U. S. DEPARTMENT OF THE INTERIOR

GEOLOGICAL SURVEY

RDARL4, a FORTRAN interface for transferring chemical analytical data from an Applied Research Laboratories electron microprobe to a PDP-11 computing system

by

J. Stephen Huebner

Open-File Report 83-713

CONTENTS

TABLE

INTRODUCTION

Microprobe analysis at the U.S. Geological Survey does not end when a complete chemical analysis is obtained by the operator. Rather, an increasing number of USGS operators want to reconstitute their chemical analyses as atomic mineral formulas, norms of endmember mineral compositions, or compositional projections. In the future, operators will want to select analyses to construct data files for plotting and for statistical analyses. The traditional method of manually transcribing analyses from microprobe printout to another computing system is laborious and prone to transcription errors. RDARL4 is a machine interface between data files created by the microprobe and the BASIC and FORTRAN programs available in Reston for recalculating and plotting microprobe data.

The electron microprobe used in the USGS's Reston Microprobe Facility is a model SEMQ (Scanning Electron Microprobe Quantometer) manufactured by Applied Research Corporation (ARL) of Sunland, California. This instrument collects chemical data for up to 9 elements simultaneously or up to 14 elements asynchronously. Chemical analyses are reduced on line and may be stored on disk if the operator so chooses. Operators may obtain 100 to 170 analyses per 4 hour operating shift. Despite the high rate of productivity, demand for access to the microprobe is great. The efficient use of both the microprobe instrumentation and the large volumes of data obtained has been a major concern of the Reston Microprobe Facility staff. Use of the program RDARL4 permits operators to transfer large volumes of data to other computers quickly, accurately, and without competing for valuable instrument time.

HARDWARE AND SOFTWARE CONSIDERATIONS

The ARL-SEMQ microprobe automated control system is built around the PDP-11 series of computers and peripherals manufactured by the Digital Equipment Corporation (DEC), specifically the LSI-11/23 processor and RX-02 floppy disk controller and drive. The PDP-11 series of equipment is commonly used for control of major laboratory instrumentation and is available "off-line" elsewhere in the microprobe facility. DEC makes available software based upon its RT-11 operating system. This software permits data to be stored in words that are 1, 2, 4, or 8 bytes in length. The microprobe operating programs are written in Applied Research Laboratories Extended Basic (ARLEB) language and use assembly language subroutines and functions that can be called from ARLEB. Although the ARL system utilizes a conventional DEC processor and diskettes, ARL's software for disk storage uses a different disk directory and writes only 6-byte data words (called "entries"), a word length that is not supported by DEC. Stored on a RX-02 disk, an ARLEB data record is 42 6-byte entries long. Each entry contains one numeric value (or up to six ASCII characters).

A compact ARLEB disk storage routine was written in ARLEB language by James J. McGee of the USGS and inserted into the microprobe control program. This routine stores all the analytical information that most USGS operators want to retain. The first record of the data storage file contains instrumental parameters that are common to all microprobe analyses that follow in the file. Each subsequent record contains information for one analysis: title, stage position coordinates, and chemical analysis in terms of weight percent and cation formula unit (Table 1). Disk storage files are created during a microprobe operating session; after the operating shift the operator removes the data disk and either stores it for later use or goes to an off-line computer to retrieve the data.

Off line computing equipment utilizes the DEC LSI-11/23 processor, RX-02 disk controller and disk drives, a video terminal, and a line printer. The off-line equipment is completely compatible with the computing equipment that controls the microprobe. This interchangeability of hardware means that the only interfacing problems involve using the DEC RT-11 family of software to access a disk written under ARLEB. Fortunately, ARL provides "RT-LIB", a library of compiled MACRO (DEC assembly language) program modules that can be called from a FORTRAN program running under the RT-11 operating system whenever it is necessary to access the ARLEB disk. Once microprobe data has been read into an RT-11 based system, it is possible to create new disk files of any of a variety of file types: sequential (ASCII character) with either FORTRAN or BASIC carriage control and direct access (binary). This enables a USGS scientist to use any of a number of different programs (Appendix I) to recalculate the analytical data.

Program RDARL4 will read a data file that contains 100 or fewer analyses of 1 to 14 elements each. It will run under the RT-11 operating system in a PDP-11 minicomputer that has 64 kbytes of lower program memory and 18.2 kbytes of virtual memory for numeric and character arrays. The version described here is a general "brute-force" program designed to run quickly and to offer many options which can be executed without rereading the ARLEB data disc. The program could be easily modified to run in a smaller memory by decreasing the number of elements allowed per analysis, limiting the number of analyses that can be processed at one time, decreasing the number of output options, and/or by making uniform the data input requirements of the programs listed in Appendix I. Although the FORTRAN portion of the code could be easily modified to run under another operating system, numerous calls are made to modules of the "RT-LIB" library which, because it is written in PDP-11 assembly language, is not readily transportable to systems that do not use the PDP-11 family of processors. Conversion to a non-DEC system would be difficult and timeconsuming.

DESCRIPTION OF PROGRAM RDARL4

The program (Appendix II) has four major parts: definition and initialization of arrays and variables; reading of the ARLEB data disk; creation of output files; and subroutines to report disk errors and to handle commonly used interactions with the operator.

Program RDARL4 uses buffer arrays (DATA, IBUF) to hold information from a single record of the ARLEB data disc file from the time the record is read until its contents can be transferred to larger arrays that store all the titles, weight percent values, cation numbers, and stage coordinates, etc., that appear in a disk file (up to 101 records). These storage arrays have nmemonic names (for instance, ITITLE, WTPCT, PEROXY, CATION) and will be discussed in conjunction with program execution, below. Also important are the byte arrays (such as BPXVD and BOXIDE) that are used to store element names in the sequence that they will be written out for a given output option. These byte arrays are filled with the necessary ASCII characters using DATA statements.

Initial program execution involves keyboard input of the name of the ARLEB data file. The program then calls FIND, an RT-LIB module that accesses the ARLEB disk directory and returns the position, length (ISIZE), and ARLEB file-type (ITYPE, BFKIND), and an error code (IERR). The next call to RT-LIB, OPEN, opens the designated file for reading and returns only an error code. Each record is then read twice because there is no way to read a partial record, yet each record contains both ASCII and numeric information which must be handled differently. First, FDGET is used to retrieve ASCII information by returning the contents of the record, byte-by-byte, in an integer buffer array IBUF. Next, FFETCH reads the ARLEB record as a block of 42 6-byte numeric entries, converts each entry to a 4-byte floating point format, and returns the converted contents in the floating point buffer array DATA.

The first record of the ARLEB data file contains operating parameters that are common to all subsequent analyses, including the analyst's name, date of analysis, names of elements analyzed (in the sequence in which concentration data are to be stored), a 4-character abbreviation for the standard used for each element, and the X-ray line used to analyze each element. The complete list of contents of this first record is given in Table 1. FORTRAN code to select ASCII and numeric information for the first record from the buffer arrays IBUF and DATA and to place that information in more permanent variables and storage arrays in memory is used only once during each program execution.

Each subsequent record contains information that pertains to a single analysis. The program contains a loop to read and store these records, one-by-one. Exit from the loop is determined either by reaching the last block of the ARLEB file (ISIZE) or by encountering an analysis with a weight percent sum equal to zero. (The test for this condition is included because an ARLEB disk file is not necessarily completely filled with analyses.)

The information retrieved and stored is also listed in Table 1 and for each analysis includes a title, weight percent values, cation and anion numbers, and a flag (IAVLFG) which indicates analysis type and number of points averaged.

Execution of RDARL4 next gives the operator the choice of 11 output formats that may be used in almost any combination. The only restrictions are that reviewing the analyses on the video terminal, if that option is chosen, must be first and that sending the analyses to a lineprinter must precede creation of output files on disk. Simple reordering of blocks of code (and some renumbering of statements) would remove this constraint on operations.

The video terminal option is useful for operators who want to review the analyses in a data file, perhaps because they do not remember exactly the contents of a particular disk file. The format of the display is compact and includes all analyses in sequence. The lines displayed exceed 80 characters in length; operators who want to use this option may first want to set the video terminal for a 130 character line length. The only information displayed from the first record is the element names. The remaining information stored on the first record can be examined with the lineprinter (see below).

The line printer option gives a hard copy of the contents of the first record (operating parameters common to all analyses), then lets the operator specify the particular analyses to be printed (subroutine SELECT). Analyses need not be specified if only a record of the microprobe operating parameters is desired. An example of the printout is included in Appendix IIIa.

The following 9 options create disk files that can be used as input to supporting programs for mineral formula calculations, plotting, and other uses. (A tenth option, EXIT, terminates program execution.) Existing programs for which 5 of the 9 options were specifically prepared are briefly summarized in Appendix I. Data input to most of these programs is in the form of sequential files so that they may be viewed (on a video terminal or printer) and edited. The choice of sequential files also permits an operator to create an input file with an editor, without using RDARL4. To facilitate editing operations, these sequential files include the symbols of elements analyzed, whether or not an auxillary program actually uses the element symbols (as in Program MINCLC) or merely reads and ignores them (as in the Programs which use disk files created by Options 2, 3, 4, and 7).

