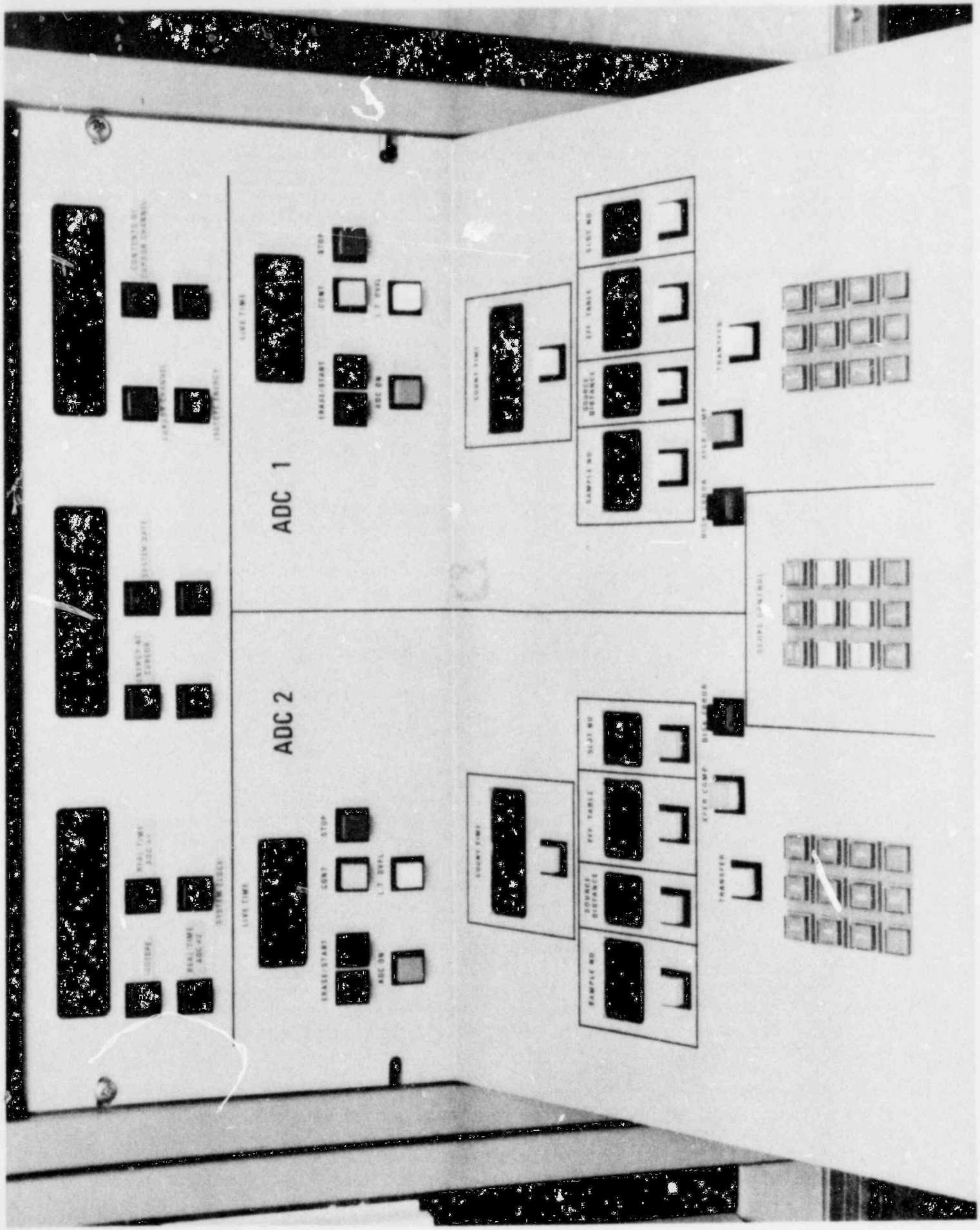


Figure 2. NOVA-1200 Control Panel

The computer reads this information for the panel and places the values for these parameters in appropriate channels of the spectrum. They then become a permanent part of the data. The sample number which was entered on the panel is the sample number of the radioactive sample from which the data is being acquired. This number is used to retrieve from the disk further information about the sample (see Section I.1.2).

When the transfer button is pushed, the data and its associated counting information, pulser data and spectrum header are transferred to the disk and then to the ECLIPSE.

I.1.2 SAMPLE LOG

When data is transfered from core to disk (and sequentially to the ECLIPSE), information which describes the sample together with the counting information is written into the first 18 channels of the spectrum. The information which describes the sample is extracted from the Sample Log. This information must have been previously entered by the user and includes:

1. Sample number
2. Sample alpha I.D.
3. Sample clock time (time at start of sample collection)
4. Sample date (date of sample collection)
5. Sample time (sample collection duration)
6. Volume of sample (normally in milliliters)

As each sample is counted and the data transfered to the disk an entry is automatically made in the sample log for that sample and includes:

1. The run number of spectrum
2. Time at which the count was started
3. The date when the count was started
4. The count duration (live time)
5. The distance at which the sample was counted
6. The efficiency table number

Up to eight entries can be made for each sample.

The sample log has room for 999 samples and has been organized into 7 groups. The samples assigned to each group are given in the following table.

SAMPLE	GROUP
1 - 19	1
20 - 99	2
100 - 199	3
200 - 299	4
300 - 499	5
500 - 699	6
700 - 999	7

TABLE - Sample Group Assignment

The following example shows how a sample is entered into the sample log.

EXAMPLE - Entry into sample log

```
*E
(SAMPLE,S.CLK,S.DATE,S.TIME,VOL)
>613,1425,041577,100H,100
THIS IS A SAMPLE HEADER
>↑D
*
```

The first entry made by the user is the sample number which must be an integer N such that $1 \leq N \leq 999$. Subsequent entries, S.CLK,S.DATE,S.TIME are separated by commas. S.TIME has units appended to it. Valid units are: S for seconds; M for minutes; H for hours and D for days.

After the sample numeric information has been entered, the alpha I.D. is entered. This I.D. can be any length up to 1 line (72 characters). It is terminated by a carriage return.

When the alpha I.D. is entered, the computer writes the data to disk and responds with a caret(>). Information for another sample can now be entered.

To terminate the entry of sample information, a CONTRL D (↑D) is typed. The ↑D is typed immediately after the caret as indicated in the example.

Information contained in the sample log can be printed in three ways. The first utilizes the SAM command. This command prints the contents of one sample entry. Input to the command is sample number:

EXAMPLE - Print one sample entry

```
*SAM
>613
**613.  1425.  41577.  100.H   100.
THIS IS A SAMPLE HEADER
*
```

It is also possible to print the sample information contained in one or more groups using the U command. Input to this command is one or more group numbers. For multiple printing of groups, entries must be separated by commas. The entries do not have to be in particular order. There can, however, be no more than 7 entries. All samples contained in the log can also be printed by typing a carriage return, for the sample group number:

EXAMPLE - Print sample groups

```
*U
* ENTER ALL GROUPS TO BE PRINTED
>6
**613.  1425.  41579.  100.H  100.
PRIMARY WATER UNIT #2 COLD LEG
 3.  1630.  41579.  1000.S.  30.  415.
34.  1100.  42579.  2000.S.  10.  420.
```

In the above example, there was only one entry in the 6th group of the log, and the sample had been counted twice. The sample information follows the double '*' and is the sample number, sample clock time, sample date, sample time and the volume of sample respectively. The sample alpha ID follows on the next line. The following two lines describe how and when the sample was counted. For example, the first count was given run number 3. It was counted at 1630 hours on the 15th of April 1979. The live-time count duration was 1000 seconds and was counted at 30 cm. Efficiency Table 415 was used for this count.

The last command which can be used to print sample information is the ULOG command. This command is used to print information from those samples whose sample date is greater than a user supplied cut-off date:

EXAMPLE - Print sample information with cut-off date

```
*ULOG
ENTER CUT-OFF DATE:  041079
ENTER ALL GROUPS TO BE PRINTED
>6
**613.  1425.  41576.  100.H  100.
THIS IS A SAMPLE HEADER
```

The cut-off date is entered as month-day-year. In the above example, the user wished to print all information about the samples in group 6 which were acquired after the 10th of April, 1979. In this example, the sample had not been counted. If it had been counted that information would have been printed.

I.1.3 PROGRAMMABLE SAMPLE CHANGER

The Programmable Sample Changer (PSC) is a device which allows the computer to automatically select a sample from a possible ten positions and bring the selected sample into a counting position (distance from detector). The software in the NOVA allows the user to program the sample to be counted, time for each count and position at which the count is to be made and the efficiency table to use. After each sample is counted, a header for the data is automatically constructed and is transfered, together with the data, to the ECLIPSE computer.

All samples to be counted are selected from the sample log. The sample number in the sample log is used to determine the sample to be counted. The first step in programming the PSC is to make entries in the sample log for those samples to be counted (using the "E" command). Once all entries are made in the sample log, a count queue is constructed. The count queue describes the order in which the entries in the sample log are to be counted, the distance and time for each count and the efficiency table number. The queue also has a provision for specifying a timed interval between counts. The format for the count queue is:

SAMPLE NUMBER, COUNT POSITION, COUNT DISTANCE, COUNT TIME, EFF. TABLE NUMBER

The "Q" command is used to construct a count queue:

EXAMPLE - Construct a count queue

```
*Q
(SMP,POS,DIS,CT,EF)
>1, 2, 30, 500, 32
>10, 4, 19, 400, 33
>D100
>7, 10, 15, 215, 35
>↑D
*
```

In the above example, samples in positions 2, 4, 10 are to be counted. The "D100" causes the computer to wait 100 seconds between the counting of sample 10 and 7. There are no restrictions on the order in which samples may be counted. The total number of entries allowed in the count queue is 20.

To begin the data acquisition from the count queue, the "CFQ" command is issued. This command causes the samples to be counted, a title record to be constructed, counting information to be placed in the data and the data to be written to the ECLIPSE.

I.1.4 ECLIPSE-NOVA COMMUNICATION

The spectral data transmitted to the ECLIPSE is divided into two parts: (1) Spectral Alphanumeric ID header, and (2) the spectral

data which includes the counting and sample times. The data is transfered in a serial form at 9600 bits/second. It takes approximately 15 seconds to transfer both parts of the spectral data.

The Spectral Alphanumeric data is a block of 256 16-bit words which is used to describe the sample from which the data was collected. In the normal counting sequence, the information is constructed automatically from the sample log information. All information in the ID header is in 6-bit ASCII code formated into lines with a maximum of 80 characters per line. Each line is separated by a carriage return character. The first line in the spectrum ID record is a numeric spectrum number (NSN). The NSN is used by the ECLIPSE software as the file identifier for each spectrum received. That is, when a spectrum is received by the ECLIPSE, a file is created whose name is NSN. All further references to this spectral data is by the NSN. The NSN should therefore be unique to the data being collected. It has the format of a 8-digit number, YYMMDDSS. The YYMMDD part of the NSN is the date which the data collection process began. YY is the year, MM the month and DD the day. This date is constructed when the spectrum accumulation process begins and does not relate to the date at which the counting stops. The SS part of the NSN is the spectrum run number. The SS value starts at a value supplied by the user during the initialization of the software in the NOVA-1200. A unique starting value for SS is given for each detector system. The value for each SS is incremented by 1 each time data from a detector system is written to the disk and sent to the ECLIPSE. When the day parameter changes, the values for SS are reset to their initial starting values. The second line in the spectrum aphanumeric ID is the mnemonic "=500D". This mnemonic is a flag which indicates that additional counting information is contained in the beginning 58 channels of the spectrum. Following these lines is an alphanumeric header which describes the sample. This information is a duplicate of the alpha ID for the sample as it was constructed in the sample log.

The second part of the information which is transmitted to the ECLIPSE is the pulse height spectrum. The spectrum contains further information about the sample being counted in the first 58 channels. The information passed in these channels is given in the following table:

CHANNEL NO.	CONTENTS
0	Live-time of data accumulation
1	Real-time of data accumulation (elapsed time)
2	Sample number
3	Sample type; 0 for normal spectrum; 1 for a thorium calibration spectrum

CHANNEL NO.	CONTENTS
4	Time at start of sample collection
5	Date at start of sample collection
6	Sample collection duration
7	Sample collection duration units; 0=seconds; 1=minutes, 2=hours, 3=days
8	Time at start of count
9	Date at start of count
10-11	Analyzer name; 7-bit ASCII code, two characters per channel
12-13	Detector name; same format as analyzer name
14	Detector number
15	Counting distance in mm
16	Volume of sample (normally in milliliters)
17	Efficiency table number
18	Linearity table number
19-20	The width equation coefficients (a,b), where W(X)=a+bX; X is the channel number. Channel 19 contains a*1000, and $b*10^6$ is in Channel 20.
21	Low amplitude pulser energy equivalent in EV
22	High amplitude pulser energy equivalent in EV
23	Beginning channel of low amplitude pulser data window
24	Beginning channel of high amplitude pulser data window
25	Number of pulser counts which did not fit in a 17 channel window (i.e., number of pulser rejects)
26-41	Low amplitude pulser data
42-58	High amplitude pulser data

The rest of the channels (59-4095) contain the pulse height spectrum.

Channels 0 - 4095 are sent to the ECLIPSE in blocks of 128 channels with 3 bytes/channel in binary format. Each block is transmitted with a three-byte check sum. If the check sum value computed by the ECLIPSE does not match the transmitted value, the ECLIPSE will instruct the NOVA-1200 to retransmit the data block.

The following steps are performed by the NOVA when communicating with the ECLIPSE.

1. Send a "receive spectrum" command to the ECLIPSE.

2. Wait up to 20 seconds for a response from the ECLIPSE. If no response is received within the required time, a meassage "ASYNC LINE ERROR" is printed on the NOVA operator console.

3. Send spectrum ID alphanumeric header.

4. Wait up to 20 seconds for a response as in Step 2.

5. Send a spectrum data block.

6. Wait up to 20 seconds for a response, as in Step 2.

7. Check response for an indication to retransmit block. If the block must be re-sent, go to Step 5. Proceed until all data blocks have been sent.

A spectrum is sent to the ECLIPSE under two conditions: (1) the transfer button is pressed on the control panel or (2) the command TE1 or TE2 is given on the operator console. The TE1 command transmits the data from spectrometer #1. The TE2 transmits the data from spectrometer #2. The TE1 and TE2 commands do not place counting information in the first 58 channels of the spectrum, but send the spectrum alphanumeric ID and pulse height data as it exists when the command was given. The transfer button on the control panel will construct the counting information from the sample log and then write the data to disk and then to the ECLIPSE.

I.1.5 ENERGY CALIBRATION

The precision pulser supplies energy calibration information about the spectrum being collected in real time. It must be calibrated from time-to-time, however, to insure that its absolute pulse amplitude has not changed. The calibration of the pulser amounts to determining the energy equivalents of the pulses being injected into the system. The method used

to perform this function is to collect a ^{228}Th spectrum and derive an energy scale, then, superimpose this energy scale on the pulser data which was collected during the collection of the ^{228}Th spectrum. This process is done automatically by the user typing the command CLBT1 or CLBT2 after a good thorium spectrum has been collected in a normal manner. The CLBT1 command is used when performing a pulser calibration on detector #1 and CLBT2 when calibrating system #2. The software begins the calibration process by computing the centroids of 5 gamma-ray lines of known energy in the thorium spectrum. These centroids and energies are used to perform a linear least-squares fit to obtain the coefficients of a quadratic energy equation of the form $E=a+bx+cx^2$ where x is the channel number and E the energy. The lines used in this calculation are at 238.624, 583.174, 860.53, 1620.7, and 2614.476 keV. The peak centroids are calculated using a non-linear least-squares fit of the data to a gaussian function. This energy scale is then used to determine the equivalent energies of the two pulser peaks. The centroids of the pulser peaks are computed using a moments method.

The calibration procedure prints the energy equivalents on the console. The operator then has the option to change the current equivalent values if he desires to do so.

CHANGING PULSER EQUIVALENTS

The pulser energy equivalents are kept on the disk and are changed by the CPQ1 command for detector #1 and CPQ2 for detector #2. This command will first print the current low amplitude energy equivalent, then wait for the user to input a new value. The process is then repeated for the high amplitude pulse:

EXAMPLE - Change pulser equivalents

```
*CPQ1
260.458 261.365 ↵
2605.37 2606.421 ↵
*
```

After the new values have been entered, the software updates these values on the disk. The user can also use the CPQ1 or CPQ2 tc inspect current values for the pulser equivalents. If a carriage return is typed without entering a new value, no change is made in the current values.

EXAMPLE - Inspect pulser equivalents

```
*CPQ1
260.458 ↵
2605.37 ↵
*
```

I.1.6 DISK FORMAT AND USAGE

The floppy disk on the NOVA-1200 is a dual drive system with a capacity of 300,000 16-bit words per disk. The two drives have been logically divided into a system drive and a data drive. The data drive is used exclusively to store spectral data. It is the drive which is used to record the accumulated spectrum when the transfer button is pressed on the control panel. The system drive is used to store information which must be accessed on a regular basis. The kinds of information stored on this disk are the program binary loader, linearity tables, sample log, isotope library and efficiency tables.

The critical nature of the information contained on the system drive requires that a current backup of that disk be kept at all times. This is particularly important if many new entries have been made to the sample log. The FORMAT command is used to create a backup copy of the system disk. This command formats a disk located in the data disk drive and copies the system disk block-by-block to the newly formatted disk. The process requires about 5 minutes to complete and destroys any information which might have been on the data disk previously. The use of this command is straightforward. The user is prompted for the steps he must take. This command is also used to format a new disk for data storage. The system information which is recorded on the new disk will be overwritten when spectral data is written to that disk. A newly formatted disk cannot be used as a system disk backup after spectral data has been written on it.

Spectral data recording on the data disk is accomplished by writing the data to specific "slots" on the disk. These slots are really specific disk blocks which are used as starting points for the recording of the spectral data. There are 48 spectral data slots on each floppy. This method of recording spectral data allows random spectral data access without requiring the use of an index block. Spectra which have been recorded on the floppy disk are accessed by slot numbers. Commands which read or write spectral data require the input of a slot number. For example, the R1 command is used to read data from the disk into the memory space used by detector #1 for data storage:

EXAMPLE - Read data from disk

```
*R1
SLOT:  13
*
```

In the above example, data was read from slot 13. A similar command R2 is used to read data into the memory space used by detector #2. The 01, 02 commands are used to write data from detector #1 and #2, respectively to disk. The user is again required to input a slot number. No clock or

sample information is placed in the spectrum when data is written to disk using the 01 or 02 command. Data is recorded as it exists when the command is executed.