All auxillary mineral formula programs except MINCLC require concentration data that is predetermined with respect to the nature of required elements and the sequence in which they appear. The microprobe operating system has no constraints on the number, nature, or sequence of elements analyzed and stored in an ARLEB data file. The reading and storage of data by RDARL4, as previously described, preserves the original number, nature, and sequence of elements in the byte array BELNAM. The new

sequence in which element data (one datum is two ASCII character element symbols followed by a floating point numeric concentration value) are to be put out is determined by comparing BELNAM, byte-by-byte, with an array such as BOXIDE, BPCVD, BCAMP, or BBASIC that contains the desired element symbols in the correct sequence:

```
      DO 965, I=1,13
      IFLAG=0
      DO 965, J=1,NELMTS
      IF(BELNAM(1,J).NE.BOXIDE(1,I)) GOTO 963
      IF(BELNAM(2,J).NE.BOXIDE(2,I)) GOTO 963
      KCOUNT(I)=J
      IFLAG=1
963   IF(IFLAG.EQ.1) GOTO 965
      KCOUNT(I)=16
965   CONTINUE
```

Whenever two pairs of bytes match (the element symbols are the same), the position of the desired element is stored in array KCOUNT. If no match is found (microprobe analysis did not include an element expected by the auxillary program), the array element is set to 16. Thus KCOUNT contains the sequence in which element concentrations are to be written out from WTPCT and CATIONS. A value of 16 will cause a WTPCT value of 0.00 and a CATION concentration of 0.000 to be written to the disk file. Array KCOUNT is used in implied DO LOOPS to control the actual sequence in which data are written from BELNAM and WTPCT or CATIONS to disk:

```
968   WRITE(2,971) N,IAVFLG(N),(ITITLE(L,N),L=1,24)
      DO 970, I=1,13
      WRITE(2,972)(BOXIDE(J,I),J=1,2),WTPCT(KCOUNT(I),N)
970   CONTINUE
```

The routine for constructing an input file for the general mineral formula program MINCLC is different. MINCLC can be configured to operate upon different sets of elements; at present the set consists of elements with atomic numbers 1 to 30 (hydrogen through zinc), although it could be changed or enlarged. During execution, MINCLC identifies the elements to which each concentration datum pertains, regardless of the number of the elements or their sequence in the input file. Program RDARL4 does not, therefore, contain code to write out elements in a particular sequence. Rather, the code is designed to write out only a number of data fields corresponding to the number of elements that were actually analyzed; records that contain fewer than 8 elements contain "$$" to indicate the last element-concentration pair of an analysis. RDARL4 also writes an initial record that tells MINCLC what to do ("****RDOXIONS" plus the number of anions and cations in the formula unit) and a title (from ITITLE) that preceeds each analysis. Here another advantage of creating an output file that can be edited becomes apparent: if the operator includes more than one kind of mineral in a file created for MINCLC, he may want to change the formula basis by inserting additional control records (indicated by the initial "****") in that file.

Four disk file options were not designed to create an input file for a mineral recalculation program. Option 5 (BASIC virtual array) creates a direct access file for programs that have not yet been written. Both weight percent values and cation numbers are put out to the disk file. Option 6 creates a sequential file that can be used as input to the MINC BASIC statistical programs provided by DEC. Either weight percent values or cation numbers are put out. Option 8 (filing of "known-unknown" analyses) was written so that all analyses of well-characterized material (used to gage the quality of microprobe operation) can be filed for later use. Two uses are anticipated: detect and examine changes in microprobe performance with time; and evaluate the suitability of particular standards for particular elements and mineral groups. Option 8 stores operating parameters and weight percent values, but not cation numbers. Option 9 creates a sequential file that can be edited, searched, and concatenated using a keypad editor or BASIC language routines written by Elaine McGee of the USGS. In this manner, an operator can sort analyses into groups for plotting or archiving.

Four segments of code have been placed in subroutines because they are either unlikely or very likely to be called into use. Errors in reading the ARLEB disk or writing the DEC disk are rare, but if an error is encountered, the main program calls subroutine ERRORS, prints a brief description of the particular disk error encountered, and either halts execution (ARLEB disk error) or returns to the main program (DEC disk error). The other subroutines are used frequently.

Subroutine SETUP reminds the operator of the name of the ARLEB file by displaying it on the videoscreen, then asks the operator to provide a name for an output file that is to be written on disk. Because analyses from a microprobe operating shift can be used many different ways, operators may want to retain the ARLEB file name and append a file type that indicates the intended use of the output file. Thus, the ARLEB input filename "D1027" could be used in output files for several mineral recalculation programs, for example, D1027.MNC, D1027.PXV, and D1027.FEL.

Subroutine SELECT asks the operator to select the analyses that will be sent to the line printer or written to a disk file. The operator has several options: averaged analyses of unknowns; all the known-unknown analyses; or analyses, individually or in groups, designated by the sequence number (NANAL) in which they appeared in the ARLEB file and are, during execution, stored in arrays in memory. Each analysis is associated with a variable (IAVFLG) that indicates how many points were averaged to produce the analysis. Furthermore, if that analysis is of a "known-unknown", that is, an analysis of a well characterized material such as a microprobe standard that is used to gage overall system performance, the value 80 has been added to the number of points averaged to indicate that the analysis is of a "known-unknown". Thus averages of unknowns are those analyses with IAVFLG greater than 1 but less than 80 and "known-unknowns" are analyses with IAVFLG greater than or equal to 80.

Subroutine CHOICE gives the operator the possibility of adding analyses to the current output file; that file is not closed until the operator responds "N" to the question posed by CHOICE.

OPERATOR INTERACTION DURING EXECUTION

Operators should have some understanding of the flow or sequence of routines before attempting to run RDARL4. In Reston, the disk which contains the program contains a file of instructions (Appendix IV) that are printed at the video-terminal whenever the computer is started. The actual program begins by instructing the operator to place an ARLEB disk in drive #1, the right-hand drive on an RX-02 disk unit. After reading the file, the operator is first asked if the microprobe analyses are to be viewed on the video terminal; next, whether or not a printout is desired; and finally, whether or not a disk output file is to be written. If a disk file is to be created, the operator is given the choice of 9 file options or an exit from the program. Each of the disk options ends with an opportunity to select another option so that more than one kind of disk file can be produced without rereading the ARLEB disk. If another option is not selected, the operator has an opportunity to loop back to the beginning of the program to start again with a new ARLEB data file.

During execution, RDARL4 provides the operator with numerous prompts for information. These prompts are designed to be self explanatory, particularly if the operator has run the program before. However, operators will want to prepare in advance to respond to questions that involve more than a Y (yes), N (no), or C (continue) response. These questions concern:

Name of ARLEB data file (5 characters maximum)

Use to be made of the disk output file, if any (option number)

Output filename in format DYn:xxxxxx.YYY
where n=0 or 1 (left or right disc drive, respectively),
and YYY represents a file type (default DAT). Suboutine SETUP.

Cation to anion ratio (Option 1 - MINCLC output file) in the floating point format (F12.7, F12.7). For example, "4.0, 6.0" is an acceptable response

Weight % or cations (Option 6)

Number of analyses to be put out (Option 6)

Choice of individual analyses, averages only, or both
(Option 8 - Extraction of "known-unknown" analyses)

Choice of selecting averaged unknown analyses, all "known-unknown" analyses, or designated analyses for most options. (Subroutine Select)

If the operator intends to use RDARL4 to create an output file on a disk in drive #1, that disk must have already been formatted when it is exchanged for the ARLEB disk during execution.

COMPILING THE PROGRAM

RDARL4 was compiled without overlays under DEC's RT-11 single job monitor that had been patched to load at address 28000 (to enable support for virtual memory arrays). The program is long; as listed in Appendix II the "no line-numbers" option of the compiler must be used. A command file for compiling and linking is given below. RTLIB and RX02 are contained in an ARL software kit, V.1.2-99.

```
DATE
TIME
FORTRAN/EXTEND/LIST:TT:/SHOW:0/NOLINENUMBERS/OBJECT:DY1: RDARL4
RLINK
RDARL4=DY1:RDARL4,DY0:RTLIB,RX02
 C
TIME
```

References

Smyth, Joseph R. (1980) Cation vacancies and the crystal chemistry of breakdown reactions in kimberlitic omphacites. American Mineralogist, vol. 65, p. 1185-1191.

Stormer, J.C., Jr. (1983) The effects of recalculation estimates of temperature and oxygen fugacity from analyses of multicomponent oxides. Amer. Mineral., vol 68, 586-594.

Appendix I: Brief descriptions of auxilliary mineral formula programs

In this appendix, references to variable and array names will be capitalized as they are in Program RDARL4, Appendix II. Examples of printable input files are given in Appendix III; the input files are the disk files generated by RDARL4.

Option 1. MINCLC is a FORTRAN program for converting a weight percent analysis to an oxygen-based formula unit. The program will attempt to achieve a specified cation-to-anion ratio by adjusting the proportions of the various valence states of the multivalent elements Ti, Cr, Mn, and Fe, if present. If the specified cation:anion ratio is 4:6, the program will calculate an idealized pyroxene site occupancy and pyroxene formula norm. It will also accept anions which substitute for oxygen. MINCLC was written in 1977 for punched card input by Phelps Freeborn, then of the USGS. When it was converted to run on an RT-11 based system by Huebner, January 25, 1982, the options were simplified and data input was made to be from a disk file, but the data format was left unchanged:

```
****RDOXIONS      OXYgens     CATions
NANALysis, TITLE: ITITLE
up to eight pairs of element symbols (BELNAM) and WTPCT
ditto; input terminated by $$
NANAL (etc.)
```

The first byte in the input file is a non-printing FORTRAN carriage control character, ASCII line feed.

Option 2. PXVOID is a FORTRAN program that recalculates pyroxene analyses to a pyroxene formula unit based upon six oxygens. Unlike MINCLC, PXVOID considers the possibility that deviations from the ideal 4:6 cation:oxygen ratio of pyroxenes are real and caused by nonstoichiometry. As originally written by Smyth (1980), data was input from the keyboard during execution. When converted to run under RT-11 by Huebner, March 7, 1982, data input was changed to be from disk file. The file format is similar to MINCLC with the following exceptions. The control record beginning with **** is replaced by an integer, 1 to indicate oxide weight percent input or 0 to indicate element weight percent (the corresponding RDARL4 variable is NTYPE). Element symbols are written in the required sequence from array BPXVD. The first byte in the input file is the non-printing FORTRAN carriage control character, ASCII line feed.

Option 3. RECAMP is a FORTRAN program that recalculates an amphibole analysis to several different formula units, using different assumptions. This program was originally coded by F. Spear and K. Kimball of the Massachusetts Institute of Technology, November 10, 1981, and requested that data be input from the keyboard. For USGS use, the program was modified by J.S. Huebner, March 3, 1982, to accept data from a disk file. The disk input file is particularly simple:

```
NANALysis     ITITLE
9 pairs of element symbol (BCAMP) and oxide WTPCT.
```

Because the order of the elements is fixed, it is not necessary to include element symbols in the list. They were retained to facilitate examination and editing of the input file. The first byte of the input file is the non-printing FORTRAN carriage control character, ASCII line feed.

Option 4. OXCALC (following Stormer, 1983) is a BASIC program that converts microprobe chemical analyses of spinel and ilmenite to formula units and mole fractions of ulvospinel and magnetite in preparation for estimates of the temperature and pressure of last equilibration. OXCALC was modified by J.C. Stormer, February 18, 1983, for input of data from disk. Program RDARL4 contains a format statement that suppresses the initial non-printing FORTRAN carriage control character so that the file can be read under the BASIC operating system:

```
     OPEN(UNIT=2,NAME=FILSPC,CARRIAGECONTROL='FORTRAN')
     WRITE(2,966) NTYPE
966  FORMAT('+',I1)
```

The analyses which follow consist of 14 records each:

```
NANALysis, ITITLE
13 pairs of element symbol (BOXIDE) and WTPCT
```

Option 7. AMPHI, MICA, CORD, FELDS, CHLOR, and GARN, are BASIC language programs written by Marta Kempa Flohr of the USGS to calculate mineral formulas, including formal site occupancies, for amphiboles, micas, cordierites, feldspars, chlorites, or garnets, respectively. These programs require oxide or element weight percent concentration data for 18 elements in the order in which the element symbols appear in byte array BBASIC. Program RDARL4 writes a sequential file without the initial byte of FORTRAN carriage control. The contents of the file are as follows:

```
NTYPE     (0 = element, 1 = oxide weight precent)
BBASIC (18 records, each containing an element symbol)
NANAL, ITITLE for first analysis in file
```

```
IQFILE (name of standard file), PEROXY (number of oxygens in cation's
       formula unit)
WTPCT value for silicon (first element), CATION value for silicon
  .                                    ,
  .                                    ,
  .                                    ,
  .                                    ,
  .                                    ,

WTPCT value for chlorine (last element), CATION value for chlorine
NANAL, ITITLE   for second analysis in file
```

Option 9. This option creates sequential files on disk that can be concatenated and edited. Analytical data is read out in the order in which it was stored on the ARLEB disk. The initial byte of FORTRAN carriage control is supressed so that the file created with this option can be searched and concatenated with a BASIC language program or an editor.

```
NELMTS, ***, PEROXY
BELNAM elements analyzed, listed one element-per-record
NANAL, ITITLE, PEROXY, IAVFLG
WTPCT, CATION first element is silicon
 .   ,
 .   ,
 .   ,
 .   ,
 .   ,

WTPCT,  CATION last element is chlorine
WTPCT (15), CATION (15) weight percent and cation sums
NANAL, ITITLE
```

The file does not contain an initial FORTRAN carriage control character.

Appendix II. Listing of Program RDARL4

```
C   RDARL4.FOR PERMITS AN RT-11 BASED SYSTEM TO READ AN ARL DATA DISK.
C   RDARL.OBJ LINKS TO ARL'S RTLIB, A COLLECTION OF COMPILED MACRO UTILITIES
C   THAT PERFORM VARIOUS DISK OPERATIONS AND CONVERT NUMERIC DATA FROM THE ARL
C   3-WORD TO THE DEC 2-WORD FP FORMAT WHILE PRESERVING ASCII INFORMATION.
C
C   WRITTEN BY J S HUEBNER AS RDARL1.FOR  MARCH, 1982 (NUMERIC ONLY) AND
C   AS RDARL2.FOR ON  JULY 13, 1982 (NUMERIC & ASCII)
C   OUTPUT OPTIONS BY HUEBNER, WRITTEN AS NEEDED. LAST EDIT 06/15/83
C
C   ARLEB USES A BLOCK OF 42 6-BYTE ENTRIES. FORMAT OF DATA FILED BY THE ARLEB
C   STORAGE ROUTINE WRITTEN BY MCGEE FOR THE USGS RESTON PROBE FACITLTY:
C
C      FIRST RECORD CONTAINS OPERATING PARAMETERS:
C         ENTRY NUMBER      CONTENTS                          RDARL4 VARIABLE NAME
C            1 - 4         OPERATORS' NAME                        IOPNAM
C            5 - 7         ARL SYSTEM DATE                        IARDT
C              8           NAME OF *QFILE                         IQFILE
C              9           NAME OF DATA OUTPUT FILE               IARLOT
C             10           DATA REDUCTION FLAG                    NREDCT
C             11           NUMBER OF ELEMENTS ANALYZED            NELMTS
C             12           OXIDE  OR ELEMENT WT % FLAG            NTYPE
C             13           TEMPORARY COUNTER USED BY ARL          NCOUNT
C             14           FUTURE USE                             NOUSED
C           15 - 28        ELEMENT NAME,X-RAY LINE                BELNAM,BXRLIN
C           29 - 42        STANDARD NAME                          BSTAND
C      SUBSEQUENT RECORDS CONTAIN ONE ANALYSIS EACH
C            1 - 8         TITLE                                  ITITLE
C              9           STAGE POSITION X                       X
C             10           STAGE POSITION Y                       Y
C             11           AVERAGING & KNOWN-UNKNOWN FLAG         IAVFLG
C           12 - 25        WEIGHT PERCENT                         WTPCT
C             26           TOTAL WEIGHT PERCENT                   WTPCT
C           27 - 40        CATION NUMBERS PER FORMULA UNIT        CATION
C             41           SUM OF CATIONS                         CATION
C             42           NUMBER ANIONS PER FORMULA UNIT         PEROXY
C
      DIMENSION DATA(42)                !buffers numeric records from FETCH
      DIMENSION IBUF(128)               !buffers ASCII records from FDGET
      INTEGER IOPNAM(12),IQFILE(3),IARLOT(3),KCOUNT(14),UPPER,NAME(3)
      INTEGER*4 ITIME
      BYTE BFILNM(15),BFKIND(18),BSTAND(4,14),BBUF(256)
      BYTE BELNAM(2,14),BXRLIN(2,14),FILSPC(15),BFLAG1,BFLAG2,BFLAG3
      BYTE BRPLYO,BPXVD(2,13),BCAMP(2,9),BBASIC(2,18),BOXIDE(2,13)
      VIRTUAL ITITLE(24,100),WTPCT(16,100),CATION(16,100)
      VIRTUAL NANAL(100),IAVFLG(100),PEROXY(100),X(100),Y(100)
      EQUIVALENCE (NAME(1),BFILNM(5)),(BBUF(1),IBUF(1))
      COMMON BFLAG1,BFLAG2,BFLAG3
      DATA BFILNM/'D','Y','1',':','D','E','F','A','L','T',6*0/
      DATA BFKIND/'B','A','S','I','C',' ','S','Y','S','T','E','M',
     &'D','A','T','A',' ',' '/
      DATA BPXVD/'S','I',' ',' ','A','L','T','I','C','R','F','C',
     &'F','E','M','G','M','N','C','A','L','I','N','A','K',' '/
      DATA BCAMP/'N','A','M','G','A','L','S','I','K',' ','C','A',
     &'T','I','M','N','F','E'/
      DATA BOXIDE/'S','I','T','I','A','L','V',' ','C','R','F','C',
     &'F','E','M','N','M','G','C','A','Z','N','N','I','N','B'/
```