I.1.7 NOVA-1200 DATA ANALYSIS

The NOVA-1200 software has minimal data analysis capability in that it can automatically locate peaks in the spectrum, perform a non-linear least squares fit to a gaussian and make an isotopic identification of the analyzed line. The analysis of a spectrum is performed on the spectral data located in the computer memory. The command 'F1' is used for detector #1 and 'F2' for detector #2. The following example illustrates the use of the F1 command. The command first prints the first 20 channels of the spectrum, then the efficiency table number and title and finally the analysis results.

EXAMPLE - NOVA-1200 DATA ANALYSIS

*F1

```
     2000.     2570.       .         .         .         .         .          9.
        2.        2.      17.       182.      451.      1152.     3290.      6286.
     9884.    13583.   16172.    17862.
```

430 EFF. TABLE POINT SOURCE

	CHANNEL	ENERGY	FWHM	AREA	PERR	GAMMA/SEC	ISOTOPE
1	337.9	238.63	1.5	2793846.	11.8	.1396923E+04	THD228.
1	341.2	240.94	1.6	268774.	3.4	.1343870E+03	THD228.

I.2 FRONT-END ELECTRONICS

The front-end electronics (see Figure 3) for the source term Ge(Li) spectrometer systems includes a Princeton GammaTEC (PGT) right-angle coaxial detector and preamplifier, a Nuclear Diodes (ND) integral dewar detector with preamp, and dual electronic system, with an Ortec 459 high-voltage supply, Ortec Model 401 Nim Bin and a Northern Scientific Model 623, 100 MHz analog-to-digital converters (ADC) as well as EGG developed analog pulse shaping modules.

The EGG developed modules include an R-C filter amplifier, positive baseline restorer, fast differentiating amplifier, negative baseline restorer, discriminator, pile-up rejector, and precision pulser. Measured system performance using a Nuclear Diodes 50 cc, 5-sided coaxial detector and Model 111 preamplifier and a Northern Scientific NS-623 100 MHz 4906

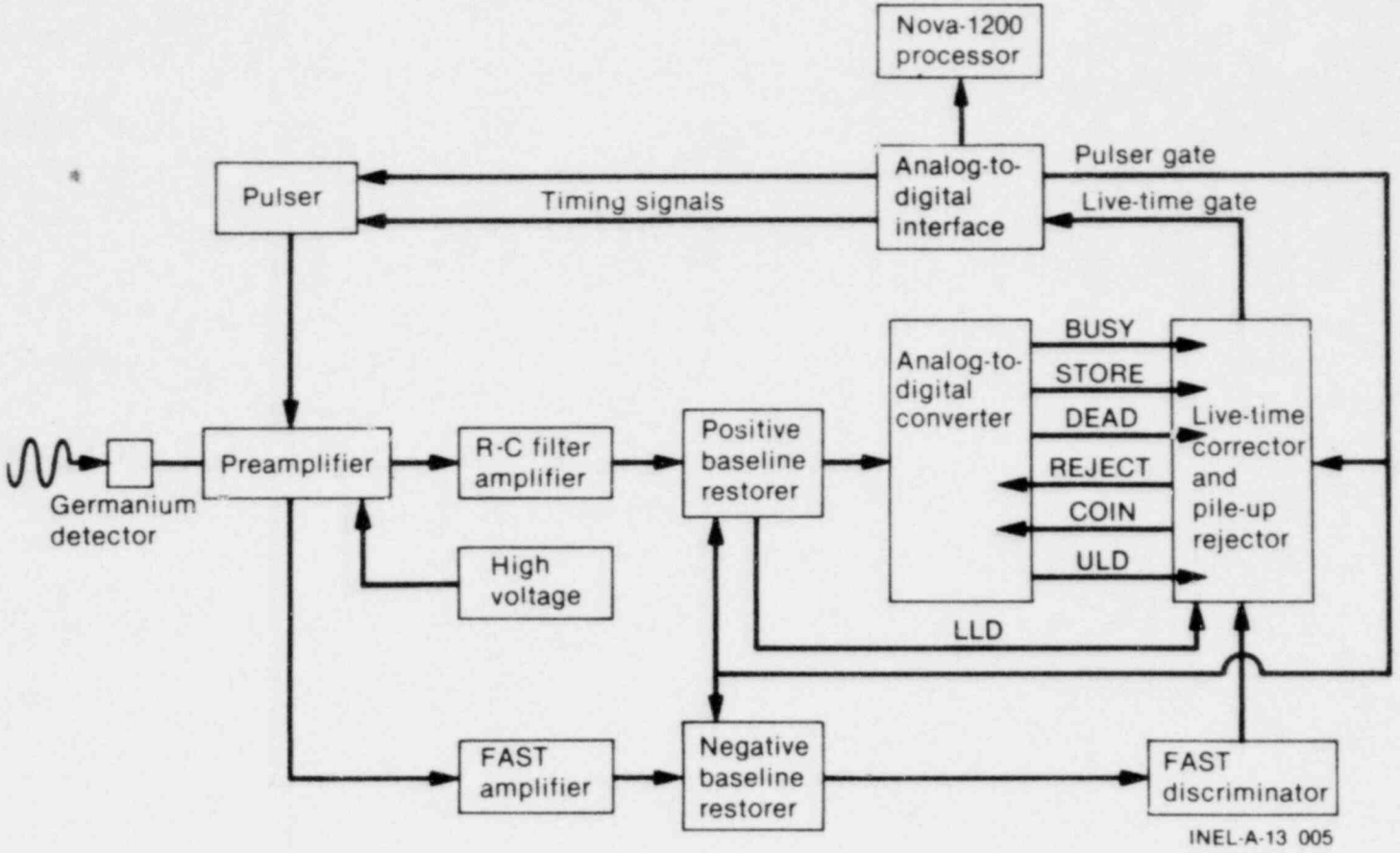


Figure 3. Front End Electronics

ADC is as follows: (Results are for a fixed ^{228}Th source at 5,000 counts per second (cps) and a movable ^{60}Co source to vary count rate. System gain was 0.705 keV/channel).

Zero shift - (input rate changing from 5,000 cps to 100,000 cps)
< 1 channel

Gain shift - (input rate changing from 5,000 cps to 100,000 cps)
< 0.01%

Peak Areas of Fixed Source Lines Deficit - 1.33%/10,000 cps
(energy independent)

Maximum Throughput - 17,500 stored events/second - occurs at
35,000 cps input rate

Resolution Degradation (FWHM) - 17% at 2614 keV line, < 30% at
300 keV line at 100,000 cps input rate.

This equipment was developed to satisfy all of the above specifications and to provide calibration information so that complete spectral validation, including energy calibration, energy independent pile-up correction, and live-time correction could be done reliably and repeateadly over an input counting rate range of up to 100,000 cps.

The R-C filter amplifier is a fairly standard active integrator design with 2 microsecond (μsec) differentiating time constant and active integrator stages which yield a peaking time of about 6 μsec and a baseline pulse width of about 18 μsec. There is no interstage capacitive coupling to interfere with accurate pole-zero compensated differentiation, and the amplifier times used in the filter amplifier are optimized to yield the best resolution for the detector used.

The positive baseline restorer is a highly assymmetric gated restorer with very high restore loop gain which enables it to hold the output baseline to less than 0.025% shift up to restorer duty cycles in excess of 90%. In a gated restorer, "droop" current is gated off for any input pulse exceeding 20 millivolts (MV) in amplitude so the restoration circuit has very little charge to replace on the restorer capacitor following each pulse. The restorer is also disabled for the duration of the negative pulse generated at the filter amplifier output on the trailing edge of the precision pulser square pulse. Restorer "droop" current is front panel adjustable to enable accommodation of detectors which exhibit a wide variation in low-frequency microphonics.

Pulse pile-up rejection is accomplished by processing the pulse train from the preamplifier separately through a 100 nanosecond (ns) differentiating amplifier, baseline restorer, discriminator, and pile-up rejector.

The differentiating amplifier is a direct coupled completely fed-back amplifier with frequency response capable of reproducing detector charge collection times (30-300 ns). The 100 ns differentiation is fully pole-zero compensated. It results in a maximum dead time per pulse of about 700 ns for non-overload pulses.

Output pulses from the differentiating amplifier are negative, so a high restore loop gain negative baseline restorer accepts differentiating amplifier output pulses and establishes a rate-stable baseline. This restorer will hold the baseline stable to less than 0.1% shift at duty cycles in excess of 90%. The negative restorer is also disabled for the duration of the negative pulse generated at the filter amplifier output by the trailing edge of the precision pulser square wave.

The fast discriminator inspects the negative restorer output and generates a logic-compatible pulse whose width is equal to the width of the negative restorer pulse in excess of the discriminator setting. Delay is < 15 ns and its dynamic range > 10,000:1.

The live-time corrector pile-up rejector requires 7 input signals:

1. Fast discriminator output - FD
2. Lower level discriminator (from positive restorer droop-current gate) - LLD
3. ADC Busy (indicates peak detection) - B
4. ADC Store (indicates that ADC has stored a pulse) - STORE
5. ADC Dead Time (indicates complete signal processing time in ADC) - DT
6. ADC upper level discriminator (indicates input pulse amplitude greater than range of ADC) - ULD
7. Blank (from pulser control, indicating trailing edge of pulser pulse) - Blank.

Output signals from the live-time corrector pile-up rejector are:

1. Clock Control (to ADC interface to control live-time clock)
2. Reject (to ADC, initiates reject cycle)
3. Input Rate (to external BNC) - R in
4. Throughput Rate (to external BNC) - R out
5. Coincidence (to ADC coincidence input to open linear gate) - Coin.

A pulse processing cycle is initiated by opening the ADC linear gate whenever a fast discriminator pulse is received if the LLD is reset and DT is reset (shaping amplifier and ADC ready to receive a pulse). If a second fast discriminator pulse is not received before peak amplitude is detected in the ADC, the pulse amplitude is digitized and stored. If a second fast discriminator pulse is received in this time period, an ADC reject cycle is initiated. Reject is also set and the linear gate is closed during the gate-off period for pulser calibrated systems.

Input discriminator signals are shaped to be wider than the widest negative restorer output pulse to insure energy independence of the pulse pile-up rejector circuit.

A corrected dead-time signal is generated in the live-time corrector by summing shaping amplifier dead-time (LLD), ADC dead-time (DT), and negative pulse gate time (for pulser calibrated systems), along with a pile-up rejector signal which operates as follows: Each time the ADC linear gate is opened by a FD pulse, the pile-up dead-time flop is set. This flop is reset only by ADC store or ADC ULD occurring before a reject pulse is generated. If the pulse is of an amplitude which would have fallen in the range of the ADC but was rejected due to pile-up detection, live-time has already been charged for the pulse. This live-time cannot conveniently be subtracted. Therefore, no more live-time is charged until after an uncontaminated pulse is processed.

The dead-time signal thus generated is a combination of true dead time plus a first order correction for pile-up losses. System live time is extended to compensate for pulse pile-up losses.

The precision pulser is a stable (1- ppm/year, 2 ppm/°C), two amplitude pulser housed in a temperature controlled enclosure. It is optically isolated from the pulser control interface, which determines both its frequency of operation and square pulse duration. The only shape pulse which may be fed into the preamplifier front end through the internal (to cryostat) charge injection capacitor without causing baseline perturbation is a square wave. The width of this pulse is set to yield a positive-negative pulse pair at the filter amplifier output separated in time by at least the pulse-pair resolution time of the filter amplifier. The entire system is gated off for the duration of the negative pulse with appropriate dead-time compensation.

II. THE ECLIPSE SYSTEM

This chapter deals almost exclusively with software. Hardware is mentioned only as needed to clarify points about software operations. A description of the hardware configuration is given in Chapter I: SOURCE TERM GAMMA-RAY SPECTROMETER SYSTEM.

The ECLIPSE software includes system software supplied by Data General, as well as application programs written specifically for the Source Term project. In the section on Data General's software, only those features pertinent to Source Term use is discussed. Additional information can be obtained from the various Data General user and reference manuals. This chapter describes the operational characteristics of the system only. A description of the algorithims used in each analysis step is described in Chapter III. A user command summary is given in Appendix A. A description of how to load and start the ECLIPSE System is given in Appendix B.

The user software for the ECLIPSE System has been written for the most part in the FORTRAN language, using Data General's F-5 compiler. A few utility programs and the foreground program for the acceptance of data from the NOVA computer have been written in assembly language. The use of assembly language has been used only where program size and speed were of primary importance.

II.1. DESCRIPTION AND USE OF THE ECLIPSE OPERATING SYSTEM

Data General: Real-Time Disk Operating System (RDOS) is used to control all Source Term operations on the ECLIPSE computer. Some of the major features of RDOS are:

- Disk and core resident operating system
- Multiple user overlays
- Spooling (disk buffering) of output
- Support for real-time FORTRAN 5
- Flexible file structuring
- Hardware protected foreground/background operation

This section describes the features most applicable to Source Term applications.

II.1.1. File Structure

.) Disk units, directories and file linkage:

There are two disk units in the system and all programs and data reside on one or the other of the two disks.

The fixed platter disk, called DPØF, is used primarily for storage of user programs, user libraries, system programs and utilities, and system libraries.

The removable platter disk, called DPØ, is used for storage of spectral data and for the results of spectral analyses.

The basic unit of storage on a disk is called a file. A file is a collection of information treated as a unit. There can be both data files or program files. An example of a data file would be a 4096 channel spectrum with its alphanumeric header. An example of a program file would be the object code capable of analyzing a spectrum. I/O devices are also treated as files by the operating system.

Files on the ECLIPSE system are grouped into logical sets called directories. For example, directory SPC contains all spectral data files; directory PROGM contains all the files needed to write and run user programs. A directory can be compared to a drawer in a file cabinet. All files within an open drawer are accessible. Files in other drawers cannot be directly accessed since only one drawer can be open at any one time.

The following chart illustrates the current disk file structure, and lists some of the directories on the two disk units. Figure 4 illustrates the relationship between the various directories.

DISK UNIT	DIRECTORY	FILE EXAMPLE
DPØF	PROGM - contains all files, and links to files in other directories, needed to write and run user programs	PDLIB - Parent daughter correction library
		MAIN.SV - general sample analysis program
		BACKGROUND.SV - background analysis program
		SEARCH - search table
		GAUSS - gauss fitting results

DISK UNIT	DIRECTORY	FILE EXAMPLES
		UTIL:SPEED.SV - text editor (link)
		UTIL:FORTRAN.SV - Fortran Compiler (link)
		LIB:MASTER - master isotope library (link) etc.
	LIB - contains isotope and limit libraries	MASTER - master isotope library
		LIMLIB - limit library etc.
	TABLE - linearity and efficiency tables	230.EF - efficiency table 230
		80.LN - linearity table 80 etc.
	UTIL - contains system utility programs	SPEED.SV - text editor
		FORTRAN.SV - Fortran 5 Compiler
		RLDR.SV - Linking Loader
		MAC.SV - Macro-assembler etc.
	F5 - contains Fortran 5 system routines	
	Other System Directories	
DPØ	SPC - contains raw spectral data data files	79011160.SP - main spectrum
		79010512.BK - background spectrum etc.
	BKG - contains results of background spectrum analyses	79010512.IS
	FINAL - contains final results of main spectrum analyses in summary format	79011160.SM

It is customary to speak of being "in" a directory. That is, at any one time, only one directory can be current and only its files can be accessed. If files are needed that exist in a different directory, they can be accessed by means of LINKs. A link is simply a pointer in one directory to a file in a different directory.

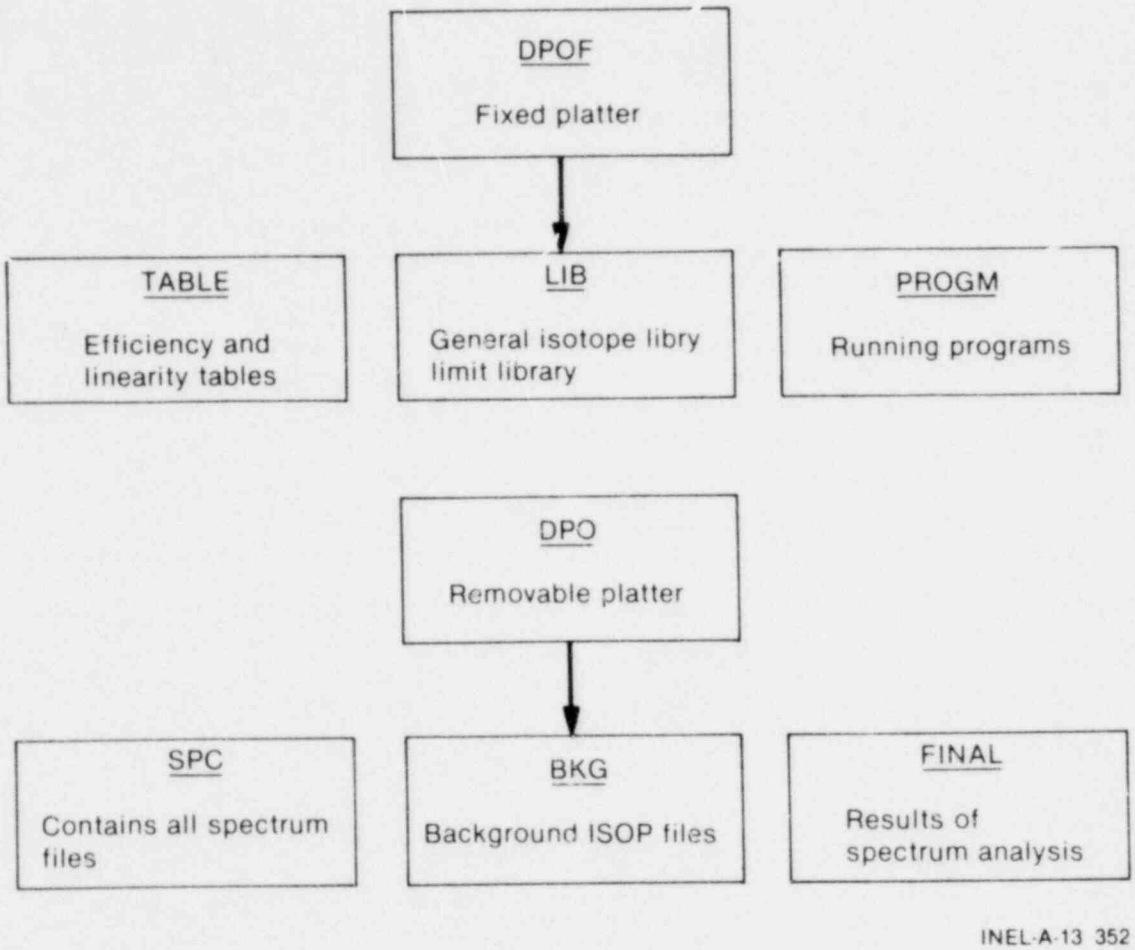


Figure 4. ECLIPSE Disk Structure

Generally, the user doesn't have to concern himself with linking to files in other directories. Linkage is done automatically in the programs that need the links and is therefore transparent to the user.