Appendix II. Listing of Program RDARL4. (continued)

```
      DATA BBASIC/'S','I','T','I','A','L','F','C','C','R','F','E',
     &'Z','N','M','N','N','I','M','G','C','A','S','R','B','A',
     &'N','A','K',' ','P',' ','F',' ','C','L'/
C
C  INITIALIZE VARIABLES AND GET SYSTEM DATE AND TIME
C
      TYPE*,' '
      TYPE*,'PROGRAM RDARL4 PERMITS RT-11 TO READ ARLEB DATA FILES.'
      TYPE*,'IT WAS WRITTEN BY J S HUEBNER FOR THE RESTON PROBE FACILITY.'
      TYPE*,'THIS VERSION WAS COMPILED 06/15/83.'
      CALL IDATE(IDATE1,IDATE2,IDATE3)
      IF(IDATE2.EQ.0) STOP 'SET SYSTEM DATE AND TIME, THEN BEGIN AGAIN'
      CALL GTIM(ITIME)
      CALL CVTTIM(ITIME,IHOURS,IMIN,ISEC,ITICK)
199   BFILNM(15)=0
      IDRIVE=1
200   IERR=0
      IREC=0
      TYPE*,' '
      TYPE*,'PLACE DISK WITH ARLEB DATA FILE IN DRIVE DY1:'
      TYPE*,' '
      TYPE*,'ENTER THE ARLEB FILENAME'
      TYPE*,'(ARLEB accepts no more than 5 characters for the filemanme.)'
      ACCEPT 210,NAME
210   FORMAT(3A2)
C
C  FIND AND OPEN THE ARLEB OUTPUT DATA FILE
C
      TYPE*,' '
      TYPE*,'The disk access and data transfer will begin.'
      ICHAN=IGETC()
      TYPE*,'Channel',ICHAN,' allocated by the computer.'
      CALL FIND (NAME,IDRIVE,IADD,ISIZE,ITYPE,IERR)
      IF(IERR.NE.-1) CALL ERRORS(IERR,IREC)
      IF(ITYPE-2) 310,320,330
310   N=1
      GOTO 340
320   N=7
      GOTO 340
330   N=13
340   TYPE 350,NAME,(BFKIND(I),I=N,N+5),IADD,ISIZE
350   FORMAT(1H0,3A2,' is a ',6A1,' file whose first record is ',/,1H ,
     &'block ',I4,' and whose length is ',I3,' blocks of 128 words.')
      CALL OPEN (NAME,IDRIVE,IERR)
      IF(IERR.NE.-1) CALL ERRORS(IERR,IREC)
C
C  READ THE ARL OUTPUT FILE, BLOCK BY BLOCK, REFORMAT EACH RECORD (BLOCK)
C  FOR STORAGE IN ARRAYS, AND DISPLAY HEADER INFORMATION ON VIDEO SCREEN
C
      IREC=1                                    !read $QSET and $PSET information
      CALL FDGET(IBUF,IADD,IDRIVE,IERR)          !first without format conversion
      IF(IERR.NE.-1) CALL ERRORS(IERR,IREC)
      DO 125 I=1,256
      IF(BBUF(I).EQ.0) BBUF(I)='040
125   CONTINUE
      CALL FFETCH (IREC,IDRIVE,DATA,IERR)    !with 3 to 2 word format convers.
```

```
      IF(IERR.NE.-1) CALL ERRORS(IERR,IREC)  !assign it to variables, arrays:
      DO 130 I=1,12
      IOPNAM(I)=IBUF(I)
130   CONTINUE
      IARDT2=DATA(5)
      IARDT1=DATA(6)
      IARDT3=DATA(7)
      DO 134 I=1,3
      IQFILE(I)=IBUF(I+21)
      IARLOT(I)=IBUF(I+24)
134   CONTINUE
      NREDCT=DATA(10)
      NELMTS=DATA(11)
      NTYPE=DATA(12)
      NCOUNT=DATA(13)
      NOUSED=DATA(14)
      DO 150 J=1,14
      DO 145 I=1,2
      BELNAM(I,J)=BBUF(6*J+78+I)
      BXRLIN(I,J)=BBUF(6*J+81+I)
145   CONTINUE
      DO 150 I=1,4
      BSTAND(I,J)=BBUF(6*J+163+I)
150   CONTINUE
      TYPE 152,(IOPNAM(M),M=1,12)
152   FORMAT(1H0,'THE MICROPROBE OPERATOR WAS ',12A2)
      TYPE 153,IARDT2,IARDT1,IARDT3
153   FORMAT(1H ,'DATE ANALYSES MADE: ',I2,'-',I2,'-19',I2)
      TYPE*,' '
155   FORMAT(9H ELEMENT ,I3,' is ',2A1,1X,2A1,' with standard ',4A1)
      DO 160 K=1,NELMTS
      TYPE 155,K,(BELNAM(I,K),I=1,2),(BXRLIN(I,K),I=1,2),(BSTAND(I,K),I=1,4)
160   CONTINUE
      TYPE*,' '
      TYPE*,'I AM STILL READING THE ARLEB DATA DISC - PLEASE BE PATIENT!'
      TYPE*,' '
      DO 175 IREC=2,ISIZE                    !read $ANLZ output data
      KOUNTR=IREC-1                          !count # of analysis
      CALL FDGET(IBUF,IADD+IREC-1,IDRIVE,IERR)  !no format conversion
      IF(IERR.NE.-1) CALL ERRORS(IERR,IREC)
      DO 165 N=1,256
      IF(BBUF(N).EQ.0) BBUF(N)='040
165   CONTINUE
      CALL FFETCH (IREC,IDRIVE,DATA,IERR)       !3 to 2 fmt conversion
      IF(IERR.NE.-1) CALL ERRORS(IERR,IREC)
      IF(DATA(26).EQ.0) KOUNTR=KOUNTR-1         !last+1 has wt%=0.00
      IF(DATA(26).EQ.0.) TYPE*,'LAST ANALYSIS IS #',KOUNTR
      IF(DATA(26).EQ.0.) GOTO 176
      NANAL(KOUNTR)=KOUNTR
      DO 172 L=1,24
      ITITLE(L,KOUNTR)=IBUF(L)
172   CONTINUE
      X(KOUNTR)=DATA(9)
      Y(KOUNTR)=DATA(10)
      IAVFLG(KOUNTR)=DATA(11)
      DO 174 M=1,15
```

Appendix II. Listing of Program RDARL4. (continued)

```
      WTPCT(M,KOUNTR)=DATA(M+11)
      CATION(M,KOUNTR)=DATA(M+26)
174   CONTINUE
      PEROXY(KOUNTR)=DATA(42)
175   CONTINUE
176   CONTINUE
C
C  REPORT END OF READ AND STORE-IN-ARRAY  OPERATION
C
      LBLOCK=IREC+IADD-1
      TYPE 380
380   FORMAT(1H0,'         I JUST FINISHED READING THE ARLEB DISC.')
      WRITE(7,381)LBLOCK
381   FORMAT(1H0,'            The last block read has number ',I4)
      TYPE*,' '
382   TYPE*,'*************** REMOVE ARLEB DISC FROM DRIVE DY1: ***************'
      TYPE*,' '
      TYPE*,'                    THEN TYPE (C) TO CONTINUE'
      ACCEPT 701,IFLAG
      IF(IFLAG.NE.'C') GOTO 382
C
C  VIEW DATA ON CRT
C
401   TYPE*,'EXAMINE ANALYSES WITH VIDEO TERMINAL (Y or N)?'
      ACCEPT 701, IRPLY0
      IF(IRPLY0.NE.'Y'.AND.IRPLY0.NE.'N') GOTO 401
      IF(IRPLY0.NE.'Y') GOTO 703
      DO 410, I=1,KOUNTR
      NUMAVG=IAVFLG(I)
      IF(IAVFLG(I).GE.80) NUMAVG=NUMAVG-80
      TYPE 732,NANAL(I),(ITITLE(J,I),J=1,24)
      IF((IAVFLG(I).NE.0).AND.(IAVFLG(I).NE.80)) TYPE 445,NUMAVG
      TYPE 733,((BELNAM(J,K),J=1,2),K=1,14)
      TYPE 734,(WTPCT(J,I),J=1,15)
      TYPE 735,(CATION(J,I),J=1,15),PEROXY(I)
445   FORMAT(1H ,'AVERAGE OF ',I2)
410   CONTINUE
C
C  PRINT OUT DATA STORED IN ARRAYS
C
701   FORMAT(A1)
702   FORMAT(I3,I3)
      TYPE*,' '
703   TYPE*,'PRINT OUT HARD COPY OF DATA? (respond Y or N)'
      ACCEPT 701,IRPLY1
      IF(IRPLY1.NE.'Y'.AND.IRPLY1.NE.'N') GOTO 703
      IF(IRPLY1.NE.'Y') GOTO 800
      PRINT 710
710   FORMAT(1H ,'RDARL4 ENABLES RT-11 TO ACCESS CHEMICAL ANALYSES
     & STORED IN ARLEB FORMAT BY THE RESTON ARL/SEMQ MICROPROBE')
      PRINT 711
711   FORMAT(1H0,'IT WAS WRITTEN BY J S HUEBNER FOR THE RESTON
     & MICROPROBE FACILITY AND COMPLETED JUNE 15, 1983')
      PRINT 712,IDATE1,IDATE2,IDATE3,IHOURS,IMIN
712   FORMAT(30HORT-11 RUN DATE AND TIME WERE ,I2,'-',I2,'-19',I2,
     &' AT ',I2,':',I2)
```