(1) File Naming Conventions

A file name can consist of up to ten alphanumeric characters (0-9 and A-Z) followed by an optional 2-character extension. If an extension is used, it must follow the file name and must be separated from the file name by a period. The following conventions have been adopted for Source Term applications:

(a) A file containing spectral raw data will have an 8-character numeric ID name followed by the extension SP or BK. The SP extension will be used for normal spectra and the BK extension for background spectra. All spectral data files will reside in directory SPC.

EXAMPLES: 79042601.SP, 79042602.BK } in directory SPC

(b) After a background spectrum has been analyzed, the results are saved for later use. The result file name will be identical to the background spectrum name but will have extension IS. Also, these background result files will reside in directory BKG.

EXAMPLE: 79042602.IS } in directory BKG

(c) After a normal spectrum has been analyzed, the results are saved for later reference. The result file name will be identical to the main spectrum name but will have extension SM (for summary). Also, these result files will reside in directory FINAL.

EXAMPLE: 79042601.SM } in directory FINAL

(d) All spectral analysis on the ECLIPSE involves a modular or stepwise approach. The main program controls the loading and execution of the modules and data is passed between modules via intermediate or temporary files. For example, the main program will load the search module which will search the spectrum and write the locations of peaks to file SEARCH. Next, the main program will load the gauss fitting module which will use the data in file SEARCH to perform a non-linear least-squares fit of a gaussian function to the peaks and write the results to file GAUSS. In other words, a module may produce one or more output files which are used as input to the next module.

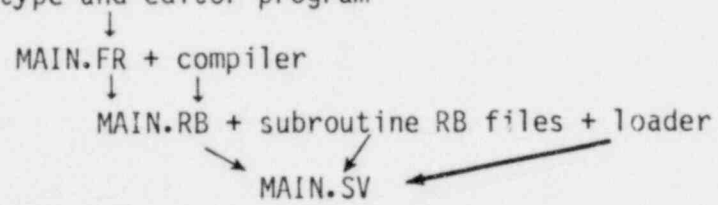
The data in these intermediate files is temporary. Data written to these files during one analysis is overwritten by data produced in a following analysis.

The file naming convention for intermediate files is simple: the file name will consist of up to ten alpha characters describing the type of data in the file. No extension will be used. Thus, in the example above, two of the intermediate files would have the names SEARCH and GAUSS.

(e) Data General has defined certain conventions that we are required to follow, especially for files involved in program development.

For example, consider the program that does a complete spectrum analysis. This program is called MAIN. The FORTRAN source code for program MAIN was loaded into the computer using the teletype and the Data General editor program. This source code was written to file MAIN.FR. The FORTRAN compiler takes the source code in file MAIN.FR and produces a relocatable binary file called MAIN.RB. The loader takes the binary file in MAIN.RB, links it with the binary files of the subroutines needed by MAIN, and produces a run or save file called MAIN.SV. It is this last file, MAIN.SV, that is actually read into core and executed when a spectral analysis is desired.

Thus, the following sequence must occur in the development of program MAIN:



The following is a list of file extensions that are defined by Data General for the ECLIPSE operating system:

.MC - macro command files
.SR - assembly language source files
.FR - FORTRAN source files
.RB - relocatable binary files
.SV - executable program files ("save" files)
.OL - program overlay files
.DR - directory files
.LB - library files
.CM - temporary command files

(f) A few words should be said about file linkage before ending this part of the discussion. There are a few application programs, written for Source Term, which require the user to input a link name. The reason for this is that the file needed by the program can exist in more than one directory. Thus, the user must specify the directory as well as the file name.

The link name is simply the directory name followed by a colon followed by the file name. For example, file 79042601.SM in directory FINAL would have the following link name:

FINAL:79042601.SM

(2) Spooling and Program Overlays

A detailed description of spooling and program overlaying is not really necessary for Source Term users. Their operation is automatic and completely transparent to the user. However, since they play such a vital part in the operation of the ECLIPSE system, a brief description is given.

All program-generated output to the line printer is spooled. That is, the output does not go directly to the output device. Instead, it is stored temporarily on disk and the operating system transfers the data to the device when the device is not busy.

The most obvious indication of spooling occurs when a program has completed execution, yet the line printer continues to print the program's output. At times, the user will notice that printing continues for quite a lengthy time after program execution. There is nothing wrong with this. In fact, after the "R" prompt has been printed on the teletype, the user can start a new program execution, even though the line printer is still busy.

The obvious benefit of spooling is that the processor is used much more efficiently than if spooling were not used. Data can be written to disk much more rapidly than it can be written to the line printer. Program execution does not have to wait for line printer availability.

Program overlaying is the memory saving technique whereby program modules or subroutines are read into core from disk as they are needed. When they are no longer needed, the same area in memory is used for other modules or subroutines.

Program overlaying is used quite extensively for SOURCE TERM spectrum analysis. To do a complete analysis, from searching the spectrum for peaks to doing the final parent-daughter decay corrections, requires much more core memory than is available. Program overlaying overcomes the memory limitations with only a very slight execution-time penalty.

II.1.2 CLI and How to Use It

The CLI (Command Line Interpreter) is a continuously running system program which acts as the interface between the user and the ECLIPSE operating system. The letter "R", printed on the teletype, is the CLI prompt. It indicates that the CLI is ready for input.

In operation, the user types in a system command or a program name. If the input is a system command, the CLI performs the appropriate action and, when it is done, it types an R prompt. If a program name is entered, the CLI loads the program and starts execution. After the program is finished executing, CLI takes over and types an R prompt.

If the user wishes to terminate command or program execution prematurely, he can do so by typing a CNTRL A. CLI will return with an R prompt.

Of the more than seventy CLI commands available to the user, only a few will be discussed here. If, after reading this section, the user wishes to learn more about the CLI, he can refer to Data General's "RDOS COMMAND LINE INTERPRETER" Reference Manual.

The following CLI commands will be most useful in SOURCE TERM applications:

GDIR - Print the name of the current directory.

This command is useful if the user is not sure which directory he is currently in. Simply type GDIR and the current directory will be printed on the teletype.

DIR - Change the current directory to a new directory.

Use the DIR command to change the current directory if you are not in the directory you want to be in.

Example:

```
DIR PROGM
```

This command will put the user in directory PROGM.

RENAME - Change a file name.

This command is particularly useful for Source Term because background spectra must be renamed before they can be used.

When a spectrum is transferred from the NOVA to the ECLIPSE, the file is automatically given the extension, ".SP" (e.g., 7901160.SP). Since the system has no way of knowing if the spectrum is a background spectrum, it is up to the user to give the file name the correct extension, ".BK". To do this, type the following:

```
DIR SPC
RENAME 79011160.SP 79011160.BK
DIR PROGM
```

The DIR SPC command puts the user in the directory where the spectra are located. The RENAME line gives the file the correct extension. The DIR PROGM command puts the user back in the correct directory to analyze the spectrum.

The RENAME command is not limited to just changing the extension of a file name. For example:

```
RENAME ROMEO.AB JULIET.CD
```

In this command, the old file named ROMEO.AB had its name changed to JULIET.CD.

DELETE - Delete a file.

Use this command to delete a file that you no longer want. For example:

```
DELETE 79011160.BK
```

LIST - List file directory information.

Use this command to list the contents of the current directory. The command can also be used to determine if a specific file exists in the current directory.

EXAMPLE:

```
LIST
```

This simple command will type the names of all the files and links in the current directory on the teletype.

EXAMPLE:

```
LIST 79011160.BK
```

If file 79011160.BK exists in the current directory, its name will be printed along with the size of the file. If it does not exist, a message will be printed indicating that the file does not exist in the current directory.

NOTE: Only two variations of the LIST command have been shown above. As with most other CLI commands, there are several other options available to the user. For more information, refer to the CLI reference manual.

INIT - Initialize a directory or a device.

This command is mentioned here because the magtape unit must be initialized before it can be used. All other directories are initialized automatically when the system is bootstrapped.

To initialize the magtape unit, load a tape and set the unit on line. Next, type the following:

```
INIT MTØ
```

where the "0" in MT0 is the number zero and not the letter "o". After this command is given, programs written for the magtape unit can be run.

II.1.3 Foreground-Background Operation

The ECLIPSE real time disk operating system allows two programs to be core-resident at the same time in two different sections of memory. These sections are called the foreground and the background. The operating system switches control between the two sections, the net effect being similar to time-sharing.

For Source Term applications, the foreground is used exclusively to control data transfer and formatting between the NOVA and ECLIPSE computers. Foreground operation is transparent to the user. All other applications (spectral analysis, new program development, etc.) are be done in the background.

The foreground program (SPECIN) is started automatically as part of the GO command in the bootstrap procedure. It will continue to run, without user intervention, as long as the system is running. Only in the case of certain error conditions does the user have to concern himself with the foreground program.

The SPECIN program is written in assembly language. It monitors and controls the serial interface between the NOVA-1200 and ECLIPSE. The sequence of operations performed by the program when it accepts a spectrum from the NOVA is as follows:

(1) Read spectrum header ID block - 256 words of alpha-numeric data.

(2) Extract the first 8 characters from the ID block. These 8 characters become the spectrum ID.

(3) Create a file in directory SPC whose name is the 8 characters extracted from the ID block.

(4) Write the ID block to the file which was just created.

(5) Read the spectral data in blocks of 256 words, change the data to floating point and write the information to the new file.

Several types of errors can occur in the running of program SPECIN. These errors are reported to the user by the computer typing the following on the operators console:

```
ERROR N
!! F G TERM
```

The letter N will be a two digit integer which indicates the type of error. The message !! F G TERM indicates that the foreground program SPECIN has been terminated. Specific actions must then be taken to restart the foreground program. The following table gives the various error numbers which might occur and the action required to eliminate the problem.

ERROR	CAUSE	CORRECTIVE ACTION
01	Cannot open the serial line for input	Hardware error; try again then call repairman
02	Cannot open the serial line for output	Hardware error; try again then call repairman
05	Did not receive the correct code to start the data transfer between the two computers	Reset the NOVA and try again; if same failure occurs, call repairman
06	Did not receive the spectrum ID block correctly	Same as for error 05
07	Unable to create the spectrum file as specified in the ID block. Several probable causes: 1. Spectrum already exists with the same name. 2. Disk is full - no other spectra can be added. 3. The spectrum ID is not in the ID block.	Make sure there is not a duplicate spectrum already on the disk. If there is no duplicate spectrum file the disk could be full. Delete a spectrum and try the process again. If none of the above procedures work, check the spectrum header with the NOVA 'P' command. Make corrections if necessary and retransmit.
08	Unable to open newly created file.	Hardware error; delete newly created file; try again then call repairman
11	Unable to write ID block	Same as error 08
12	Unable to send command to NOVA	Same as error 08
13	Unable to receive spectrum block	Same as error 08
20	Unable to write spectrum data block to disk	Same as error 08

Whenever an error occurs in the transmission of data between the two computers, the following steps must be taken.

1. If the NOVA-1200 detected the error also and types 'ASYNC LINE ERROR' then no action is required for the NOVA. If the NOVA did not detect the error, it must be initialized by typing CNTRL C or restarting the computer.

2. Restart the foreground execution by typing the command

EXFG SPECIN

on the ECLIPSE operator's console.

The operation of the foreground program can be terminated by typing a CNTRL F. If the operation of the foreground program has been terminated, it must be started with the EXFG SPECIN before any attempt is made to transmit another spectrum to the ECLIPSE. If an error in data transmission is detected by the NOVA computer, it will type the message 'ASYNC LINE ERROR' on the NOVA's operator's console. The ECLIPSE operator must then terminate and reinitialize the foreground with the CNTRL F, EXFG SPECIN sequence. This sequence of operations must be performed in the order specified above to insure that both computers are synchronized.

II.2. DESCRIPTION AND USE OF ECLIPSE SOFTWARE FOR SOURCE TERM

Several user programs have been written specifically for Source Term applications. These can be grouped as follows:

a. Spectral Analysis Routines

b. Printing Routines

c. Library Editing Routines

d. Magtape Routines

e. Macro commands

The programs in each group will be discussed, individually, with detailed instructions for their use, on the following pages.

II.2.1 Spectral Analysis Procedures

There are several spectral analysis procedures available to the ECLIPSE operator. All analysis procedures perform a set of functions which are a subset of the basic analysis procedure, MAIN. The program, BACKGROUND, is used to analyze a background spectrum. The results from the background analysis are put in a form which can be used by MAIN to perform background subtractions. The algorithms of these two routines and instructions for their use are described in paragraphs II.2.1.1 and II.2.1.2. Both routines have optional features which can be invoked by setting certain switches on the ECLIPSE console. During program operation, the switch is tested and, if it is set, the appropriate action is taken. The operation and use of the other analysis procedures is given in paragraph II.2.1.3.

II.2.1 Background Analysis Procedure

Program BACKGROUND will analyze a background spectrum file. The results of the analysis will be saved on disk so that background corrections can be done later in program MAIN.

The result file produced by program BACKGROUND will have the same 8 character numeric ID as the background spectrum file but will reside in directory "BKG" and will have the extension ".IS".

The following chart illustrates the action taken when program BACKGROUND is run. As an example, we will assume that the spectrum to be analyzed has the ID 79011052. We will also assume that none of the optional switch-selected features are to be invoked.

INPUT	ACTION TAKEN	OUTPUT
User types: 79011052	(1) Read in background spectrum ID name.	None
file SPC:79011052.BK	(2) Open the file.	None
file SPC:79011052.BK	(3) Do an energy calibration	Calibration constants
file SPC:79011052.BK	(4) Search the spectrum for peaks. Write search table to file SEARCH.	file SEARCH
file SPC:79011052.BK file SEARCH	(5) Gauss the peaks and write results to file GAUSS. Save the locations of unfittable peaks in file EXCESS.	file GAUSS file EXCESS
file SPC:79011052.BK file LIMLIB file EXCESS	(6) Do limit calculations on peaks listed in limit library that were not fitted by gauss routine. Also do limit fits to peaks listed in file EXCESS. Add all fit results to file GAUSS.	file GAUSS
file GAUSS file MASTER	(7) Tag peaks in file GAUSS and write results to file ISOP. Write gauss results of peaks that cannot be tagged to file UNTAG. The master isotope library will be in file MASTER.	file ISOP file UNTAG

INPUT	ACTION TAKEN	OUTPUT
file ISOP	(8) Eliminate results which are not greater than some detection limit from file ISOP	file ISOP
file ISOP	(9) Copy results to final destination.	file BKG:79011052.IS

The following console switches can be set to invoke additional features:

SWITCH NUMBER	FUNCTION
1	Print energy calibration results.
2	Request the user to type in the search sensitivity (the default is 10).
3	Print the search table (file SEARCH) and intermediate gauss iterative results.
4	Print the contents of files ISOP and UNTAG produced in step (7).
5	Allow the user to edit file ISOP before it is copied to the final destination in step (9). Editing allows the user to add new entries to the file or to delete existing entries. The editing routine asks the user simple questions to determine the type of editing to be done.
10	Print limit fitting intermediate results.

If both switchs 4 and 5 are set, then the contents of the edited file will also be printed. All printing is done on the line printer.

To run program BACKGROUND, set the current directory to PROGM (i.e., DIR PROGM) if the current directory is not already PROGM. Next, select any of the optional features by setting the appropriate console switch. Finally, enter the program name, BACKGROUND, on the teletype to start the program. You will be requested to enter the background spectrum ID (e.g., 79011052). The rest is automatic.

II.2.1.2 Main Analysis Procedure

Program MAIN will do a normal spectrum analysis. The results of the analysis will be saved permanently on disk, unless deleted by the user. The result file produced by program MAIN will have the same 8 character numeric ID as the spectrum file but will reside in directory "FINAL" and will have the extension ".SM".

The following chart illustrates the action taken when program MAIN is run. As an example, we will assume that the spectrum to be analyzed has the ID 79011160 and that the appropriate background spectrum (previously analyzed using program BACKGROUND) has the ID 79011052. We will also assume that none of the optional switch-selected features are to be invoked.

INPUT	ACTION TAKEN	OUTPUT
file SPC:79011160.SP	(1) Read in spectrum ID name	None
	(2) Open the file.	None
	(3) Search directory BKG for the background spectrum	ID name 79011052
file BKG:79011052.IS	(4) Open the file.	None
file SPC:79011160.SP	(5) Do an energy calibration	Calibration constants
file SPC:79011160.SP	(6) Search the spectrum for peaks. Write search table to file SEARCH.	file SEARCH
file SPC:79011160.SP file SEARCH	(7) Gauss the peaks and write results to file GAUSS. Save the locations of unfittable peaks in file EXCESS.	file GAUSS file EXCESS
file SPC:79011160.SP file EXCESS file LIMLIB	(8) Do limit calculations on peaks listed in limit library that were not fitted by gauss routine. Also do limit fits to peaks listed in file EXCESS. Add all fit results to file GAUSS.	file GAUSS

INPUT	ACTION TAKEN	OUTPUT
file GAUSS file MASTER	(9) Tag peaks in file GAUSS and write results to file ISOP. Write gauss results of peaks that cannot be tagged to file UNTAG. The master isotope library will be in file MASTER.	file ISOP file UNTAG
file ISOP	(10) Copy file ISOP to file EDISOP. (Editing would be done here if switch 5 were set. Since it is not set, only a copy is done.)	file EDISOP
file EDISOP file BKG:79011052.IS	(11) Do background corrections.	file BKGCORISOP
file BKGCORISOP file INTLIB	(12) Do interference decontaminations. Interference library will be in file INTLIB.	file INTCORISOP
file INTCORISOP	(13) "clean-up" the results. (Eliminate peaks with greater than 100% error. Convert results to "less-than" if primary or clean line is not present, etc.)	file CLNUPISOP
file CLNUPISOP	(14) Resolve multiple gamma lines to single weighted averages.	file FINAL
file FINAL file PDLIB	(15) Do parent-daughter decay corrections. PD library will be in file PDLIB.	file FINAL:79011160.SM

The following console switches can be set to invoke additional features:

SWITCH NUMBER	FUNCTION
1	Print energy calibration results.
2	Request the user to type in the search sensitivity (The default is 10).
3	Print the search table (file SEARCH) and intermediate gauss iterative results.

SWITCH NUMBER	FUNCTION
4	Print the contents of files GAUSS and UNTAG produced in step (9).
5	Allow the user to edit file ISOP before it is copied to file EDISOP in step (10). Editing allows the user to add new entries to the file or to delete existing entries. The editing routine asks the user simple questions to determine the type of editing to be done.
6	Print background-corrected isotope file (file BKGCORISOP) producted in step (11).
7	Print interference-decontaminated isotope file (file INTCORISOP) produced in step (12).
8	Print "cleaned-up" isotope file (file CLNUPISOP) produced in step (13).
9	Print results of multiple gamma line resolutions (file FINAL) produced in step (14).
10	Print limit fitting intermediate results.