Appendix II. Listing of Program RDARL4. (continued)

```
      PRINT 721,(IOPNAM(I),I=1,12),IARDT2,IARDT1,IARDT3
721   FORMAT(22HOMICROPROBE OPERATOR: ,12A2,25X,'MICROPROBE ANALYSIS
     & DATE ',I2,'-',I2,'-19',I2)
      PRINT 722,(IQFILE(I),I=1,3),(IARLOT(I),I=1,3)              !integer arrays
722   FORMAT(60HOTHE TITLES OF THE *QSET INPUT AND $ANALZ OUTPUT
     &FILES WERE ,3A2,' and ',3A2)
      PRINT 723
723   FORMAT(1HO,'   ELEMENT      X-LINE       STANDARD')
      DO 724 J=1,NELMTS
      PRINT 725,(BELNAM(I,J),I=1,2),(BXRLIN(I,J),I=1,2),(BSTAND(I,J),I=1,4)
724   CONTINUE
725   FORMAT(1H ,5X,2A1,10X,2A1,10X,4A1)
      PRINT*,' '
      IF(NTYPE.EQ.0) PRINT*,'ANALYSES STORED AS ELEMENT WEIGHT PERCENT'
      IF(NTYPE.EQ.1) PRINT*,'ANALYSES STORED AS OXIDE WEIGHT PERCENT'
      PRINT*,' '
      IF(NREDCT.EQ.0) PRINT*,'REDUCTION METHOD NOT SPECIFIED'
      IF(NREDCT.EQ.0) GOTO 750
      IF(NREDCT.GT.5) PRINT*,'REDUCTION CODE =',NREDCT
      IF(NREDCT.GT.5) GOTO 750
      GOTO (726,727,728,729,730) NREDCT
726   PRINT*,'DATA REDUCED WITH MAGIC-IV ROUTINE OF J W COLBY'
      GOTO 750
727   PRINT*,'DATA REDUCTION BY METHOD OF BENCE & ALBEE'
      GOTO 750
728   PRINT*,'DATA REDUCTION BY ARLs G-FACTOR METHOD'
      GOTO 750
729   PRINT*,'DATA NOT REDUCED BEYOND K-RATIO CALCULATION'
      GOTO 750
730   PRINT*,'DATA REDUCTION BY METHOD OF ZIEBOLD & OGILVIE'
      PRINT*,' '
      REWIND 6
750   PRINT*,' '
      CALL SELECT(LOWER,UPPER,INDFLG,JAVFLG,KNWFLG)
      IF(INDFLG.EQ.1) PRINT 755
      IF(JAVFLG.EQ.1) PRINT 756
      IF(KNWFLG.EQ.1) PRINT 757
      IF(INDFLG.NE.1) LOWER=1
      IF(INDFLG.NE.1) UPPER=KOUNTR
      IF(LOWER.EQ.0.AND.UPPER.EQ.0) GOTO 731
      DO 731 I=LOWER,UPPER
      IF(INDFLG.EQ.1) GOTO 751
      IF(JAVFLG.EQ.1.AND.IAVFLG(I).GT.1.AND.IAVFLG(I).LT.80) GOTO 751
      IF(KNWFLG.EQ.1.AND.IAVFLG(I).GE.80) GOTO 751
      GOTO 731
751   PRINT 732,NANAL(I),(ITITLE(J,I),J=1,24)
      IF(X(I).NE.0) PRINT 740,X(I),Y(I)
      PRINT 733,((BELNAM(J,K),J=1,2),K=1,14)
      PRINT 734,(WTPCT(J,I),J=1,15)
      NUMAVG=IAVFLG(I)
      IF(IAVFLG(I).GE.80) NUMAVG=NUMAVG-80
      IF((IAVFLG(I).NE.0).AND.(IAVFLG(I).NE.80)) PRINT 745,NUMAVG
      PRINT 735,(CATION(J,I),J=1,15),PEROXY(I)
731   CONTINUE
      REWIND 6
732   FORMAT(1HO,I3,' TITLE: ',24A2)
```

Appendix II. Listing of Program RDARL4. (continued)

```
740   FORMAT(1H+,60X,'X = ',F7.1,5X,'Y = ',F7.1)
733   FORMAT(5H      ,2X,14(2A1,5X),'TOTAL')
734   FORMAT(5H WT %,14(F6.3,1X),F7.3)
745   FORMAT(1H+,124X,'AV OF ',I2)
735   FORMAT(5H CAT ,14(F6.3,1X),1X,F6.3,1X,'PER ',F7.3,' ANIONS')
755   FORMAT(1H0,'INDIVIDUALLY SELECTED ANALYSES OF ALL TYPES:')
756   FORMAT(1H0,'AVERAGES OF UNKNOWN ANALYSES ONLY:')
757   FORMAT(1H0,'ALL KNOWN-UNKNOWN ANALYSES:')
      CALL CHOICE(ANSWER)
      IF(ANSWER.EQ.'Y') GOTO 750
C
C  SELECT FORMAT FOR STORAGE OF DATA IN DISC FILE
C
800   TYPE*,'DISC OUTPUT? (reply Y or N)'
      ACCEPT 701,IRPLY4
      IF(IRPLY4.EQ.'N') GOTO 1999
      TYPE*,' '
      TYPE*,'PLACE OUTPUT DISC in DRIVE 1'
801   TYPE*,' '
      BFLAG1='040
      BFLAG2='040
      BFLAG3='040
      TYPE*,'SELECT OUTPUT FILE OPTION FOR ONE OF THE FOLLOWING:'
      TYPE*,'   1 MINCLC       5 BASIC VIRTUAL ARRAY'
      TYPE*,'   2 PXVOID       6 BASIC SEQUENTIAL FILE FOR REGRESSIONS'
      TYPE*,'   3 RECAMP       7 BASIC MINERAL FORMULA RECALCULATIONS'
      TYPE*,'   4 OXIDES       8 EXTRACT ANALYSES OF KNOWN-UNKNOWNS'
      TYPE*,'                  9 BASIC SEQUENTIAL FILE FOR DATA STORAGE'
      TYPE*,'                 10 EXIT FROM PROGRAM'
      TYPE*,' '
      ACCEPT 802,IRPLY5
802   FORMAT(I3)
      IF(IRPLY5.GT.10) TYPE*,'REPLY OUT OF BOUNDS. TRY AGAIN.'
      IF(IRPLY5.GT.10) GOTO 801
      GOTO (805,840,860,960,880,900,920,940,1100,1999),IRPLY5
C
C     OPTION 1, MINCLC, COMPLETED JULY 14, 1982
C
805   TYPE*,'YOU ELECTED A MINCLC FILE. NOW YOU MUST ENTER THE IDEAL'
      TYPE*,'NUMBER OF CATIONS PER NUMBER OF OXYGENS (f12.7,f12.7):'
      READ(5,806,ERR=803) CAT,OXY
      IF(CAT.LE.0.01) GOTO 803
      IF(OXY.LE.0.01) GOTO 803
      GOTO 804
803   IERR=12
      CALL ERRORS(IERR,IREC)
      GOTO 805
806   FORMAT(2(F12.7))
804   CALL SETUP(IARLOT,FILSPC)
      OPEN(UNIT=2,NAME=FILSPC,ERR=807,CARRIAGECONTROL='LIST')
      GOTO 809
807   IERR=11
      CALL ERRORS(IERR,IREC)
      GOTO 804
809   WRITE (2,808,ERR=807) OXY,CAT
808   FORMAT(12H****RDOXIONS,38X,F10.0,F10.0)
```