If both switches 4 and 5 are set, then the contents of the edited file (file EDISOP) will also be printed. All printing is done on the line printer.

To run program MAIN, set the current directory to PROGM (i.e., DIR PROGM) if the current directory is not already PROGM. Next, select any of the optional features by setting the appropriate console switch. Finally, enter the program name, MAIN, on the teletype to start the program. You will be requested to enter the spectrum ID (e.g., 79011160), and the background spectrum ID (e.g., 79011052). The rest is automatic. At the end of the analysis, parent-daughter corrected results will be printed on the line printer with a summary of the results preselected for the Source Term project. No special switches need to be set for this printout.

These special analysis procedures are started by the operator typing the routine name. All required operator input is identified by the computer printing prompt messages on the operator's console. The same switch settings which are used for routine MAIN are also used for these special analysis procedures.

During the execution of the main analysis procedure, the analyzed lines can be operated on in various ways, i.e., limit lines, doublet lines fit by GAUSS, interference decontaminated, etc. The operations performed on each peak are indicated by numeric values printed in the flag column on all analysis printouts. The following table gives the meanings of the various flag values.

FLAG VALUES:

0	-	"NORMAL" GAUSS result, singlet
1	-	"NORMAL" LIMFIT result, singlet
2	-	"NORMAL" GAUSS result, part of a multiplet
3	-	"NORMAL" LIMFIT result, part of a multiplet
4	-	Code is not used
5	-	"LESS-THAN" LIMFIT result, singlet
6	-	Code is not used
7	-	"LESS-THAN" LIMFIT result, part of a multiplet
8	-	Background-corrected, "NORMAL" result
16	-	Background-corrected, "LESS-THAN" result
32*	-	Interference-decontaminated, "NORMAL" result
64*	-	Interference-decontaminated, "LESS-THAN" result
128	-	Listed as "CLEAN" in Interference Library
256	-	Converted to "LESS-THAN" in CLNUP
512	-	OUTLIER - not used in weighted average

For flag values 0 through 7, the three corresponding bit positions (0,1 and 2; bit 0=LSB) have the following values:

Bit 0	-	0 for GAUSS; 1 for LIMFIT
Bit 1	-	0 for a singlet; 1 if part of a multiplet
Bit 2	-	0 for "NORMAL" result; 1 for "LESS-THAN" result

Thus, the combinations with values 4 and 6 are not possible since they would indicate "LESS-THAN" results for GAUSS fits.

The net value of a flag is the sum of the appropriate values listed above, except for flag values less than 8 which are broken down individually. As an example, a net value of 554 (=2+8+32+512) indicates the following:

1. The initial gauss fit result was normal. The peak was part of a multiplet.

2. The peak was corrected for background. The result was "NORMAL".

3. The peak was interference-decontaminated. The result was "NORMAL".

4. The peak was rejected as an outlier and was not used in the final weighted average.

*A value of 96 (=32+64) is possible if a double decontamination is done. The first decontamination would have to yield a "NORMAL" result and the second decontamination would have to yield a "LESS-THAN" result.

II.2.1.3 Special Analysis Procedures

The special analysis procedures and their functions are given in the following table.

ROUTINE NAME	FUNCTION
GAMM	Performs the same functions as MAIN except the analysis procedure is terminated after the analyzed lines have been identified and written to file ISOP.
UMAIN	Performs the same analysis functions as MAIN except the pulser data is not used to energy calibrate the spectrum. User supplied parameters are used to establish the energy equation.
UGAMM	Performs the same analysis functions as GAMM except the energy calibration is performed in the same manner as UMAIN.
THORIUM	This routine allows the operator to energy calibrate using a thorium spectrum, then impose the calibration on another spectrum for analysis using the MAIN procedure.
BUNCH	Allows several spectra to be analyzed with the same background using the MAIN procedure.
STDMAIN	Allows the user to perform analysis using the MAIN procedure, except the isotope library, STANDARD, is used for peak identification and the limit library STDLIM is used to perform limit calculations.

II.2.2 Printing Routines

There are special purpose printing programs that have been developed for Source Term use. They are:

(1) PRINTGAUSS - Print the current contents of file GAUSS.

(2) PRINTUNTAG - Print the current contents of file UNTAG.

(3) PRINTISOP - Print the contents of an ISOP-type file. Files with extension ".IS" in directory "BKG" also fall into this category.

(4) PRINTFINAL - Print the contents of a FINAL-type file. Files with extension ".SM" in directory "FINAL" also fall into this category.

(5) PRINTINTLB - Print the contents of the interference-decontamination library.

(6) PRINTPDLIB - Print the contents of the parent-daughter decay correction library.

All printing is done on the line printer.

Each of the programs is described individually below.

PROGRAM PRINTGAUSS

This program will print the current contents of file GAUSS. No special input is required. Before running the program, the user must be in directory PROGM. To run the program, simply enter the program name, PRINTGAUSS, on the teletype.

The user should be aware that the contents of file GAUSS changes with each spectrum analysis. That is, each time program MAIN or BACKGROUND is run, the old results in file GAUSS are overwritten with new data. Thus, if a printout of file GAUSS is needed, it must be done before starting the next spectrum analysis.

PROGRAM PRINTUNTAG

This program will print the current contents of file UNTAG. No special input is required. Before running the program, the user must be in directory PROGM. To run the program, simply enter the program name, PRINTUNTAG, on the teletype.

The user should be aware that the contents of file UNTAG changes with each spectrum analysis. That is, each time program MAIN or BACKGROUND is run, the old results in file UNTAG are overwritten with new data. Thus, if a printout of file UNTAG is needed, it must be done before starting the next spectrum analysis.

PROGRAM PRINTISOP

This program will print the contents of an ISOP-type file. The following temporary files (in directory PROGM) can be printed with this routine:

ISOP
EDISOP
BKGCORISOP
INTCORISOP
CLNUPISOP

All permanent files in directory BKG can be printed with program PRINTISOP, since they are also in the correct format.

Before running the program, the user must be in directory PROGM. To start the program, simply enter the program name, PRINTISOP, on the teletype.

The program will ask the user to type in the file's link name. This is necessary because the temporary files are located in directory PROGM, while the permanent files are in directory BKG. The link name consists of the directory name followed by a colon followed by the complete file name.

EXAMPLES:

PROGM:BKGCORISOP
BKG:79011052.IS

Keep in mind that the contents of the temporary files change with each spectrum analysis. Thus, if a printout of one of these files is needed, it must be done before starting the next spectrum analysis.

PROGRAM PRINTFINAL

This program will print the contents of a FINAL-type file. There is only one temporary file, file FINAL (in directory PROGM), that falls into this category. All permanent files in directory FINAL (do not confuse with file FINAL) can be printed by program PRINTFINAL, since they are also in the correct format.

Before running the program, the user must be in directory PROGM. To start the program, simply enter the program name, PRINTFINAL, on the teletype.

The program will ask the user to type in the file's link name. This is necessary because the temporary file, FINAL, is located in directory PROGM, while the permanent files are in directory FINAL. The link name consists of the directory name followed by a colon followed by the complete file name.

EXAMPLES:

PROGM:FINAL
FINAL:79011160.SM

Keep in mind that the contents of the temporary file change with each spectrum analysis. Thus, if a printout of temporary file FINAL is needed, it must be done before starting the next spectrum analysis.

PROGRAM PRINTINTLB

This program will print the contents of the interference-decontamination library, file INTLIB. No special input is required. Before running the program, the user must be in directory PROGM. To run the program, simply enter the program name, PRINTINTLB, on the teletype.

PROGRAM PRINTPDLIB

This program will print the contents of the parent-daughter decay correction library, file PDLIB. No special input is required. Before running the program, the user must be in directory PROGM. To run the program, simply enter the program name, PRINTPDLIB, on the teletype.

II.2.3 Library Editing Routines

There are two (2) library editing routines that have been developed for Source Term use. They are:

(1) EDITINTLIB - Edits the interference-decontamination library.

(2) EDITPDLIB - Edits the parent-daughter decay correction library.

All editing is done by the user on the teletype.

Each of the four programs is described individually below.

PROGRAM EDITINTLIB

This program will allow the user to edit the interference-decontamination library on file INTLIB.

Before running the program, the user must be in directory PROGM. To start the program, simply enter the program name, EDITINTLIB, on the teletype.

The program will print the available editing options on the teletype as follows:

1 - Start new file
2 - Add an entry
3 - Delete an entry
4 - Print file
5 - End edit

It will then ask the user to type in the number of the option he chooses. After all editing is done, the user can end the program by typing the number 5 when an edit choice is requested.

If the user chooses to start a new file (edit choice=1), then all existing entries are deleted.

If the user chooses to add an entry (edit choice=2), the program will request the energies and isotopic identification data for both the contaminated and clean lines. The user should have this information ready before starting the library editing. Entries are added to the end of the library. The file INTLIB can hold up to 21 entries.

If the user chooses to delete an existing entry (edit choice=3), the program will request the contaminated and clean line energies of the entry to be deleted. An entry is deleted by "moving up" all the entries that follow it, if any, by one position.

If the user chooses to print the file (edit choice=4), the library will be printed on the line printer.

IMPORTANT NOTE: The decontamination routine was designed so that any peak could be decontaminated and the result later used as a clean peak in another decontamination. For example, ^{131}I at 772 kev can be decontaminated of ^{187}W. The new ^{132}I result could then be used to decontaminate ^{58}Co at 810 keV. The new ^{58}Co result could be used to decontaminate another peak, etc.

In order to insure correct results for this type of situation, the library entries must be in the proper order. Thus, the first decontamination must be somewhere ahead of the second decontamination in the library. The second must be somewhere ahead of the third, etc. They do not need to be contiguous.

In some cases, it may be necessary to first delete an entry and then re-enter it so that its new position will be at the end of the library. As an example, if you accidentally add the second decontamination before you add the first, it will be necessary to delete the second decontamination and add it a second time so that it follows the first decontamination.

PROGRAM EDITPDLIB

This program will allow the user to edit the parent-daughter decay correction library on file PDLIB.

Before running the program, the user must be in directory PROGM. To start the program, simply enter the program name, EDITPDLIB, on the teletype.

The program will print the available editing options on the teletype as follows:

1 - Start a new library
2 - Add a decay chain
3 - Delete a decay chain
4 - Print library
5 - End edit

It will then ask the user to type in the number of the option he chooses. After all editing is done, the user can end the program by typing the number 5

when an edit choice is requested.

If the user chooses to start a new library (edit choice=1), then all existing entries are deleted.

If the user chooses to add a decay chain (edit choice=2), the program will request the atomic mass number, A, for the chain, the number of members in the chain, and isotopic identification data for each member. The user should have this information ready before starting the program. Only chains with beta decays and isomeric transitions are allowed. Alpha decay chains are not allowed. The library can hold enough data for up to 36 chains. The order of the entries in the library is not important.

If the user chooses to delete an existing decay chain (edit choice=3), the program will request the atomic mass number, A, for the chain.

If the user chooses to print the library (edit choice=4), it will be printed on the line printer.

II.2.4 Magtape Routines

There are three magtape routines that have been developed for Source Term use. They are:

(1) MTLIST - List the contents of a magtape on the line printer.

(2) MTWRITE - Write spectral data to magtape.

(3) MTREAD - Read spectral data from magtape.

Each of the programs is described individually below.

PROGRAM MTLIST

This program will print out the file number and the alphanumeric header block of each spectral data file on a magtape. All printing will be done on the line printer. No special input to the program is required by the user.

Before running the program, the tape should be loaded, the magtape unit should be put on-line and initialized (INIT MT0), and the user should be in directory PROGM. To start the program, simply enter the program name, MTLIST, on the teletype. The rest is automatic.

PROGRAM MTWRITE

This program writes spectral data from directory SPC to magtape. Each spectrum occupies one file and up to 100 files can be written to one tape.

Before running the program, the tape should be loaded, the magtape unit should be put on-line and initialized (INIT MT0), and the user should be in directory PROGM. To start the program, simply enter the program name, MTWRITE, on the teletype.

The program will start by asking the user:

ENTER START FILE NUMBER (00-99):

The user must enter a two digit number indicating at which file writing is to start. For example, if four files are already on the tape (00-03), and you do not want to erase them, then answer with "04". If you wish to keep only the first two files (00 and 01), then answer with "02". If you wish to start a new tape, then answer with "00".

Next, the program determines which spectra are to be written to magtape from the user's answer to the prompt:

ENTER SPECTRUM ID NAME(S):

There are six acceptable answering formats to the above prompt. They are illustrated with the examples below:

INPUT EXAMPLE	MEANING
790405	Write all spectra that were taken on April 5, 1979.
79040512	Write the spectrum with sample number 12 taken on April 5, 1979.
790405-790502	Write all spectra taken between April 5 and May 2, 1979, inclusive.
79040510-79050203	Starting with sample number 10 taken on April 5, 1979, write all spectra up to and including sample number 03 taken on May 2, 1979.
790405,790123,etc.	Write all spectra taken on the dates indicated. Any number of dates can be entered as long as they fit on one line and are separated by commas. The dates can be in any order.
79040512,79031204, etc.	Write the individual spectra indicated. Any number of spectra can be entered as long as they fit on one line and are separated by commas. The spectra can be in any order.

For each spectrum written to magtape, the file number and the spectrum ID will be printed on the teletype. Only those spectra that are found in directory SPC will be written to tape. Thus, if the user answers with 790405, the program will search for 79040500, 79040501, 79040502, . . . , 79040599, but it will only write to magtape the spectra that it finds.

After the program has processed the input line, it will repeat the prompt, allowing the user to do more writing. When the user finishes with all writing, he should answer the prompt by simply hitting the RETURN key which will end the program.

PROGRAM MTREAD

This program will read a spectrum from the magnetic tape and write it to disk in directory SPC. The user is required to input the file number of the desired spectrum. The first spectrum on the tape is in file 0, the second in file 1, etc.

Example 1, Reading the 11th spectrum

```
R
MTREAD
FILE #: 10
R
```

II.2.c. Macro Commands

Macro commands are used in the ECLIPSE System to allow the user to perform a series of predetermined CLI commands by typing a one-word command. The macro commands of use to the operator of the ECLIPSE are GO, OFF and STOP. The following paragraphs describe the use of these commands.

GO

This command is used by the operator when the ECLIPSE has been shut down and the system is being reinitiated. This command should be issued after the time and date have been entered in the initial dialogue and the system is in directory DPØF. The command performs the following functions:

1. Gets the background-foreground memory portion.
2. Starts the foreground running.
3. Opens all necessary directories.
4. Goes to directory PROGM

OFF

This command will stop the line printer and kill any line printer spooling in progress. The command must be issued from directory PROGM.

STOP

The STOP command is used to terminate the operation of ECLIPSE System. It releases all open files and halts the ECLIPSE. STOP can be issued only when the foreground is not running. To stop the foreground type CNTRL F. The system can then be restarted using the startup procedure givein in Appendix B. The following example illustrates the shutdown procedure.

Example 2, ECLIPSE Shutdown

```
R
↑F
FG TERM

R
STOP )
MASTER DEVICE RELEASED
```

II.3 ERROR CONDITIONS AND THEIR CODES

All of the programs written for the ECLIPSE System have unique error codes associated with the operation of each routine. If a routine detects an error condition it will type the message STOP N on the operator's console. N will be an integer. The operator can then determine what the error was and what routine detected the error by referring to the list of stop code values given in Appendix C.

III. ECLIPSE DATA ANALYSIS ALGORITHMS

This chapter describes the algorithms used in the analysis of gamma-ray pulse height data on the ECLIPSE computer. The analysis procedure invokes a series of subroutines each of which performs a specific function. All routines are written in the FORTRAN language and are as independent from each other as possible. The use of large common blocks for subroutine communication have been held to an absolute minimum with communication between routines accomplished via disk files. This approach has made it easy to modify any one routine without impacting the others, moreover, the task of changing the analysis routines has been simplified. The following routines are described in this chapter:

PICLB	-	Energy calibrates the spectrum.
SEARCH	-	Locates the position of peaks in the spectrum.
GAUSS	-	Fits gaussian function to the data.
LIMFIT	-	Performs limit calculations.
ISOLK	-	Performs isotopic identification and computer decay corrections.
BKGRD	-	Corrects spectrum for the counting background.
INTFER	-	Performs interference-decontaminations on peaks contaminated by contributions from other isotopes.
CLNUP	-	Eliminates isotopic identifications based on major lines from that isotope being present or absent in the spectrum.
RESOL	-	Averages the results obtained from several lines to obtain a quantitative result for an isotope.
PDCOR	-	Performs parent-daughter corrections

Our goal in writing this chapter is to give the user a basic understanding of the calculational processes involved. It is not intended to give the user sufficient knowledge to re-write or in any way to modify existing routines.

III.1 SPECTRUM ENERGY CALIBRATION

The normal method used to energy calibrate the spectrum is via the precision pulser data located in the spectrum. The NOVA-1200 computer (the data collection processer) collects the pulser data and places it in channels 25 - 58 of the spectrum which is then transmitted to the ECLIPSE. The precision pulser is a two-amplitude pulser, consequently the pulser data is a two-peak spectrum. PICLB extracts the position of these two peaks from channels 25 - 58 of the spectrum and calls subroutine GAUSS which fits a gaussian function to these peaks. The result of this fit is then used to establish the position of the peaks from which the energy scale of the spectrum is computed. The result of this energy calibration is then placed in the calibration block for the spectrum. Subsequent analysis routines extract the energy information as required.

More information than energy scale determination is available from the pulser data. As a matter of fact, the information obtainable from the pulser data can give an indication of how the spectrometer front-end is performing. For example, the width of the pulser peaks indicate the amount of electronic noise currently in the system. Several checks are made on the pulser data to verify that the system is operating correctly. A running history of various pulser parameters extracted from the pulser data is kept in a pulser history file which can be used as a quality-control tool. Moreoever, if the computer determines some parameter is not within a specified range a warning message is printed. The parameters which are kept in the history file are given below:

(1) Pulser widths

(2) Pulser amplitudes

(3) Pulser peak areas

(4) Total pulser counts

(5) Pulser energy equivalents

(6) Pulser rejects

(7) Total gamma counts

III.2 PEAK LOCATION

The purpose of the peak location algorithm is to locate peaks in the spectrum and place their locations in file SEARCH. This file is then read by the GAUSS fitting subroutine for the locations of peaks to analyze. The maximum number of peak locations which can be placed in SEARCH is 511.