Appendix II. Listing of Program RDARL4. (continued)

```
810   CALL SELECT(LOWER,UPPER,INDFLG,JAVFLG,KNWFLG)
      IF(INDFLG.NE.1) LOWER=1
      IF(INDFLG.NE.1) UPPER=KOUNTR
      DO 825 K=LOWER,UPPER
      IF(INDFLG.EQ.1) GOTO 811
      IF(JAVFLG.EQ.1.AND.IAVFLG(K).GT.1.AND.IAVFLG(K).LT.80) GOTO 811
      IF(KNWFLG.EQ.1.AND.IAVFLG(K).GE.80) GOTO 811
      GOTO 825
811   WRITE(2,826) K,(ITITLE(J,K),J=1,24)
      IF(NELMTS.GT.8) GOTO 818
      GOTO(813,814,815,816,817,818,819) NELMTS-2
813   WRITE(2,833)(((BELNAM(I,J),I=1,2),WTPCT(J,K)),J=1,3)
      GOTO 825
814   WRITE(2,834)(((BELNAM(I,J),I=1,2),WTPCT(J,K)),J=1,4)
      GOTO 825
815   WRITE(2,835)(((BELNAM(I,J),I=1,2),WTPCT(J,K)),J=1,5)
      GOTO 825
816   WRITE(2,836)(((BELNAM(I,J),I=1,2),WTPCT(J,K)),J=1,6)
      GOTO 825
817   WRITE(2,837)(((BELNAM(I,J),I=1,2),WTPCT(J,K)),J=1,7)
      GOTO 825
818   WRITE(2,838)(((BELNAM(I,J),I=1,2),WTPCT(J,K)),J=1,8)
      GOTO (825,819,820,821,822,823,824) NELMTS-7
819   WRITE(2,831)(BELNAM(I,9),I=1,2),(WTPCT(9,K))
      GOTO 825
820   WRITE(2,832)(((BELNAM(I,J),I=1,2),WTPCT(J,K)),J=9,10)
      GOTO 825
821   WRITE(2,833)(((BELNAM(I,J),I=1,2),WTPCT(J,K)),J=9,11)
      GOTO 825
822   WRITE(2,834)(((BELNAM(I,J),I=1,2),WTPCT(J,K)),J=9,12)
      GOTO 825
823   WRITE(2,835)(((BELNAM(I,J),I=1,2),WTPCT(J,K)),J=9,13)
      GOTO 825
824   WRITE(2,836)(((BELNAM(I,J),I=1,2),WTPCT(J,K)),J=9,14)
825   CONTINUE
      CALL CHOICE(ANSWER)
      IF(ANSWER.EQ.'Y') GOTO 810
826   FORMAT(I3,' TITLE: '24A2)
831   FORMAT(2A1,F6.2,'$$')
832   FORMAT(2(2A1,F6.2),'$$')
833   FORMAT(3(2A1,F6.2),'$$')
834   FORMAT(4(2A1,F6.2),'$$')
835   FORMAT(5(2A1,F6.2),'$$')
836   FORMAT(6(2A1,F6.2),'$$')
837   FORMAT(7(2A1,F6.2),'$$')
838   FORMAT(8(2A1,F6.2))
      CLOSE(UNIT=2)
      GOTO 990
C
C     OPTION 2, PXVOID, COMPLETED AUGUST 18, 1982
C
840   DO 841, N=1,KOUNTR
      WTPCT(16,N)=0.0                              !for elems not analyzed
841   CONTINUE
      TYPE*,'You elected a Fortran sequential file for PXVOID'
      TYPE*,'Your data must be in OXIDE WEIGHT PERCENT'
```

Appendix II. Listing of Program RDARL4. (continued)

```
      TYPE*,' '
      DO 846, I=1,12                                  !sort element names
      IFLAG=0
      DO 846, J=1,NELMTS
      IF(BELNAM(1,J).NE.BPXVD(1,I)) GOTO 845
      IF(BELNAM(2,J).NE.BPXVD(2,I)) GOTO 845
      KCOUNT(I)=J                                     !stores sequence of WTPCT
      IFLAG=1
845   IF(IFLAG.EQ.1) GOTO 846
      KCOUNT(I)=16
846   CONTINUE
      TYPE 847,(KCOUNT(I),I=1,13)
847   FORMAT(27H SEQUENCE ELEMENTS PUT OUT:,13(2X,I2))
      TYPE*,'A VALUE OF 16 INDICATES ELEMENT NOT ANALYZED - 0.00%'
842   CALL SETUP(IARLOT,FILSPC)
      OPEN(UNIT=2,NAME=FILSPC,ERR=843,CARRIAGECONTROL='LIST')
      GOTO 844
843   IERR=11
      CALL ERRORS(IERR,IREC)
      GOTO 842
844   WRITE(2,855,ERR=843) NTYPE
848   CALL SELECT(LOWER,UPPER,INDFLG,JAVFLG,KNWFLG)
      IF(INDFLG.NE.1) LOWER=1
      IF(INDFLG.NE.1) UPPER=KOUNTR
      DO 850 N=LOWER,UPPER
      IF(INDFLG.EQ.1) GOTO 849
      IF(JAVFLG.EQ.1.AND.IAVFLG(N).GT.1.AND.IAVFLG(N).LT.80) GOTO 849
      IF(KNWFLG.EQ.1.AND.IAVFLG(N).GE.80) GOTO 849
      GOTO 850
849   WRITE(2,856) N,(ITITLE(L,N),L=1,24)
      WRITE(2,858) (BPXVD(J,1),J=1,2),WTPCT(KCOUNT(1),N),
     &((BPXVD(J,I),J=1,2),WTPCT(KCOUNT(I),N),I=3,7)
      WRITE(2,858) ((BPXVD(J,I),J=1,2),WTPCT(KCOUNT(I),N),I=8,13)
850   CONTINUE
855   FORMAT(I1)
856   FORMAT(I3,1X,24A2,20X)
858   FORMAT(6(2A1,F6.2))
      CALL CHOICE(ANSWER)
      IF(ANSWER.EQ.'Y') GOTO 848
      CLOSE(UNIT=2)
      GOTO 990
C
C     OPTION 3, RECAMP, COMPLETED AUGUST 19, 1982
C
860   DO 861, N=1,KOUNTR
      WTPCT(16,N)=0.0
861   CONTINUE
      TYPE*,'You elected a Fortran sequential file for RECAMP'
      TYPE*,' '
      DO 866, I=1,9
      IFLAG=0
      DO 866, J=1,NELMTS
      IF(BELNAM(1,J).NE.BCAMP(1,I)) GOTO 865
      IF(BELNAM(2,J).NE.BCAMP(2,I)) GOTO 865
      KCOUNT(I)=J
      IFLAG=1
```

Appendix II. Listing of Program RDARL4. (continued)

```
865   IF(IFLAG.EQ.1) GOTO 866
      KCOUNT(I)=16
866   CONTINUE
862   CALL SETUP(IARLOT,FILSPC)
      OPEN(UNIT=2,NAME=FILSPC,ERR=863,CARRIAGE CONTROL='LIST')
      GOTO 867
863   IERR=11
      CALL ERRORS(IERR,IREC)
      BFLAG1='N'
      GOTO 862
867   CALL SELECT(LOWER,UPPER,INDFLG,JAVFLG,KNWFLG)
      IF(INDFLG.NE.1) LOWER=1
      IF(INDFLG.NE.1) UPPER=KOUNTR
      DO 870, N=LOWER,UPPER
      IF(INDFLG.EQ.1) GOTO 868
      IF(JAVFLG.EQ.1.AND.IAVFLG(N).GT.1.AND.IAVFLG(N).LT.80) GOTO 868
      IF(KNWFLG.EQ.1.AND.IAVFLG(N).GE.80) GOTO 868
      GOTO 870
868   WRITE(2,871,ERR=863) N,(ITITLE(L,N),L=1,18)
      WRITE(2,872) ((BCAMP(J,I),J=1,2),WTPCT(KCOUNT(I),N),I=1,9)
870   CONTINUE
871   FORMAT(I3,X,18A2)
872   FORMAT(9(2A1,F6.2))
      CALL CHOICE(ANSWER)
      IF(ANSWER.EQ.'Y') GOTO 867
      CLOSE(UNIT=2)
      GOTO 990
C
C     OPTION 5, DIRECT ACCESS FILE, COMPLETED JULY 6, 1982
C
880   TYPE*,'YOU CHOSE A VIRTUAL (DIRECT ACCESS) DISC ARRAY FILE. ALL DATA'
      TYPE*,'ARE PUT OUT; SELECTION OF DATA IS NOT YET POSSIBLE (8/31/82).'
      TYPE 881,(IARLOT(I),I=1,3)
881   FORMAT(1H0,'THE ARLEB OUTPUT FILENAME WAS ',3A2)
      TYPE*,'ENTER BASIC VIRTUAL ARRAY FILESPEC AS DY?:??????.???  '
      TYPE*,' '
      LENGTH=4*(NELMTS+1)+3  !length of each record in 2-byte words
      CALL ASSIGN(2,' ',-1)
      DEFINE FILE 2 (KOUNTR,LENGTH,U,NREC)
      NREC=1
      DO 890 I=1,KOUNTR
      WRITE(2'NREC) KOUNTR,NANAL(I),NELMTS,(WTPCT(J,I),J=1,NELMTS),
     &WTPCT(15,I),(CATION(J,I),J=1,NELMTS),CATION(15,I)
890   CONTINUE
      CLOSE(UNIT=2)
      TYPE*,NREC,' RECORDS WERE PUT OUT TO DISC.'
      TYPE*,'EACH RECORD CONSISTS OF',LENGTH,' 2-BYTE WORDS.'
      GOTO 990
C
C     OPTION 6, BASIC REGRESSION FILE, COMPLETED AUGUST 3, 1982
C
900   TYPE*,'YOU SELECTED A MINC BASIC SEQUENTIAL FILE FORMATTED FOR THE'
      TYPE*,'MINC CORRELATION MATRIX AND REGRESSIONS. THESE PROGRAMS ARE'
      TYPE*,'LIMITED TO 75 ANALYSES. YOU WILL NEED TO HAVE SELECTED YOUR'
      TYPE*,'ANALYSES IN ADVANCE AND KNOW THE TOTAL NUMBER TO BE WRITTEN'
      TYPE*,' '
```

Appendix II. Listing of Program RDARL4. (continued)

```
      TYPE*,'IF YOU ARE NOT PREPARED TO PROCEED, TYPE ^C (control C)'
      TYPE*,' '
      TYPE*,'DO YOU WANT TO STORE WT % (1) OR CATIONS (2)?'
901   FORMAT(I3)
      ACCEPT 901,IRPLY7
912   CALL SETUP(IARLOT,FILSPC)
      OPEN(UNIT=2,NAME=FILSPC,ERR=913,CARRIAGECONTROL='FORTRAN')
      GOTO 914
913   IERR=11
905   CALL ERRORS(IERR,IREC)
      GOTO 912
914   NUMBER=0
      TYPE*,'HOW MANY ANALYSES SELECTED?'
      ACCEPT 901,IRPLY8
      WRITE(2,902,ERR=913) IRPLY8,NELMTS
902   FORMAT('+',I3,',',I2)
904   TYPE*,'NOW ENTER THE FIRST AND LAST ANALYSIS NUMBERS IN A GROUP'
      TYPE*,'OF CONSECUTIVE ANALYSES (I2,I2)'
906   FORMAT(I3,I3)
      ACCEPT 906,K,L
      IF(IRPLY7.EQ.2) GOTO 910      !branch to write out cations
      DO 907 I=K,L
903   FORMAT(1X,I3)
      WRITE(2,903,ERR=919) I
      NUMBER=NUMBER+1
      DO 907 J=1,NELMTS
      WRITE(2,909,ERR=919) WTPCT(J,I)
907   CONTINUE
      IF(NUMBER.GE.IRPLY8) GOTO 916
      GOTO 904
909   FORMAT(1X,F6.2)
      GOTO 916
910   CONTINUE
911   FORMAT(1X,F6.3)
      DO 915 I=K,L
      WRITE(2,903,ERR=919) I
      NUMBER=NUMBER+1
      DO 915 J=1,NELMTS
      WRITE(2,911,ERR=919) CATION(J,I)
915   CONTINUE
      IF(NUMBER.GE.IRPLY8) GOTO 916
      GOTO 904
916   CLOSE(UNIT=2)
      TYPE*,' '
      TYPE*,'USING THIS OPTION,'
      TYPE*,'YOU HAVE CREATED A FILE OF ',NUMBER,' ANALYSES.'
      TYPE*,'DO YOU WANT ANOTHER DISC FILE USING THIS OPTION (Y or N)?'
      ACCEPT 701,IRPLY9
      IF(IRPLY9.EQ.'Y') GOTO 900
      GOTO 990
919   IERR=13
      GOTO 905
C
C     OPTION 7, BASIC FORMULA RECALCULATIONS, COMPLETED JUNE 08, 1983
C
920   TYPE*,'YOU CHOSE TO CREATE A BASIC SEQUENTIAL FILE FOR MINERAL
```

Appendix II. Listing of Program RDARL4. (continued)

```
     & FORMULA CALCULTIONS.'
      DO 921,N=1,KOUNTR
      WTPCT(16,N)=0.0
      CATION(16,N)=0.0
921   CONTINUE
      DO 925,I=1,18
      IFLAG=0
      DO 925,J=1,NELMTS
      IF(BELNAM(1,J).NE.BBASIC(1,I)) GOTO 923
      IF(BELNAM(2,J).NE.BBASIC(2,I)) GOTO 923
      KCOUNT(I)=J
      IFLAG=1
923   IF(IFLAG.EQ.1) GOTO 925
      KCOUNT(I)=16
925   CONTINUE
935   CALL SETUP(IARLOT,FILSPC)
      OPEN(UNIT=2,NAME=FILSPC,ERR=936,CARRIAGECONTROL='FORTRAN')
      GOTO 937
936   IERR=11
      CALL ERRORS(IERR,IREC)
      GOTO 935
937   WRITE(2,926,ERR=936) NTYPE
      DO 922,L=1,18
      WRITE(2,924) (BBASIC(K,L),K=1,2)
922   CONTINUE
924   FORMAT(1X,2A1)
926   FORMAT(1H+,I1)
927   CALL SELECT(LOWER,UPPER,INDFLG,JAVFLG,KNWFLG)
      IF(INDFLG.NE.1) LOWER=1
      IF(INDFLG.NE.1) UPPER=KOUNTR
      DO 929, N=LOWER,UPPER
      IF(INDFLG.EQ.1) GOTO 928
      IF(JAVFLG.EQ.1.AND.IAVFLG(N).GT.1.AND.IAVFLG(N).LT.80) GOTO 928
      IF(KNWFLG.EQ.1.AND.IAVFLG(N).GE.80) GOTO 928
      GOTO 929
928   WRITE(2,930) N,(ITITLE(L,N),L=1,24)
      WRITE(2,931) (IQFILE(I),I=1,3),PEROXY(N)
      DO 929, I=1,18
      WRITE(2,932) WTPCT(KCOUNT(I),N),CATION(KCOUNT(I),N)
929   CONTINUE
930   FORMAT(1X,I3,',',24A2)
931   FORMAT(1X,3A2,',',F6.3)
932   FORMAT(1X,F6.2,',',F6.3)
      CALL CHOICE(ANSWER)
      IF(ANSWER.EQ.'Y') GOTO 927
      CLOSE(UNIT=2)
      GOTO 990
C
C     OPTION 8, EXTRACT AND FILE KNOWN-UNKNOWN ANALYSES, COMPLETED 10/7/82
C
940   TYPE*,' '
      TYPE*,'MINC WILL FILE ON DISC ONLY THE ANALYSES OF KNOWN-UNKNOWNS.'
      L=0
      M=0
      NUMBER=0
      DO 941, I=1,KOUNTR
```

Appendix II. Listing of Program RDARL4. (continued)

```
      IF(IAVFLG(I).EQ.80) L=L+1
      IF(IAVFLG(I).GT.80) M=M+1
941   CONTINUE
      IF(M+L.EQ.0) GOTO 959
      TYPE 942,L,M
942   FORMAT(1H0,'MINC FOUND ',I3,' INDIVIDUAL AND ',I3,' AVERAGE
     & ANALYSES OF KNOWN-UNKNOWNS')
      TYPE*,' '
      TYPE*,'DO YOU WANT TO FILE (1)INDIVIDUAL, (2)AVERAGE, (3)BOTH,'
      TYPE*,'OR (4)NEITHER KIND OF ANALYSES ? (respond 1, 2, 3, or 4)'
      ACCEPT 958, IRPLYO
958   FORMAT(I1)
      IF(IRPLYO.EQ.4) GOTO 990
943   CALL SETUP(IARLOT,FILSPC)
      OPEN(UNIT=2,NAME=FILSPC,TYPE='NEW',ERR=948,CARRIAGECONTROL='FORTRAN')
      GOTO 949
948   IERR=11
      CALL ERRORS(IERR,IREC)
      GOTO 943
949   DO 950,K=1,KOUNTR
      IF(IRPLYO.EQ.1.AND.IAVFLG(K).EQ.80) GOTO 945
      IF(IRPLYO.EQ.1.AND.IAVFLG(K).EQ.81) GOTO 945
      IF(IRPLYO.EQ.2.AND.IAVFLG(K).GT.81) GOTO 945
      IF(IRPLYO.EQ.3.AND.IAVFLG(K).GE.80) GOTO 945
      GOTO 950
945   NUMBER=NUMBER+1
      IF(NUMBER.GE.1) GOTO 946
      WRITE(2,944,ERR=948)K,(IOPNAM(I),I=1,12),IARDT2,IARDT1,IARDT3
      GOTO 947
946   WRITE(2,953)K,(IOPNAM(I),I=1,12),IARDT2,IARDT1,IARDT3
947   WRITE(2,954)(IQFILE(I),I=1,3),(IARLOT(I),I=1,3),NELMTS,NREDCT
      NUMAVG=IAVFLG(K)-80
      WRITE(2,955)(ITITLE(I,K),I=1,24),NUMAVG
      WRITE(2,956)((BELNAM(I,J),I=1,2),(WTPCT(J,K)),J=1,NELMTS)
      WRITE(2,952)WTPCT(15,K)
      WRITE(2,957)((BSTAND(I,J),I=1,4),J=1,NELMTS)
950   CONTINUE
      CLOSE(UNIT=2)
      TYPE*,' '
      TYPE*,NUMBER,' ANALYSES WERE JUST FILED'
944   FORMAT(2H+$,I4,' OPERATOR: ',12A2,2X,'DATE: ',I2,'-',I2,'-19',I2)
953   FORMAT(2H $,I4,' OPERATOR: ',12A2,2X,'DATE: ',I2,'-',I2,'-19',I2)
954   FORMAT(8H  *QSET=,3A2,' DATA STORED IN ',3A2,2X,'NELMTS=',I2,3X,
     &'NREDCT=',I1)
955   FORMAT(2H  ,24A2,2X,'AVERAGE OF ',I2)
956   FORMAT(2H  ,14(2A1,F6.2))
952   FORMAT(2H  ,'SUM=',F6.2)
957   FORMAT(2H  ,4A1,13(1X,4A1))
      GOTO 990
959   TYPE*,'NONE FOUND'
      GOTO 990
C
C     OPTION 4, OXIDES, COMPLETED AUGUST 31, 1982
C
960   TYPE*,'YOU CHOSE THE FORMAT FOR OXIDE FORMULAS AND GEOTHERMOMERTY'
      DO 961, N=1,KOUNTR
```