The peak location or search algorithm used in the ECLIPSE system is a simple cross-correlation of the square root (truncated to an integer) of the spectral data with a zero area square wave. The cross-correlation is performed using integer arithmetic. Consequently, the algorithm is very fast.

The zero area square wave correlator is recalculated at three distinct points in the spectrum. A different correlator is used for the first 1024 channels of the spectrum, the second 1024 channels, etc. If we let N_i be the value of the correlator, then the correlator is defined as follows:

$$N_i = -1 \quad i = 1, k$$

$$N_i = 2 \quad i = k + 1, 2k \qquad (1)$$

$$N_i = -1 \quad i = 2k + 1, 3k$$

where $\sum_{i=1}^{3k} N_i = 0$ and $k = \text{INT}\left(\frac{W(X)}{2.0}\right) * 2 + 1.$

W(X) is the linear width equation evaluated at 1024, 2048, 3072 (i.e., as the search process moves through the spectrum a new correlator is computed at channels 1024, 2048, 3072.) The INT function is a truncation of the floating number $\frac{W(X)}{2.0}$. The minimum value allowed for k is 3.

The correlation function C(X), for channel X is defined as follows:

$$C(X) = \sum_{i=1}^{3k} Y \cdot N_i$$

where Y is the square root of the contents of channel X. N_i is defined as in equation (1).

The value for the correlation function becomes positive and increasing when moving over a peak and is zero or negative when no significant spectral structure exists. The program monitors the changes in the correlation function to determine when a peak has been detected. The sensitivity of the

search is controlled by a sensitivity constant, ISEN. The positive values of the correlation must exceed the value for ISEN in order for a peak to be accepted. The default value for ISEN is 10.

III.3 THE GAUSSIAN FITTING ALGORITHM

Subroutine GAUSS is used to perform a nonlinear least squares fit of the photo peaks to a gaussian function. This routine extracts from file SEARCH the peak locations of the lines to be analyzed, determines fit limits, number of peaks in the fit, and performs a nonlinear least squares fit of a Gaussian function to the data.

The least-squares fitting procedure minimizes the sum of squares of the deviations of a functional form from the data. Let Y(i) represent the data at channel i, F(i) the assumed function to be fit, and U(i) a statistical weighting term. The objective of the procedure is to determine the values of the parameters of F(i) that minimize:

$$R^2 = \sum U(i)\Big(Y(i) - F(i)\Big)^2,$$

where the indicated summation is over all channels, i, in the region to be fit. The function assumed by the GAUSS programs to fit gamma-ray photopeaks is a gaussian:

$$F(i) = He^{-k(i-C)^2/W^2},$$

where k = 4(ln)2, H is the height of the gaussian, C is the centroid, and W is the full width at half maximum. The GAUSS program will fit up to three simultaneous gaussians if a multiplet fit is required:

$$F(i) = \sum_n H_n e^{-k(i-C_n)^2/W_n^2},$$

but since the generalization to multiple gaussians will become obvious, for the sake of simplicity in notation, the description will deal with the simpler situation.

The gaussian function is nonlinear in the parameters C and W, the centroid and width, so it is necessary to make a linear approximation. Good initial estimates can be made for the parameters, so a Taylor's series expansion to first order terms is used:

$$F(i) \simeq F_0(i) + \frac{\partial F}{\partial H}\bigg|_0 h + \frac{\partial F}{\partial C}\bigg|_0 c + \frac{\partial F}{\partial W}\bigg|_0 w,$$

where lower-case symbols are used to represent differential changes in the corresponding (upper-case) variables.

A linear least-squares procedure (below) will determine the values of h, c and w that minimize R^2. However, because an approximate functional form has been used, it is necessary to determine a new value for $F_o(i)$ using H + h, W + w, etc., and repeat the linear fitting process. The iterations continue until the values of all of the increments are simultaneously less than predetermined convergence criteria (or until a predetermined number of iterations has been made).

The linearized form of F(i) has coefficients that are the derivatives of the original (nonlinear) function:

$$\frac{\partial F}{\partial H} = e^{-K(i-C)^2/W^2}$$

$$\frac{\partial F}{\partial C} = \frac{2Hk(i-C)}{W^2} \frac{\partial F}{\partial H}$$

$$\frac{\partial F}{\partial W} = \frac{(i-C)}{W} \frac{\partial F}{\partial C}$$

The increase of peak width with channel is insignificant over the range of variation permitted C in converging to the final centroid value, so the dependence is not taken into account in the fitting procedure.

Denoting the coefficients as $D_1(i)$, $D_2(i)$, etc., the sum of squares of the deviations is given by:

$$R^2 = \sum U(i) \left(Y(i) - F_o(i) - hD_1(i) - wD_2(i) - cD_3(i) \right)^2$$

and the values of h, c and w that minimize R^2 are the solutions of the set of linear equations:

$$\frac{dR^2}{dh} = 0 \qquad \frac{dR^2}{dc} = 0 \qquad \frac{dR^2}{dw} = 0 \ .$$

Define the vectors P and Q and matrix M:

$$P = h, c, w$$

$$Q_j = \sum U(i) \ (Y(i)-F_o(i)) \ D_j(i)$$

$$M_{jk} = \sum U(i) \ D_j(i) \ D_k(i) \ ,$$

Then the set of linear equations that minimize R^2 are:

$$\sum_k M_{jk} \ P_k = Q_j \ ,$$

with the solution:

$$P_k = \sum_j \ (M^{-1})_{kj} Q_j \ .$$

Now, having the values of h, c and w that minimize R^2 for the initial estimates of H, W, C, the "better" values, H + h, W + w, C + c, are used to re-evaluate the coefficients, and the process is repeated. Provided that the initial estimates are sufficiently close to the "true" values, the iterations eventually converge to those "true" values. In practice, the iterations are stopped when all parameters are simultaneously less than predetermined convergence criteria, or after a preset number of iterations has been performed.

Generalization to multiplet fitting is straightforward: let H_n, C_n and W_n represent the height, centroid and width for the n^{th}peak, and h_n, c_n and w_n the associated linearized (differential) variables. Define the derivative coefficients:

$$D_3 = \frac{\partial F}{\partial H_1} \quad D_6 = \frac{\partial F}{\partial H_2} \quad D_9 = \frac{\partial F}{\partial H_3}$$

$$D_4 = \frac{\partial F}{\partial C_1} \quad D_7 = \frac{\partial F}{\partial C_2} \quad D_{10} = \frac{\partial F}{\partial C_3}$$

$$D_5 = \frac{\partial F}{\partial W_1}$$

and define the vector, P:

$$P = (h_1, c_1, w_1, h_2, c_2, h_3, c_3)$$

Q_j and M_{jk} are defined as above, except that the indices, instead of having values 1 through 3, will have values 1 through 2N+1 for N peaks in the multiplet.

For multiplet fits the widths are not allowed to vary independently for each peak. Each width is varied the same amount, at each iteration step the same increment is added to each peak width.

ESTIMATION OF ERRORS

The estimated variance in the final values of the gaussian parameters P_j, by analogy with linear analysis, are calculated from the elements of the final (inverse) matrix:

$$\sigma_j^2 = \frac{R^2}{m} \left(M^{-1}\right)_{jj} ,$$

where m is the number of degrees of freedom in the analysis (i.e., the number of channels in the fit region reduced by the number of parameters being fit, and R^2 is the final value of the sum of the squares of the residuals. The factor R^2/m is the common reduced χ.

The expected variance in the centroid and width of the gaussian fit to the peak are:

$$\sigma^2(C) = \frac{R^2}{m} (M^{-1})_{2,2} ,$$

$$\sigma^2(W) = \frac{R^2}{m} (M^{-1})_{3,3} .$$

Because the area is derived from two parameters, the gaussian width and height, its error includes a covariance term:

$$\sigma^2(\text{area}) = \frac{R^2}{m} \frac{\pi}{k} \left[H^2(M^{-1})_{33} + W^2(M^{-1})_{11} + 2HW(M^{-1})_{13} \right] . \quad (1)$$

If the fitting includes multiple gaussians, the estimated errors are derived from the appropriate 3x3 minors of the inverse matrix.

In practice, because the analysis does not meet all of the requirements for χ evaluation, in order to assure that estimated errors are not unrealistically low, if $\chi = R^2/m$ is less than one, it is set equal to one for generation of the error values.

PEAK FIT LIMITS

The determination of fit limits and background values for a normal gaussian fit are done automatically. This process involves the use of a linear width function. This function describes the widths of peaks as a function of channel number. The coefficients of this function are established either by the NOVA computer and passed in the header information for the spectrum being analyzed, or by an ECLIPSE energy calibration routine which calibrates the spectrum prior to the execution of the GAUSS fitting process. This width function is denoted by W(X) where X indicates the channel number at which the value of the width equation is computed. The following diagram illustrates how the various fit limits are determined.

X = Integral peak position as determined by search
LUL = Left lower limit for background determination
LFL = Left fit limit
RFL = Right fit limit
RUL = Right upper limit for background determination

The location of these values is determined as follows.

$$LFL = X - [W(X) + .5]$$
$$RLF = X + [W(X) + 1.7]$$
$$LUL = X - [W(X) * 6.0 + .5]$$
$$RUL = X + [W(X) * 6.0]$$

For multiplet fits the left most peak position is used for the determination of LFL and LUL. The right most peak is used for the determination of RFL and RUL.

BACKGROUND DETERMINATION

The background under the peak is assumed to be a straight line drawn between a point on the high side of the peak to a point on the low side of the peak. These high and low side points are determined by computing the lowest 3 or 5 point average of the contents of the channels between RFL and RUL; LUL and LFL; respectively (see Figure 5). A 3 point average is used for the first 1024 channels of the spectrum and a 5 point average for the remaining channels.

MULTIPLET DETERMINATION

The determination of whether a peak should be fit as a doublet with adjacent peaks is a function of how close the adjacent peak is to the peak which is to be fit.

Let X_i be the peak extracted from the SEARCH file. Let X_{i+1} be the position of the next peak in the SEARCH file.

If $| X_i - X_{i+1} | < W(X_i) * 2.5 - .5$ then X_i and X_{i+1} are fit as a doublet. If X_i and X_{i+1} are to be fit as a doublet, then a check is made between X_{i+1} and X_{i+2} using the same distance criteria to determine if the set of peaks X_i, X_{i+1}, and X_{i+2} should be fit as a triplet. The process is terminated when 3 peaks have been identified for a multiplet fit.

CONVERGENCE CRITERIA

The nonlinear least-squares fitting process is an iterative process for which new incremental values for the variables X, Y and W are computed at the end of each interation. X, Y and W are the variables for the gaussian position, height, and width, respectively. After each iteration, a check is made to determine whether the change in the variables is within bounds or whether the change is small enough to indicate that the variable has converged.

The boundary limits for the three variables are given in the following table. If any one of the variables exceed its boundary limit, the fitting process is terminated and the peak position for this unfittable peak is written to file EXCESS.

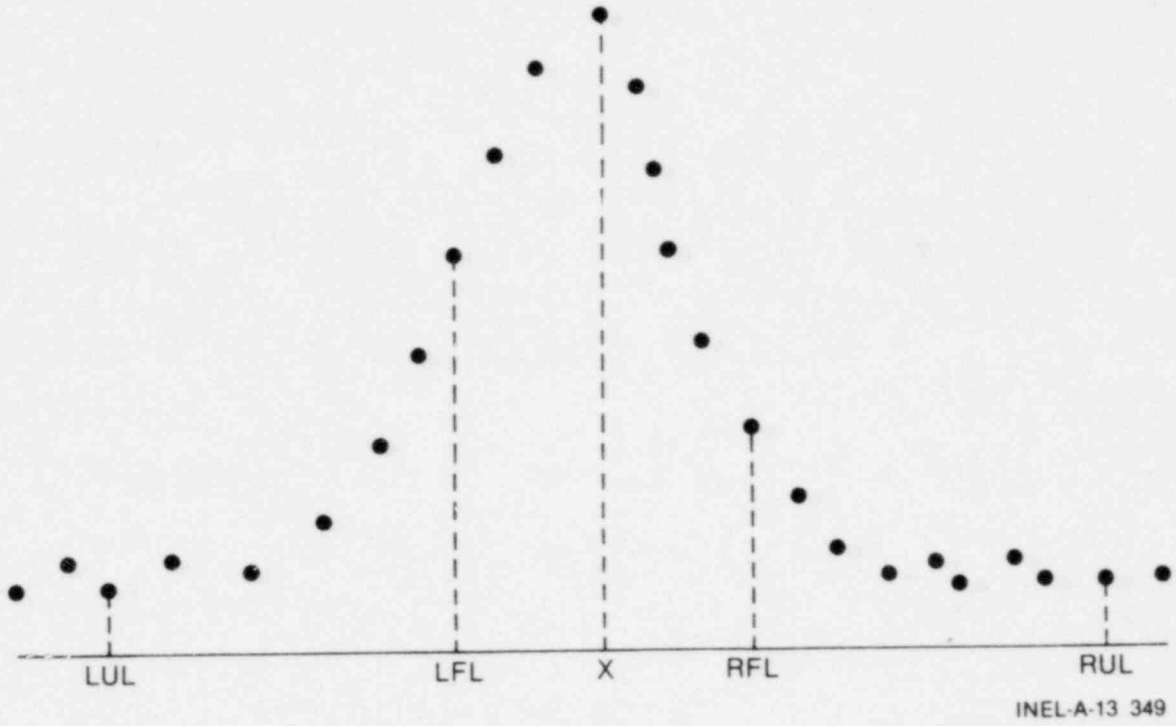


Figure 5. Fit Limits and Background Determination

VARIABLE	LOWER BOUNDARY	UPPER BOUNDARY
X	lower fit channel	Upper fit channel
Y	the initial Y value divided by 8	the initial Y value multiplied by 8
W	W(X)/10.0	W(X) 10

The criteria used to indicate whether a variable has converged is given below. The fitting process terminates when all three variables have converged simultaneously. The results obtained from the fit are then used to compute the area of the peak and its energy.

VARIABLE	CONVERGED CHANGE
X	< .005
Y	<.5% of the previous Y value
W	< .005

AREA AND ENERGY DETERMINATION

The primary function of the fitting process is to compute the area under photopeaks. The area A in the ECLIPSE system is computed directly from gaussian height Y and width W.

$$A = W * Y * 1.06446 * SC$$

The variable SC is the random coincidence summing correction for the spectrum:

$$SC = (T * .0133)/(LT * 10000.0) + 1$$

where T is the total counts in the spectrum and LT is the live time.

The percent error in the area is computed from the error in the area, (area), given in equation 1, ESTIMATION OF ERRORS. It is defined to be:

$$\% (A) = (area)/A$$

The nonlinearity correction for each of the two detector systems is represented by functions given in tabular form. The GAUSS routine selects the proper table for the spectrum being analyzed depending upon the spectrometer system which was used to collect the data. For an explanation of how these tables are used, see Reference 1. The energy of each analyzed peak is a function of its position, X, the nonlinearity correction, L(X), and the energy equation parameters. Let $\bar{z}$ be the zero intercept of the linear energy equation and a, b coefficients of the energy equation. Then the assigned energy, E(X), of the analyzed peak is

$$E(X) = a * (L (X + \bar{z}) + (X + \bar{z})) + b$$

III.4 ECLIPSE LIMIT CALCULATION

The limit calculations are performed by two subroutines:

(1) LIMFIT is the main and controlling routine for the limit calculation process. It performs limit calculations on peaks for which there are no interference peaks which have already been fit by the GAUSS subroutine.
(2) LIMIT is a routine which is called by LIMFIT when a doublet limit calculation is required.

The limit calculations are performed on two sets of peaks. The first set is the peaks which were found by SEARCH and which GAUSS was unable to fit. The locations of these peaks are in a file called EXCESS and are placed there by GAUSS. The other set of peaks is identified by a list of energies in the file LIMLIB. This list of energies is a subset of the energies of lines contained in the master isotope library. The list of energies is converted to channel numbers based on the current energy calibration for the spectrum being analyzed after the peak locations have been identified. Limit calculations are performed in the same manner for peaks identified by either source.

The maximum number of lines contained in the limit library is 511. The maximum number of peaks contained in the EXCESS file is 127.

III.4.1 The LIMFIT Routine

LIMFIT extracts the channel numbers from each of the two files EXCESS and LIMLIB and determines the limits over which the peak is to be fit. These fit limits are obtained from the locations of peaks which are in file GAUSS (the peaks which were found by the SEARCH routine and fit by the GAUSS routine). Given the location of the peak, X_P, for which a limit is to be calculated, the two adjacent peaks X_L and X_H are located in file GAUSS such that $X_L \leq X_P \leq X_H$. Tests are then made between X_L, X_P and X_H to determine where the fit limits on X_P should be placed and whether a doublet limit fit should be made. The following table describes the tests which are made and the resultant fit limits if the peak X_P passes that test. The tests are performed in the order described. If a peak fails to pass any of the tests, it is fit as if it passed test 1. A flow chart of the operations performed in the limit procedure is shown in Figure 6.

TABLE 1 Fit Limits for Limit Calculations

TEST ORDER	TEST	RESULTANT FIT LIMITS
1	$X_P < X_H - 5 * W(X_H)$ $X_P > X_L + 5 * W(X_L)$	$LF = X_P - 2.0 * W(X_P)$ $UF = X_P + 2.0 * W(X_P)$
2	$X_P < X_H - 3 * W(X_H)$ $X_P > X_L + 5 * W(X_L)$	$LF = X_P - 3.0 * W(X_P)$ $UF = X_P + 1.0 * W(X_P)$
3	$X_P < X_H - 5 * W(X_H)$ $X_P > X_L + 2.5 * W(X_L)$	$LF = X_P - 1.0 * W(X_P)$ $UF = X_P + 3.0 * W(X_P)$
4	$\lvert E(X_P) - E(X_H) < 1 \rvert$	LIMIT not calculated
5	$\lvert E(X_P) - E(X_L) < 1 \rvert$	LIMIT not calculated
6	$X_P - X_L = X_H - X_P$	$LF = X_P - 2.0 * W(X_P)$ $UF = X_P + 2.0 * W(X_P)$
7	$X_P - X_L < X_H - X_P$	DOUBLET LIMIT FIT $LF = X_L - [W(X_L) + 0.5]$ $UF = X_P + [W(X_P) + 1.7]$
8	$X_H - X_P < X_P - X_L$	DOUBLET LIMIT FIT $LF = X_P - [W(X_H) + 0.5]$ $UF = X_H + [W(X_P) + 1.7]$

$W(X_i)$ = value of width equation at position X_i.
LF = Lower fit limit
UF = Upper fit limit
$E(X_i)$ = value of energy equation at position X_i.

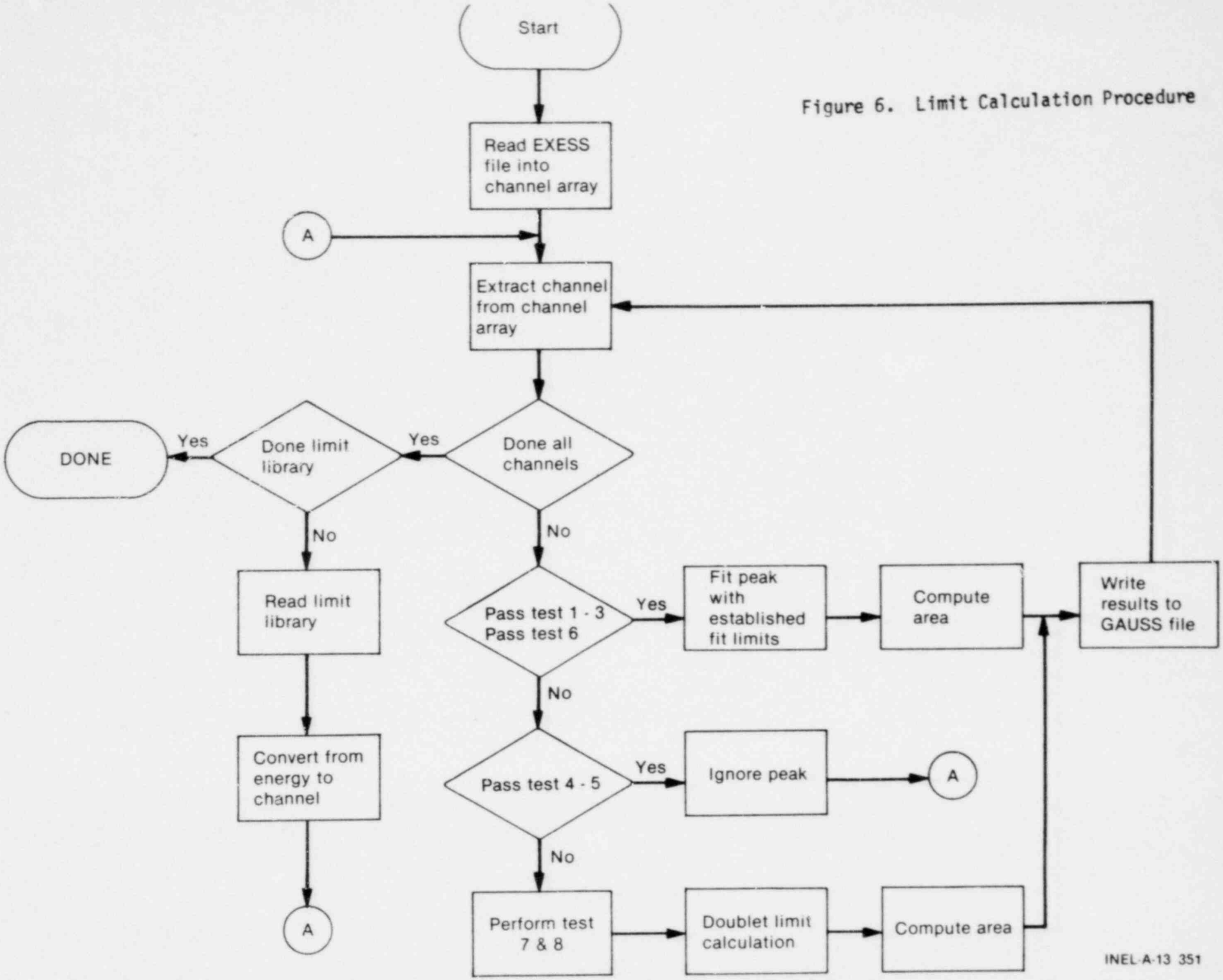


Figure 6. Limit Calculation Procedure

Once the peak positions and fit limits have been determined, the assumption is made that the peak, if any, is represented by a small gaussian and a linear background.

Let

y_i = contents of channel i

$\sigma_i = y_i$ = uncertainty in y_i

$g_i = -\left(\frac{X_i - X}{W(X)/k}\right)$ where X is the gaussian center

W(.) is the width of the gaussian at X and

$k = (4 * \ln 2)^{1/2}$

$\bar{Y}_i = a + bx_i + Yg_i$ where a and b are the coefficients of the linear background line and Y is the height of the peak.

Then values for a, b and Y are found that minimize the equation

$$R = \sum_{i=1}^{n}\left(\frac{y_i - \bar{Y}_i}{\sigma_i}\right)^2$$ where n is the number of data channels between the fit limits.

At the minimum we have the set of equations

$$\frac{\partial R}{\partial a} = -2 \sum \frac{1}{\sigma_i^2} (y_i - a - bx_i - Yg_i) = 0$$

$$\frac{\partial R}{\partial b} = -2 \sum \frac{X_i}{\sigma_i^2} (y_i - a - bx_i - Yg_i) = 0$$

$$\frac{\partial R}{\partial y} = -2 \sum \frac{g_i}{\sigma_i^2} (y_i - a - bx_i - Yg_i) = 0$$

If M is the matrix of coefficients from the above set of equations and $\vec{\nabla}$ the vector of coefficients of the constant terms we solve for the vector $\vec{P}$ such that

$$M \cdot \vec{P} = \vec{\nabla}$$
$$\vec{P} = \vec{\nabla} \cdot M^{-1}$$

the terms of the vector P then contains the results for the variables a, b, Y. The error $\sigma(y)$ in the height of the gaussian Y is determined from the larger of

$$\sqrt{M^{-1}_{3,3}} \quad \text{or} \quad \sqrt{M^{-1}_{3,3} \cdot \frac{\chi^2}{n-3}}$$

where

$$\chi^2 = \sum \left(\frac{y_i - \bar{y}_i}{\sigma_i} \right)^2$$

The area A of the peak is then determined from

$A = 1.06 \cdot W(X) \cdot Y \cdot C$ where $W(X)$ is the value of the width equation at X and C is the random coincidence summing correction. The error in the area, $\sigma(A)$, is determined from

$$\sigma(A) = \chi \cdot \sqrt{M^{-1}_{3,3}} \cdot 1.06 \cdot C$$

The value of the peak area which is given to subsequent analysis routines is either an upper limit, UL, or a true area depending upon the relationship between A, and $\sigma(A)$.

If $A > 2 * \sigma(A)$ then the reported area is A. (1)

If $A < 2 * \sigma(A)$ then the area is an upper limit, where (2)

$$UL = \begin{cases} A + 2.0 * \sigma(A) & \text{if } A \geq 0 \\ 2.0 * \sigma(A) & \text{if } A < 0 \end{cases}$$

If the value for a peak has been determined to be an upper limit value, the error in the area $\sigma(A) = -\sigma(A)$, the error will be reported as negative. This negative value is used only as a flag to the operator that the associated area is an upper limit.

When limit calculations are performed on peaks whose areas are less than 150 counts, a change is made in the fitting algorithm used. When the y_i values in the fitting region are small, the approximation $\sigma_i = \overline{y_i}$ provides a poor estimate of the actual standard deviation of y_i. Since σ_i is used as a weighting factor in the non-linear least squares fit, poor estimates of σ_i cause inappropriate weights for the data points during the fitting process. To avoid this error, if a first pass limit calculation (performed as previously described) results in an area of less than 150 counts, a second pass limit calculation is performed using a five channel smoothed weighting factor $\overline{\sigma}_i$ computed as:

$$\overline{\sigma}_i = \left(\frac{y_{i-2} + 2y_{i-1} + 3y_i + 2y_{i+1} + y_{i+2}}{9} \right)^{1/2} .$$

III.4.2 DOUBLET LIMIT CALCULATION

If a limit peak passes test 7 or 8 as described in Table 1, then a doublet limit calculation is performed on the limit peak X_p and the interference peak X_I. The interference peak was a peak which was found in the GAUSS file and is the peak which was used to satisfy test 7 or 8.

An attempt is then made to fit the two peaks to two gaussians in the same manner as a doublet fit is made in the GAUSS routine. (See Section III.2) The initial peak positions for the two peaks are X_p and X_I respectively with an initial width for both peaks which was previously computed by the GAUSS routine for peak X_I. If the fit is successful, a value for the area of the limit peak is computed as described in the following paragraph. If the fit is not successful, (that is, one of the parameters exceeds its limit value or the fitting process is unable to converge) a linear least-square refit of the data is performed to determine the heights of the gaussian functions whose position and width are fixed at the initial values used to begin the non-linear multiplet fit.

The criteria used to establish whether the resultant areas calculated from a multiplet limit fit are different from the singlet limit fit calculation. Let A be the area of the peak, and $\sigma(A)$ the error in the area. The following relationships are used to determine whether the area is a real area or an upper limit:

(1) If $A < 2\ \sigma(A)$ the area is a limit value, LV, where

$$LV = \begin{cases} A + 2\ \sigma(A) & \text{if } A \geq 0 \\ 2\ \sigma(A) & \text{if } A < 0 \end{cases}$$

(2) If $A \geq 3\ \sigma(A)$ the area is real value and the value for A is used for subsequent analysis

(3) If $2\ \sigma(A) \leq A < 3\sigma\ (A)$ the area is a limit value, LV, where

$$LV = \begin{cases} A + 3\ \sigma(A) & \text{if } A \geq 0 \\ 3\ \sigma(A) & \text{if } A < 0 \end{cases}$$

III.5 ISOTOPIC IDENTIFICATION

Subroutine ISOLK makes isotopic assignments to the analyzed lines in file GAUSS. An isotopic assignment is made if the energy of the analyzed line meets specified requirements with the energies located in the isotope library file. An entry is made in file ISOP for every line which has been identified as belonging to a particular isotope. For the lines which cannot be identified, an entry is made in file UNTAG. The file ISOP is then passed to the background subtraction subroutine, the next step in the analysis procedure. Each entry in file ISOP contains the following information:

(a) Isotope name

(b) Type of analysis flag

(c) Atomic number and mass of the assigned isotope

(d) Calculated decay correction divided by the volume of the sample

(e) The energy of the line

(f) Isotope branching ratio

(g) Error in isotopic branching ratio

(h) The intensity of the line (area/live time)

(i) Percent error in the intensity

(j) The detector efficiency at this energy

(k) Error in energy

(l) Isotopic line predominance

(m) isotope half life

The master isotope library from which identifications are made is a collection of up to 900 gamma-ray lines. Each line in the library contains:

(1) Isotope name

(2) Atomic number and mass number

(3) Branching ratio

(4) Error in branching ratio

(5) Line predominance

(6) Half life

The line predominance is an arbitrary integer which indicates the "worth" of this line compared to the other lines for this isotope. The smaller the predominance the more likely that line will show up in the spectrum if that particular isotope is present. No restriction is made on how many lines (library entries) exist for each isotope. Given the energy of an analyzed line extracted from file GAUSS, a binary search is made of the isotopic library to determine which three entries in the library are closest to the analyzed line energy. The energies of these three isotope library entries are compared to the analyzed line energy to determine if any are less than 1 keV. A nuclide assignment is made for every entry whose difference is less than 1 keV. It is possible, therefore, that one analyzed line could have up to three isotopic identifications. If none of the library entries are within 1 keV of the analyzed line, no isotopic assignment is made and all information for that line is moved from file GAUSS to file UNTAG.

DECAY CORRECTION

The following variables are used in the decay correction algorithm routine ISOLK:

$T_{1/2}$ = the half life of the assigned isotope

TCAY = the difference between the counting time and sampling time in seconds

RT = the spectrum real

ST = sampling time of sample from which the spectrum was collected

$\lambda = .693147/T_{1/2}$

$$E1 = \begin{cases} 1 - (\lambda * TCAY) & \text{if } \lambda * TCAY \leq .0001 \\ e^{-(\lambda * TCAY)} & \text{if } \lambda * TCAY > .0001 \end{cases}$$

$$E2 = \begin{cases} \lambda * RT & \text{if } \lambda * RT \leq .0001 \\ 1 - e^{-(\lambda * RT)} & \text{if } \lambda * RT > .0001 \end{cases}$$

$$C1 = \begin{cases} (\lambda * RT)/(E * E2) & \text{if } RT > .1 * T_{1/2} \\ e^{(TCAY*)} & \text{if } RT < .1 * T_{1/2} \end{cases}$$

$$C2 = \begin{cases} (\lambda * ST)/[1.0 - e^{-(\lambda * ST)}] & \text{if } ST > 0.0 \\ 1.0 & \text{if } ST = 0.0 \text{ or } (ST < .1 * T_{1/2}) \end{cases}$$

The decay correction DC is then defined as the product of C1 and C2. If DC is greater than 1000.0, the isotope is assumed not to be present and the isotope assignment is ignored. Moreover, if λ*TCAY is greater than 150, the isotope assignment is also ignored.

III.6 BACKGROUND CORRECTION ALGORITHM

Three files are needed to do background corrections - two input files and one output file. The input file is generally the output of the isotope tagging routine. Conceptually, the peak count rates in the first input file are subtracted from the peak count rates in the second input file and the results are written to the output file. Or, in general terms, we could describe the process as follows:

[FILE 2] - [FILE 1] ⟶ [FILE 3]

Thus, FILE 1 must contain the results of the background spectrum analysis, FILE 2 must contain the results of a normal spectrum analysis (i.e., the file to be corrected) and FILE 3 will contain the background-corrected results. Specifically, the procedure works as follows:

1. FILE 1 is scanned once and three arrays are built in-core. The first array contains the energies of the peaks in FILE 1. The corresponding elements of the second array contain the count per second values (CPS_1) of each peak. The third array contains the one-sigma errors (SIG_1) in the count per second values. "Less-than" values are ignored - they cannot be used in a background correction.

2. FILE 2 is next read, one entry† at a time. The energy of the peak is compared with the values in the energy array and, if they agree within 1 keV, a background correction is done, as follows:

$$CPS_3 = CPS_2 - CPS_1$$

$$SIG_3 = \sqrt{(SIG_1)^2 + (SIG_2)^2}$$

$$PER_3 = 100*SIG_3/CPS_3$$

where: CPS is the count/second value
SIG is the one-sigma error in CPS
PER is the one-sigma percent error in CPS
subscripts correspond to the file numbers

The results are then tested. If CPS_3 is less than 2 times SIG_3, then CPS_3 is set equal to 2 times SIG_3, PER_3 is made negative, and a flag bit is set to indicate that CPS_3 is a "less-than" value. Otherwise, CPS_3 is left as it was calculated above. The new results are written to FILE 3(PER_3 is

†NOTE: Each isotope that tags a single gamma peak requires an entry in the file even though their energy, count rate, efficiency, etc. are the same. Thus, up to three entries may exist for a single gamma peak.

made negative so that routines that follow will not have to test individual bits of the flag to determine if a value is "less-than".)

If the peak energy of the entry in FILE 2 does not match an energy in the array, then the data for that peak is copied from FILE 2 to FILE 3 unchanged. Thus, no background correction is done. The same occurs if the FILE 2 entry is a "less-than" value - no background correction is attempted - it is simply copied from FILE 2 to FILE 3.

III.7 INTERFERENCE DECONTAMINATION ALGORITHM

Three files are needed to do interference decontaminations - an input file, a library file and an output file. The input file must contain the results of a normal spectrum analysis which are to be decontaminated. The input file is usually the output of the background correction routine. The library file must contain the interference library. The output file will contain the interference-decontaminated results.

The interference library is essentially a list of the energies and isotope names of gamma lines that are contaminated plus the energies and isotope names of clean lines that are to be used to decontaminate the contaminated lines. For example, a typical library might look like this:

INTERFERENCE DECONTAMINATION LIBRARY

CONTAMINATED LINES:					CLEAN LINES:				
ENERGY	Z	ELE	A	ISOMER	ENERGY	Z	ELE	A	ISOMER
834.827	25	MN	54	NO	2392.020	36	KR	88	NO
772.600	53	I	132	NO	479.510	74	W	187	NO
810.757	27	CO	58	NO	772.600	53	I	132	NO

Consider the first line in the library. Mn^{54} at 834.827 keV is contaminated by Kr^{88}. To decontaminate it, the Kr^{88} clean line at 2392.020 keV will be used. The actual calculation done will be described in detail below.

The decontamination algorithm allows both multiple and indirect decontaminations. In a multiple decontamination, a single peak is decontaminated of more than one isotope. In an indirect decontamination, a contaminated peak is first decontaminated and the result used to decontaminate another peak. Multiple and indirect decontaminations may be combined. They will be done in the order that they are listed in the interference library.

The number of decontaminations done in a multiple decontamination is limited by the maximum number of isotope tags that a peak can receive in the tagging routine. Since the maximum number of tags is usually three, the number of decontaminations that can be done is limited to two.

In practice, almost any number of indirect decontaminations can be done; that is, isotope A can be used to decontaminate isotope B, which can be used to decontaminate isotope C, which can be used to decontaminate isotope D, etc. The only limit here is the library size. The library, as it is currently structured, has room for a maximum of 21 entries. Each single decontamination or each step in a multiple or indirect decontamination requires one entry in the library.

It should be obvious from the above that the order of the entries in the interference library is critical for multiple or indirect decontaminations. The individual steps in a multiple or indirect decontamination will be done in the order that they are listed in the library.

The actual calculations done in an interference decontamination are as follows. In all of these calculations, when two subscripts are shown, the first refers to the isotope and the second refers to the energy:

The disintegration rate is calculated for the clean isotope from its clean line. Thus,

$$DPS_{CLN} = \frac{CPS_{CLN,CLN}}{EFF_{CLN} * BR_{CLN,CLN}}$$

where: DPS_{CLN} = the disintegration rate (d/sec) of the clean isotope.

$CPS_{CLN,CLN}$ = the count rate (counts/sec) of the clean isotope at the clean line energy.

EFF_{CLN} = the efficiency at the clean line energy

$BR_{CLN,CLN}$ = the branching ratio fractional, not percent) of the clean isotope at the clean line energy.

The effective count rate and one-sigma error for the clean isotope at the contaminated line energy are calculated:

$$CPS_{CLN,CON} = DPS_{CLN} * BR_{CLN,CON} * EFF_{CON}$$

where: $CPS_{CLN,CON}$ = the effective count rate (counts/sec) of the clean isotope at the contaminated line energy.

$BR_{CLN,CON}$ = the branching ratio of the clean isotope at the contaminated line energy.

EFF_{CON} = the efficiency at the contaminated line energy.

$$SIG_{CLN,CON} = \frac{CPS_{CLN,CON} * PER_{CLN,CLN}}{100}$$

where: $SIG_{CLN,CON}$ = the one-sigma error in $CPS_{CLN,CON}$

$PER_{CLN,CLN}$ = the one-sigma percent error in $CPS_{CLN,CLN}$

The uncorrected one-sigma error in the count rate of the contaminated isotope at the contaminated line energy is calculated:

$$SIG_{CON,CON} = \frac{CPS_{CON,CON} * PER_{CON,CON}}{100}$$

where:

$CPS_{CON,CON}$ = the uncorrected count rate of the contaminated isotope at the contaminated line energy.

$SIG_{CON,CON}$ = the one-sigma error in $CPS_{CON,CON}$

$PER_{CON,CON}$ = the one-sigma percent error in $CPS_{CON,CON}$

The desired net results are then calculated:

$$CPS_{NET} = CPS_{CON,CON} - CPS_{CLN,CON}$$

$$SIG_{NET} = \sqrt{(SIG_{CON,CON})^2 + (SIG_{CLN,CON})^2}$$

$$PER_{NET} = \frac{100 * SIG_{NET}}{CPS_{NET}}$$

A final test is made. If CPS_{NET} is less than $2*SIG_{NET}$, then CPS_{NET} is set equal to $2*SIG_{NET}$, PER_{NET} is made negative and a flag bit is set to indicate that the result is a "less-than" value. (PER_{NET} is made negative so that routines that follow will not have to test individual bits of a flag to determine if a result is a "less-than" value.)

The preceeding discussion and calculations serve as an introduction to the actual decontamination algorithm, which follows:

1. Read the interference library into core.

2. Scan the input file, one entry (note 1) at a time, and save any data in-core that will be needed to do a decontamination later on. There are four possible outcomes of this first scan of the input file. They are:

   a. The entry from the input file is a "less-than" value or it does not correspond to any entry in the library file. Thus, it does not need to be decontaminated, nor does it need to be used in a decontamination.

      ACTION TAKEN: None. The input file entry is ignored.

b. The entry from the input file corresponds to an energy and isotope in the library file that must be decontaminated but will not itself be used in an indirect decontamination. (Energies must agree within 1 keV.)

ACTION TAKEN: None. All data for the contaminated line will be available when the actual decontamination is done in the second scan of the input file (Step 3 below).

c. The entry from the input file is listed as a contaminating isotope at a contaminated energy in the library file, even if it must itself be decontaminated first.

ACTION TAKEN: Save the branching ratio ($BR_{CLN,CON}$).

d. The entry from the input file is listed as a clean isotope at a clean energy in the library file, even if it must itself be decontaminated first.

ACTION TAKEN: Calculate its disintegration rate (DPS_{CLN}) from the count rate, the efficiency and the branching ratio and save it. Also save the percent error in the count rate ($PER_{CLN,CLN}$).

3. Scan the input file a second time, one entry at a time, and do the required decontaminations using the data saved in the first scan. Write the results to the output file. There are five possible outcomes of this second scan of the input file. They are:

a. The entry from the input file is a "less-than" value or it does not correspond to any entry in the library file. Thus, it does not need to be decontaminated, nor does it need to be used in a decontamination.

ACTION TAKEN: Write the entry from the input file to the output file unchanged.

b. The entry from the input file corresponds to a contaminated energy but <u>not</u> to its contaminated isotope in the library file.

ACTION TAKEN: Ignore the input file entry. Do <u>not</u> write it to the output file (note 2).

c. The entry from the input file corresponds to an energy and isotope in the library file that must be decontaminated.

ACTION TAKEN: Perform the decontamination using the data in the input file entry plus the data for the contaminating isotope saved in steps 2.c. and 2.d. Set appropriate flag bits and write the results to the output file. Next, determine if this line is to be used later as a clean line in another decontamination. If so, then replace the disintegration rate (DIS_{CLN}) and percent error ($PER_{CLN,CLN}$) saved in step 2.d. with the newly calculated values.

d. The entry from the input file corresponds to a clean energy and isotope in the library file.

ACTION TAKEN; Set the appropriate flag bit to indicate that the line is clean and write the entry to the output file.

e. The entry from the input file corresponds to a clean energy but <u>not</u> to its clean isotope in the library file.

ACTION TAKEN: Ignore the input file entry. Do <u>not</u> write it to the output file (note 2).

NOTES:

1. Each isotope that tags a single gamma peak requires an entry in the file even though their energy, count rate, efficiency, etc. are the same. Thus, up to three entries may exist for a single gamma peak.

2. This algorithm operates under the assumption that a gamma line listed in the interference library is completely defined in the library. Thus, if a contaminated line is listed in the library as having one or more specific contaminating isotopes, then any other isotopes, tagging the same line, are assumed to be invalid and are discarded. By the same token, if a clean line has other isotope tags that are not specifically listed as contaminants in the library, then they are assumed to be invalid tags and are discarded.

Finally, once a line has been decontaminated, the valid entry corresponding to the contaminating isotope at the contaminated energy need no longer be saved and can be discarded.

III.8 "CLEAN-UP" ALGORITHM

The "CLEAN-UP" routine performs certain necessary housekeeping functions in preparation for the peak resolution routine. Two files are needed - an input file and an output file. The input file is usually the output of the interference decontamination routine. The results of the "CLEAN-UP" operation will be written to the output file.

The "CLEAN-UP" procedure works as follows:

1. Scan the input file and save, in-core, the names of the isotopes whose primary (predominance = 1) line is present. Do not save the isotope names if the percent error in the count rate is greater than 100% or if it is less than 0% (less than 0% indicates a "less-than" value).

2. Scan the input file a second time. For each entry in the file, make the following tests and perform the specified action:

   a. If the percent error in the count rate is greater than 100%, then ignore the entry. Do <u>not</u> write it to the output file.

   b. If the percent error in the count rate is less than 0%, indicating a "less-than" value, then write the entry, unchanged, to the output file.

   c. If the entry was interference-decontaminated or if it was flagged as "CLEAN" by the interference decontamination routine, then write the entry, unchanged, to the output file.

   d. If the entry's isotope name is in the list obtained in step 1 (that is, if the primary or clean line is present), then write the entry, unchanged, to the output file.

   e. If the entry does not pass any of the above tests, then convert the count rate to a "less-than" value as follows:

$$CPS_n = CPS_o + 2*CPS_o*PER_o/100$$

$$PER_n = - PER_o$$

where:

CPS_n = the new count rate

CPS_o = the old count rate

PER_n = the new percent error in the count rate

PER_o = the old percent error in the count rate

Set the appropriate flag bit to indicate that the count rate was converted to a "less-than" value and write the new data to the output file. The percent error is made negative so that routines that follow will not have to test individual bits of the flag to determine if the value is "less-than".

III.9 PEAK RESOLUTION ALGORITHM

The peak resolution routine resolves multiple gamma lines for each isotope into a single weighted average. An outlier test is performed for each isotope, and gamma lines which do not pass the outlier test are not used in the weighted average.

Two files are needed to do the peak resolutions - an input file and an output file. The input file is the output of the "CLEAN-UP" routine. Results of the peak resolutions are written to the output file.

Two different types of results could have been calculated for the isotopic lines which are in the input file. The first is a "REAL" result, that is, a result which was determined not to be a minimum detectable value. The other type of result is a "LESS THAN" result, which is a minimum detectable limit value for that isotope.

Specifically, the resolution procedure works as follows:

1. Scan the source file and build a table of isotope identifiers in core. Each time a new isotope is found, insert its name into the proper position in the table, sorting on the keys Z, A, and IS, where:

   Z = atomic number

   A = mass number

   IS = isomer tag (1 for isomer, 0 otherwise)

2. For each isotope in the table created above, scan the source file once, loading the isotope data into core in order of increasing energy. After all data for the isotope has been loaded, the following tests and calculations are performed in the order listed.

   If a result passes a test it will be reported as the final value, subsequent tests will not be made:

   a. If all results are "LESS-THAN," report the smallest.

   b. If there is only one "REAL" result (i.e., not a "less-than" result), report it. It must be either "CLEAN" or "PRIMARY" since CLNUP would have converted it to a "LESS-THAN" if it were not.

   c. If there are two or more "REAL" results, then perform the weighted-average and outlier test procedure, as follows:

WEIGHTED AVERAGE AND OUTLIER TEST PROCEDURE

1. For N "REAL" results, calculate the weighted average and net one-sigma error:

$$\bar{X} = \frac{\sum_{i=1}^{N} W_i X_i}{\sum_{i=1}^{N} W_i} \qquad \text{(weighted average)}$$

$$\bar{\sigma} = \frac{\sum_{i=1}^{N} W_i^2 \sigma_i^2}{\sum_{i=1}^{N} W_i} \qquad \text{(net one-sigma error)}$$

where: X_i = disintegrations/second of the i^{th} line.

σ_i = one-sigma error in X_i

$$W_i = \frac{1}{\sigma_i^2 \times IC_i}$$

IC_i = intensity code of the i^{th} line.

2. Next, calculate the reduced chi-square as an outlier criterion, ε^2:

$$\varepsilon^2 = \frac{1}{N-1} \sum_{i=1}^{N} \left(\frac{X_i - \bar{X}}{\sigma_i}\right)^2$$

3. If $\varepsilon^2 \leq 5$, report $\bar{X} \pm \bar{\sigma}$

4. If $\varepsilon^2 > 5$ and $N = 2$, report $\bar{X}_i \pm \bar{\sigma}_i$ for the line with the smaller X_i.

5. If $\varepsilon^2 > 5$ and $N > 2$, discard the line with the largest $\left|\frac{X_i - \overline{X}}{\sigma_i}\right|$, set $N = N - 1$ and go back to step (1).

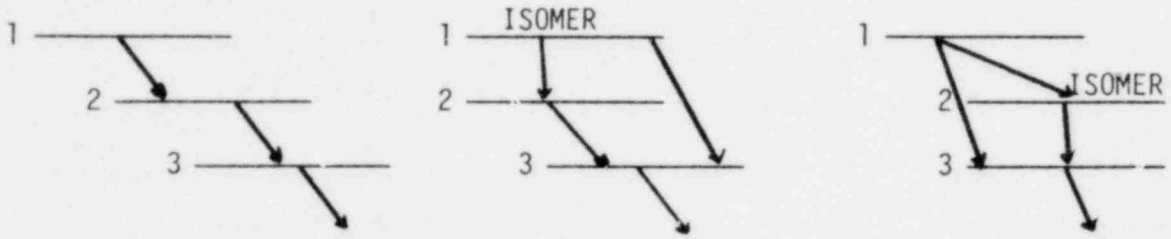
In all of the above tests, appropriate flag bits are set for the individual gamma lines to indicate the type of action taken. Once a final value has been determined, it is written to the output file along with the gamma line data used in the calculations. The procedure then continues with the next isotope until all isotopes have been resolved.

III.10 PARENT-DAUGHTER CORRECTION ALGORITHM

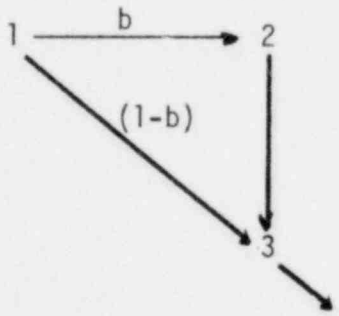
The parent-daughter correction routine corrects isotopic disintegration rates for contributions from parents and grandparents during count and decay times.

Three files are needed to do the corrections - an input file, a library file and an output file. The input file contains the results of the peak resolution routine. The library file contains the parent-daughter correction library. The output file contains all the data in the input file plus results of the parent-daughter corrections.

The parent-daughter correction library contains an entry for each decay chain for which corrections must be done. Each chain can contain up to three members and the members of the chain must undergo beta decay or isomeric transitions - alpha or other decay modes are not allowed. Decay schemes can thus be one of the following:



all of which can be represented by:



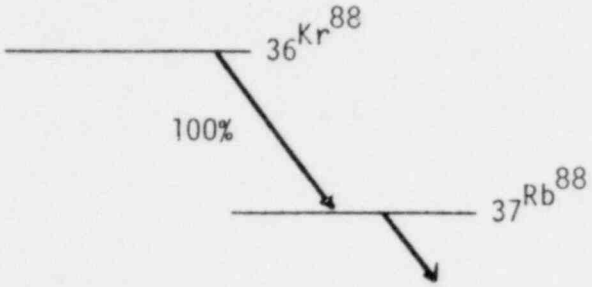
where $0 < b \leq 1$. Thus, for the first scheme, b would be exactly 1. For the second and third schemes, b would be less than 1.

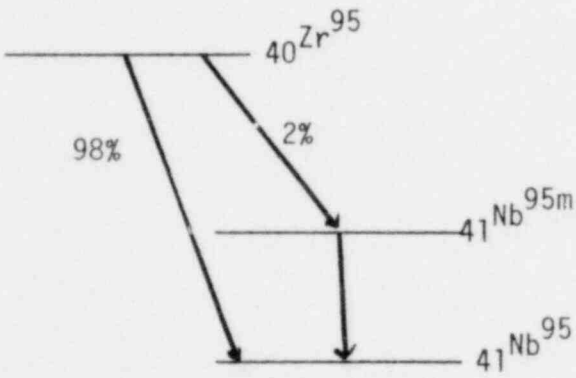
For a two member chain, only members 1 and 2 would be present and b would be exactly 1.

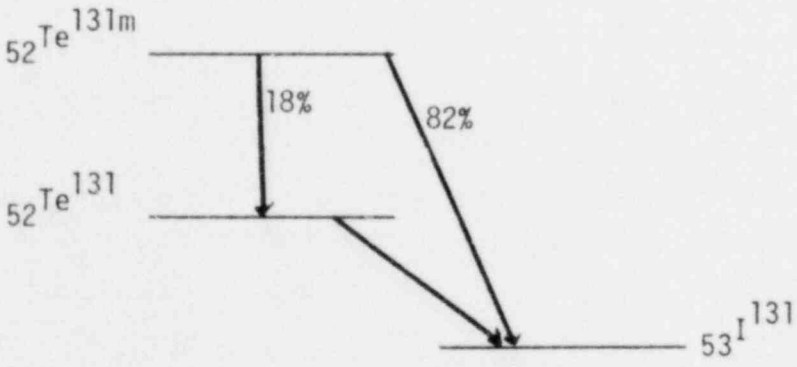
A sample parent-daughter correction library is shown below:

GRANDPARENT ISOTOPE	ISOMER	PARENT ISOTOPE	ISOMER	DAUGHTER ISOTOPE	ISOMER	b
		36 KR 88	NO	37 RB 88	NO	1.000
40 ZR 95	NO	41 NBM 95	YES	41 NB 95	NO	0.020
52 TEM 131	YES	52 TE 131	NO	53 I 131	NO	0.180

which would correspond to the following chains:







The actual calculations used to do the parent-daughter corrections are lengthy and complex and are beyond the scope of this discussion. They are described in detail in references 1 and 2 listed at the end of this section. Only a general overview of the parent-daughter correction algorithm is given here. Specifically, the procedure works as follows:

1. Read the library file into core.

2. For each chain (i.e., each entry) in the library, scan the input file and read into core the data for each member of the chain needed to do the correction.

   Do the parent-daughter correction for the chain and save the results in core. If data for a member of the chain is not present in the input file, then its contribution will be assumed negligible; that is, it will not be used in the correction.

   As each correction is done, update the parent-daughter correction flag. This flag will indicate what was done in the correction (e.g., which members of the chain were present, how many members should have been present and whether a correction could be done). Note that this flag is a separate flag used only for parent-daughter corrections. Unlike the flag used by other analysis routines, there is only one parent-daughter correction flag for each isotope. The other routines use a flag for each gamma line.

3. After corrections have been done for all chains in the library file, copy the input file to the output file. For each isotope that was parent-daughter corrected, replace the uncorrected weighted average and percent error with the corrected values. Also, write the parent-daughter correction flag to the output file.

REFERENCES

1. Helmer, R. G., "Parent-Daughter Activity Relations from a Single Spectrum," 10/1/75 (internal paper).

2. Hartwell, J. K., "Parent-Daughter-Granddaughter Corrections, Random Thoughts from Helmer's 10/1/75 Notes," 12/8/78 (internal paper).

3. R. G. Helmer and M. H. Putnam, "Gauss V A Computer Program for the Analysis of Gamma-Ray Spectra from Ge(Li) Spectrometers," ANCR 1043 (1972).

4. G. W. Phillips, "Fitting Peaks with Very Low Statistics," Nucl. Instr. and Meth. 153 (1978).

5. N. C. Nyer et al., "Procedures Source Term Measurement Program", TREE-1178 (1977).

APPENDIX A

ECLIPSE COMMAND SUMMARY

The following user commands have been added to the Eclipse system. These are in addition to the set of commands available through Data General's Command Line Interpreter.

COMMAND	FUNCTION	INPUT
BACKGROUND	Analyzes a background spectrum; creates a background file of the analyzed isotopes which will be subtracted from other spectra as appropriate.	Spectrum numbers must have a BK extension.
BUNCH	Performs a batch spectral analysis from energy calibration to Parent-Daughter correction. Creates a final results file for each spectrum.	Background spectrum number and up to 10 spectrum numbers to be analyzed with the one background spectrum.
CARD	Reads card image data written on magtape by the CDC-7600 and produces a file 'IMAGE' on the disc in ASCII code.	Magtape file number
EDITINTLIB	Allows the user to edit the peak interference library.	User is prompted for the required information.
EDITPDLIB	Allows the user to edit the parent-daughter correction library.	User is prompted for the required information.
ETABLE	Allows the user to input an efficiency table through the keyboard. A file is created whose name is the table number. This file is created in directory TABLE.	All information required to construct an efficiency table. The program prompts the user as what parameters are required.
GAMIT	Analyzes the spectrum as follows: 1. Energy Calibrate 2. Peak search 3. Least squares fitting 4. Limit calculation 5. Nuclide identification	Spectrum number

COMMAND	FUNCTION	INPUT
INEFF	Reads efficiency table from magtape.	Magtape file number
LIBRY	Reads isotope library information from file 'IMAGE' and creates a isotope library and a limit library file.	Isotope library file name
LIBPR	Prints isotope library	Library file name
LTABLE	Allows the user to input a linearity table through the keyboard.	All information required to construct a linearity table. The program prompts the user as to what parameters are required.
MAIN	Performs a spectral analysis from energy calibration to Parent-Daughter correction. Creates a final results file for this spectrum.	Background spectrum number and a spectrum to be analyzed number
MTREAD	Reads a spectrum from magtape	Magtape file number
MTWRITE	Writes a spectrum or a series of spectra to magtape.	File number which to begin writing; then a series of spectral numbers.
NBOOT	Transfer the NOVA-1200 program from magtape to the NOVA-1200.	None.
OFF	Stops the printing of information on the line printer.	None.
PRHEAD	Prints spectrum Alpha	
PRINTEFF	Prints contents of an efficiency table.	Table number.
PRINTGAUSS	Prints the information in the 'GAUSS' file.	None.
PRINTINTLB	Prints out the contents of the peak interference library.	None.

COMMAND	FUNCTION	INPUT
PRINTISOP	Prints the information in file 'ISOP'. This file contains the nuclide identification and analysis results from the previous analysis.	None.
PRINTPDLIB	Prints out the contents of the Parent-Daughter correction library.	None.
PRINTSUMRY	Prints out the final results of a previous gauss analysis which were saved by program MAIN.	Spectrum ID (e.g.,79041101)
PRINTUNTAG	Prints the information from file UNTAG. These are lines which were analyzed, but for which no nuclide identification was made.	None.
RDTAPE	Reads spectral data from magtape, but for which there is no spectrum identification number in the alpha header. Data is placed in file YYMMDDSS.	Magnetic tape file number
UGAM	Performs same functions as command GAMIT except the user can input his own energy equation coefficients.	Energy equation coefficients, width equation coefficients, linearity and efficiency table numbers.
UMAIN	Performs same function as command MAIN except user can input his own energy equation.	Same as UGAM.
SPRINT	Prints the channel contents of spectrum.	Spectrum number; beginning and ending channels of print region.
STDMAIN	Performs the same functions as the MAIN command except library STANDARD is used for the isotope library and STDLIM is used for the limit library.	Background spectrum number and a spectrum to be analyzed number

COMMAND	FUNCTION	INPUT
STOP	Releases all open files and halts the ECLIPSE.	None.
THORIUM	Performs a thorium energy calibration and then allows the user to transfer this calibration to another spectrum for analysis in the same manner as command MAIN except a pulser calibration is not performed.	Spectrum number; and if desired a spectrum number to be analyzed and a background spectrum number.

APPENDIX B

HOW TO LOAD AND START THE SYSTEM

To bootstrap the system simply means to load the operating system into core from disk and to start its execution. After this is done, the computer can be said to be running normally.

Bootstrapping is necessary to re-start the system after power-down or to restart it after it has been stopped or "released" while it is still powered-up. Procedures are listed below to cover both possibilities. They apply only to the ECLIPSE system.

PROCEDURE TO BOOTSTRAP THE SYSTEM AFTER POWER DOWN

1. Turn the system console (DECwriter) ON, and set it ON LINE. Set it to the uppercase mode with the ALPHA LOCK key.

2. Turn the ECLIPSE's power switch to ON.

3 Flip the disk power switch (not the LOAD/RUN switch) to ON.

4. When the disk unit LOAD light comes on, flip the LOAD/RUN switch down and wait for the READY light to come on.

5. Set the numbered computer front panel switches on as follows:

   0, 11, 12, 14 and 15 - all up
   1, 2, 3, 4, 5, 6, 7. 8, 9, AND !# _ ALL DOWN

6. Lift the RESET switch and release.

7. Lift the PROGRAM LOAD switch and release.

8. On the console, the prompt "FILE NAME?" will be typed. Answer it with:

   DPØF:SYS3

   (the "O" in DPOF is a zero not the letter "O";
   = RETURN)

9. Enter the date and time when it is requested and wait for the "R" prompt (R=ready).

10. Type in the command:

    GO

After a few seconds, the system will answer with "CURRENT DIRECTORY IS PROGM" and follow it with an "R" prompt.

The system is now ready for use.

PROCEDURE TO BOOTSTRAP THE SYSTEM AFTER IT HAS BEEN RELEASED

1. Flip the disk unit LOAD/RUN switch up and immediately slip it back down. If it was already up, then just flip it down. It is assumed here that the disk power was already on.

2. As soon as the READY light comes on, go to step (5) in procedure above.

APPENDIX C

STOP CODES

STOP CODE	ROUTINE	ERROR DESCRIPTION
1	MAIN	Unable to open spectrum file
2	MAIN	Unable to open SEARCH
3	MAIN	Unable to open GAUSS
4	MAIN	Unable to open EXCESS
5	MAIN	Unable to open ISOP
6	MAIN	Unable to open UNTAG
7	MAIN	Unable to create link to spectrum file
8	MAIN	Unable to open overlay file
9	MAIN	Unable to load overlay OVR00 first time
10	MAIN	Unable to load overlay OVR03
11	MAIN	Unable to open library file
12	MAIN	Unable to create link to background isotope file
13	MAIN	Unable to open interference library INTLIB
14	MAIN	Unable to open EDISOP
15	MAIN	Unable to open background isotope file
16	MAIN	Unable to open interference-corrected isotope file INTCORISOP
17	MAIN	Unable to open BKGCORISOP
18	MAIN	Unable to open file CLNUPISOP
19	MAIN	Unable to open file FINAL
20	MAIN	Unable to load overlay OVR04
21	MAIN	Unable to load overlay OVR10
22	MAIN	Unable to load overlay OVR01

STOP CODE	ROUTINE	ERROR DESCRIPTION
23	MAIN	Unable to load overlay OVR00 second time
24	MAIN	Unable to load overlay OVR11
25	MAIN	Unable to load overlay OVR12
26	MAIN	Unable to open file LIMLIB
27	MAIN	Unable to load overlay OVR02
28	MAIN	Unable to create summary file in directory FINAL
29	MAIN	Unable to link to final summary file
30	MAIN	Unable to open final summary file
31	MAIN	Unable to open file PDLIB
32	MAIN	Unable to read header block of final summary file
33	MAIN	Unable to write header block to final summary file

STOP CODE	ROUTINE	ERROR DESCRIPTION
101	GAUSS	Unable to read calibration block
102	GAUSS	Unable to read search table block
104	GAUSS	First channel of fit region is less than 1
105	GAUSS	Lower background channel limit is less than 1
106	GAUSS	Lower background channel is greater than the first fit channel
107	GAUSS	Upper background channel is less than or equal to last fit channel.
230	WRFIT	Unable to write GAUSS fit block
235	RDFIT	Unable to read GAUSS fit block
240	RDCHS	Unable to read spectral data block
245	LDEXC	Unable to write excess file to disc
246	LDEXC	An attempt was made to load a negative or zero channel value into the excess table
250	IOFIT	Unable to write GAUSS fit block
251	IOFIT	Unable to read GAUSS fit block
301	BACKGROUND	Unable to open spectral data file
302	BACKGROUND	Unable to open file SEARCH
303	BACKGROUND	Unable to open file GAUSS
304	BACKGROUND	Unable to open file EXCESS
305	BACKGROUND	Unable to open file ISOP
306	BACKGROUND	Unable to open file UNTAG

STOP CODE	ROUTINE	ERROR DESCRIPTION
307	BACKGROUND	Unable to create link to background spectrum file
308	BACKGROUND	Unable to create link to background isotope file
309	BACKGROUND	Unable to open background isotope file
310	BACKGROUND	Unable to open master library file
311	BACKGROUND	Unable to open overlay file
312	BACKGROUND	Unable to load overlay OVR00 first time
313	BACKGROUND	Unable to load overlay OVR03
314	BACKGROUND	Unable to create background isotope file
315	BACKGROUND	Unable to open file LIMLIB
316	BACKGROUND	Unable to load overlay OVR10
317	BACKGROUND	Unable to load overlay OVR01
318	BACKGROUND	Unable to load overlay OVR00 second time
319	BACKGROUND	Unable to load overlay OVR02
320	BACKGROUND	Unable to load overlay OVR11
321	BACKGROUND	Unable to load overlay OVR12
325	PRINTISOP	Unable to link to file
326	PRINTISOP	Unable to open file
335	PRINTGAUSS	Unable to open file GAUSS
340	PRGAU	Unable to read from GAUSS-type file
345	RDISO	Unable to read from ISOP-type file
350	WRISO	Unable to write to ISOP-type file
355	EDISO	Unable to read spectrum calibration block

STOP CODE	ROUTINE	ERROR DESCRIPTION
375	PRINTUNTAG	Unable to open file UNTAG
380	PRINL	Unable to read interference library from disc
385	PRINTINTLB	Unable to open file INTLIB
390	EDITINTLIB	Unable to open file INTLIB
391	EDITINTLIB	Unable to read file INTLIB from disc
392	EDITINTLIB	Unable to write file INTLIB to disc before calling PRINL
393	EDITINTLIB	Unable to write file INTLIB to disc before ending program
395	INTFER	Unable to read interference library from disc
400	WRFIN	More than 24 lines for a single isotope
401	WRFIN	Unable to write FINAL-type data to disc
405	RDFIN	Unable to read first two FINAL-type blocks from disc
406	RDFIN	Unable to read third and higher FINAL-type blocks from disc
410	RESOL	Unable to read spectrum alpha ID block
411	RESOL	Unable to read first spectrum data block
412	RESOL	Unable to write new header block to disc
413	RESOL	Unable to close empty file
425	PRFIN	Unable to read header block from disc
430	PDCOR	Unable to read header block from disc
431	PDCOR	Unable to write header block to disc

STOP CODE	ROUTINE	ERROR DESCRIPTION
432	PDCOR	Unable to read Parent-Daughter correction library from disc
435	PRPDL	Unable to read Parent-Daughter correction library from disc
440	PRINTPDLIB	Unable to open file PDLIB
445	PRINTSUMRY	Unable to create link to summary file
446	PRINTSUMRY	Unable to open final summary file
447	PRINTSUMRY	Unable to read header block of final summary file
450	EDITPDLIB	Unable to open file PDLIB
451	EDITPDLIB	Unable to read file PDLIB from disc
452	EDITPDLIB	Unable to write file PDLIB to disc before calling PRPDL
453	EDITPDLIB	Unable to write file PDLIB to disc before ending program
455	GHLF	Unable to read master library file from disc
460	PIHIS	End-of-Month Pulser History File recreation error
461	PIHIS	Unable to write block 0 of Pulser History File to disk
462	PIHIS	Unable to write last block of Pulser History File to disk
463	PRPHS	Unable to read Block 0 of Pulser History From disk
464	PRPHS	Unable to read Blocks 1 and up of Pulser History from disk
465	PRINTPIHIS	Unable to open file PULSERHIST
466	EDITPICRIT	Unable to open file PULSERHIST

STOP CODE	ROUTINE	ERROR DESCRIPTION
467	EDITPICRIT	Unable to read Block 0 of Pulser History File
468	EDITPICRIT	Unable to close, delete, recreate or re-open Pulser History file
469	EDITPICRIT	Unable to write Block 0 of Pulser History File to disk
500	PICLB	Unable to write locations of peaks to file SEARCH
501	PICLB	Unable to write to calibration block
503	PICLB	Unable to read file GAUSS
504	PICLB	Unable to write to calibration block
507	PICLB	Not all pulse peaks were fit
602	ISOLK	Unable to read GAUSS file
603, 604, 605 606, 607	ISOLK	Unable to read a block from library file
608, 609	ISOLK	Unable to write block to ISOP file
610, 611	ISOLK	Unable to write block to UNTAG file
801	LIMFIT	Unable to read spectral calibration block
810	LIMFIT	Unable to read file EXCESS
815	LIMFIT	Unable to read block from file

STOP CODE	ROUTINE	ERROR DESCRIPTION
901	LIMIT	Beginning of fit region is not greater than zero
902	LIMIT	Beginning of fit region is not less than ending fit channel
903	LIMIT	Right background is not greater than ending fit channel
8001	FGET	Unable to read a block of data from spectral file

APPENDIX D

The following table describes commands which are available in the NOVA-1200 software. Many of the commands perform operations on data collected from one detector system. A like command exists for the other detector system. These similar commands differ only in the appendage of a 1 or 2 to the basic command mnemonic. In the following table, those commands which have a dual function are annotated by "command mnemonic" 1(2), (i.e., F1(2) indicates that two commands exist, F1 and F2). F1 operates on the data collected from detector system #1 and F2 operates on the data collected from detector system #2.

COMMAND	FUNCTION	INPUT
A1(2)	Sums the counts in the spectrum from channel 59 through 4096. The sum is printed on the teletype.	None
B1(2)	Constructs an alphanumeric spectrum ID.	The alphanumeric ID. ↑D terminates input.
CFQ	Begins execution of an automatic counting sequence using the sample changer.	None
CLBT1(2)	Energy calibrates the spectrum from a Th source and computes pulser energy equivalents.	None
CPQ1(2)	Changes pulser energy equivalents.	Energy equivalents for the two pulser peaks.
CV1(2)	Changes the contents of a channel in the spectrum.	The channel number then its new contents.
DIL	Lists on the teletype the alphanumeric ID's and the first 20 channels of spectra located on the data link.	Starting slot, Ending slot.
E	Makes a new sample entry in the sample log.	Sample number, sample alpha ID, time at start of sample irradiation, date of irradiation, irradiation duration and volume of sample. ↑D terminates input.

COMMAND	FUNCTION	INPUT
EONE	Erases data collection core area for detector system #1.	None
ETWO	Erases data collection core area for detector system #2.	None
F1(2)	Searches for and performs a nonlinear least-squares fit of a gaussian to all peaks in the spectrum.	None
FORMAT	Formats a new disk in unit #1 and copies the unit #0 data to the newly formatted disk.	Any teletype key when ready to begin procedure.
GAM1(2)	Starts collection of spectral data with the pulser shut off.	None
O1(2)	Writes spectral data to disk.	Disk slot number.
ON1(2)	Starts data accumulation (same as start button on panel).	None
P1(2)	Prints alphanumeric header.	None
PEQ1(2)	Computes pulser energy equivalents based on current energy equation.	None
PI1(2)	Energy calibrates spectrum from in-core pulser data.	None
PRINT1(2)	Prints contents of selected channels.	Starting channel, Ending channel.
PQUE	Prints the current automatic counting sequence.	None

APPENDIX D (continued)

COMMAND	FUNCTION	INPUT
Q	Enters an automatic counting sequence for the sample changer.	Sample number, changer position, distance, and efficiency table. ↑D terminates input.
R1(2)	Reads data from disk into in-core data area.	Slot number.
S	Sets real-time clock.	Day, month, hour and minute.
SAM	Prints the contents of a single entry in the sample log.	Sample number.
SEN	Sets search sensitivity level for peak detection.	Sensitivity (an integer).
SRH1(2)	Searches and analyzes data between limits entered by the user. (Same algorithm as F1(2) command).	Starting channel, Ending channel.
TE1(2)	Transfers data to ECLIPSE computer.	None
U	Prints selected groups or all entries in the sample log.	Group number, or just a ↵ to print all entries.
ULOG	Prints the contents of the sample log for those samples which have been acquired after a cut-off date.	A cut-off date.
V1(2)	Allows user to enter energy scale coefficients for the spectrum. Energy equation is of the form $E=a+bx+cx^2$ where x is the channel number.	Values for a, b and c.

COMMAND	FUNCTION	INPUT
W1(2)	Allows user to enter width equation coefficients for the spectrum. Width equation is of the form W=a+bx where x is the channel number.	Values for a and b.
ZSG	Clears the interface register between the computer and the sample changer. Must be issued if sample changer is to be operated in the manual mode.	None
ZSL	Erases all entries in the sample log.	Appropriate responses to computer inquiries.

APPENDIX E

LOADING THE NOVA-1200 PROGRAM

The NOVA-1200 operating system software resides on the system drive of the floppy disk system. To load and begin program execution, the following steps must be followed.

1. Turn the operators console (TI Silent 700) on and make sure the on-line buttom is depressed.

2. Place a program disk in the system drive.

3. Lift the PRG LOAD switch on the NOVA-1200 computer console.

The program will then be loaded from the disk into computer memory and started automatically. The user is then required to answer specific questions that the computer asks before proceeding to normal operation. The following example illustrates the startup input dialogue.

EXAMPLE - INITIALIZATION DIALOGUE

```
* NAGS V80
* MONTH: 3 ↵
* DAYS: 12 ↵
* HOURS: 9 ↵
* MINUTES: 30 ↵
* PULSER LOW ADC1: 260.312  360.251 ↵
* PULSER HIGH ADC1:  2603.125  3602.51 ↵
* PULSER LOW ADC2: 150.567  155.631 ↵
* PULSER HIGH ADC2: 1550.673  1556.31 ↵
* SYSTEM #1 RUN NO: 1 ↵
* SYSTEM #2 RUN NO: 50 ↵
*
```

The initialization dialogue first asks the user to input the current time and date. After these values have been entered the pulser equivalent energy values are printed. The computer then waits for the user to input

new values. If no change in the equivalents is required, just a carriage return is typed and the computer will proceed to the next entry. The last two entries in the initialization dialogue require the user to input the starting run numbers for the two ADC's (see section I.1.4).

The program starting address is 400_8 with a restart address 0402_8. If the program is started at the restart address, the initialization dialogue will not be run.

NRC FORM 335 (7-77) U.S. NUCLEAR REGULATORY COMMISSION

BIBLIOGRAPHIC DATA SHEET

1. REPORT NUMBER (Assigned by DDC): NUREG/CR-1668

4. TITLE AND SUBTITLE (Add Volume No., if appropriate): Advanced Mobile Multi-Processor Gamma-Ray Acquisition and Analysis System

2. (Leave blank)

3. RECIPIENT'S ACCESSION NO.

7. AUTHOR(S): E.W. Killian, R.A. Morneau

5. DATE REPORT COMPLETED: MONTH July YEAR 1980

9. PERFORMING ORGANIZATION NAME AND MAILING ADDRESS (Include Zip Code):
EG&G Idaho, Inc.
Idaho National Engineering Laboratory
Idaho Falls, ID 83415

DATE REPORT ISSUED: MONTH September YEAR 1980

6. (Leave blank)

8. (Leave blank)

12. SPONSORING ORGANIZATION NAME AND MAILING ADDRESS (Include Zip Code):
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, D.C. 20555

10. PROJECT/TASK/WORK UNIT NO.

11. CONTRACT NO.: FIN No. A6075

13. TYPE OF REPORT: Topical

PERIOD COVERED (Inclusive dates): N/A

15. SUPPLEMENTARY NOTES

14. (Leave blank)

16. ABSTRACT (200 words or less)

The report describes a new Gamma-Ray Acquisition and Analysis system which has been developed for the In-Plant Source Term Measurement Program. A new coputer was added to the system described previously in Reference 5, "Procedures, Source Term Measurement Program," NUREG-0384. One computer is now used to acquire the data and the other (new computer) is used to analyze the resulting data. The throughput of the system has been dramatically improved. Data analysis times have been reduced by about a factor of 10. Moreover, the analysis procedure is much more complex and provides results which can be directly reported with minimal operator interpretation. The information contained in this report supersedes the description of the analysis package given in Appendix B of the procedures manual (NUREG-0384).

17. KEY WORDS AND DOCUMENT ANALYSIS

17a. DESCRIPTORS

Gamma Ray
Computer
Radioisotope Analysis

17b. IDENTIFIERS/OPEN-ENDED TERMS

18. AVAILABILITY STATEMENT: Unlimited

19. SECURITY CLASS (This report): Unclassified

21. NO. OF PAGES

20. SECURITY CLASS (This page): Unclassified

22. PRICE: $

NRC FORM 335 (7-77)