Appendix II. Listing of Program RDARL4. (continued)

```
      WTPCT(16,N)=0.0
961   CONTINUE
      DO 965, I=1,13
      IFLAG=0
      DO 965, J=1,NELMTS
      IF(BELNAM(1,J).NE.BOXIDE(1,I)) GOTO 963
      IF(BELNAM(2,J).NE.BOXIDE(2,I)) GOTO 963
      KCOUNT(I)=J                                    !sequence elements put out
      IFLAG=1
963   IF(IFLAG.EQ.1) GOTO 965
      KCOUNT(I)=16                                   !if elem not present, wt%=0.0
965   CONTINUE
975   CALL SETUP(IARLOT,FILSPC)
      OPEN(UNIT=2,NAME=FILSPC,ERR=976,CARRIAGECONTROL='FORTRAN')
      GOTO 977
976   IERR=11
      CALL ERRORS(IERR,IREC)
      GOTO 975
977   WRITE(2,966,ERR=976) NTYPE
966   FORMAT('+',I1)
967   CALL SELECT(LOWER,UPPER,INDFLG,JAVFLG,KNWFLG)
      IF(INDFLG.NE.1) LOWER=1
      IF(INDFLG.NE.1) UPPER=KOUNTR
      DO 970, N=LOWER,UPPER
      IF(INDFLG.EQ.1) GOTO 968
      IF(JAVFLG.EQ.1.AND.IAVFLG(N).GT.1.AND.IAVFLG(N).LT.80) GOTO 968
      IF(KNWFLG.EQ.1.AND.IAVFLG(N).GE.80) GOTO 968
      GOTO 970
968   WRITE(2,971) N,IAVFLG(N),(ITITLE(L,N),L=1,24)
      DO 970, I=1,13
      WRITE(2,972)(BOXIDE(J,I),J=1,2),WTPCT(KCOUNT(I),N)
970   CONTINUE
971   FORMAT(I3,',',I2,',',24A2)
972   FORMAT(1X,2A1,',',F6.2)
      CALL CHOICE(ANSWER)
      IF(ANSWER.EQ.'Y') GOTO 967
      CLOSE(UNIT=2)
      GOTO 990
C
C     OPTION 9. BASIC SEQUENTIAL FILE THAT CAN BE CONCATENATED,
C          SEARCHED, OR EDITED. USE FOR ARCHIVING GROUPS OF ANALYSES.
C
1100  TYPE*,'YOU ELECTED A DISK FILE OF THE ELEMENTS ANALYZED AND IN THE'
      TYPE*,' SAME ORDER IN WHICH THEY APPEARED IN THE ARL PRINTOUT. THIS'
      TYPE*,'  FORMAT CAN BE SEARCHED, CONCATENATED, AND EDITED.'
1101  CALL SETUP(IARLOT,FILSPC)
      OPEN(UNIT=2,NAME=FILSPC,ERR=1102,CARRIAGECONTROL='FORTRAN')
      GOTO 1103
1102  IERR=11
      CALL ERRORS(IERR,IREC)
      GOTO 1101
1103  WRITE(2,1105,ERR=1102) NELMTS,PEROXY(1)
1105  FORMAT(1H+,I3,',',***,',',F6.3)
      DO 1110,J=1,NELMTS
      WRITE(2,1115) (BELNAM(I,J),I=1,2)
1110  CONTINUE
```

Appendix II. Listing of Program RDARL4. (continued)

```
1115  FORMAT(1X,2A1)
      WRITE(2,1116)
1116  FORMAT(6H TOTAL)
1120  CALL SELECT(LOWER,UPPER,INDFLG,JAVFLG,KNWFLG)
      IF(INDFLG.NE.1) LOWER=1
      IF(INDFLG.NE.1) UPPER=KOUNTR
      DO 1130, N=LOWER,UPPER
      IF(INDFLG.EQ.1) GOTO 1128
      IF(JAVFLG.EQ.1.AND.IAVFLG(N).GT.1.AND.IAVFLG(N).LT.80) GOTO 1128
      IF(KNWFLG.EQ.1.AND.IAVFLG(N).GT.80) GOTO 1128
      GOTO 1130
1128  WRITE(2,1131) NANAL(N),(ITITLE(L,N),L=1,24),PEROXY(N),IAVFLG(N)
      DO 1129,I=1,NELMTS
      WRITE(2,1132) WTPCT(I,N),CATION(I,N)
1129  CONTINUE
      WRITE(2,1132) WTPCT(15,N),CATION(15,N)
1130  CONTINUE
1131  FORMAT(1X,I3,',',24A2,',',F6.3,',',I3)
1132  FORMAT(1X,F6.2,',',F6.3)
      CALL CHOICE(ANSWER)
      IF(ANSWER.EQ.'Y') GOTO 1120
      CLOSE(UNIT=2)
      GOTO 990
C
C     BRANCH TO ANOTHER OPTION OR EXIT
C
990   TYPE*,' '
      TYPE*,'ANOTHER OUTPUT FILE OPTION? (Y or N)'
      ACCEPT 701, IRPLY6
      IF(IRPLY6.EQ.'Y') GOTO 801
1999  TYPE*,'READ A NEW ARLEB DATA FILE? (Y or N)'
      ACCEPT 701,IRPLY6
      IF(IRPLY6.EQ.'Y') GOTO 199
      STOP 'GOODBYE.  HAVE A NICE DAY.'
      END
C
C--SUBROUTINE ERRORS IDENTIFIES THE DISK ERROR--
C
      SUBROUTINE ERRORS(IERR,IREC)
      IF(IERR.GE.11) GOTO 5          !ARLEB errors <11; RT-11 errors >10
      TYPE*,'ERROR AT RECORD ',IREC,' ANALYSIS ',IREC-1     !ARLEB only
5     GOTO (10,20,30,40,50,60,70,80,90,100,110,120,130),IERR+1
10    STOP 'ILLEGAL DISC ADDRESS.  BEGIN AGAIN.'
20    STOP 'BAD READ/WRITE.  DEFECTIVE DISK?  BEGIN AGAIN.'
30    STOP 'PARITY ERROR;  DEFECTIVE DATA TRANSMISSION.  BEGIN AGAIN.'
40    STOP 'LIBRARY FULL (NO ROOM FOR FILE HEADER).'
50    STOP 'DISC FULL (NOT ENOUGH FREE SPACE).'
60    STOP 'NON-EXISTANT DATA RECORD REQUESTED. CHOOSE NEW NAME, BEGIN AGAIN.'
70    STOP 'DRIVE NOT READY.  CHECK AND CORRECT, THEN BEGIN AGAIN.'
80    STOP 'NON-EXISTANT DRIVE REQUESTED.'
90    TYPE*,'FILE NOT FOUND or WRONG FILE TYPE or INCORRECT DISK.'
      STOP 'CHANGE TO ARLEB DISK or CHOOSE NEW NAME.  PLEASE BEGIN AGAIN.'
100   STOP 'BAD FETCH OR FILE.  DENSITY INCORRECT?  PLEASE BEGIN AGAIN.'
110   TYPE*,'DISC IS NOT A VALID ARL BASIC SYSTEM (BOOTABLE) DISC.'
      STOP 'PLACE ANOTHER DISK IN THE RICHT-HAND DRIVE AND BEGIN AGAIN.'
120   TYPE*,'ERROR WHILE TRYING TO OPEN AN OUTPUT FILE.  MAKE SURE THAT A'
```

Appendix II. Listing of Program RDARL4. (continued)

```
      TYPE*,'PROPERLY FORMATTED DISK WITH FREE SPACE IS PRESENT AND THAT THE'
      TYPE*,'DISK DRIVE IS READY. CAREFULLY REENTER THE NAME OF THE OUTPUT'
      TYPE*,'FILE, FOLLOWING THE SUGGESTED SYNTAX.  IF YOU DO NOT SUCCEED,'
      TYPE*,'TERMINATE THE PROGRAM WITH A CONTROL C.'
      TYPE*,' '
130   TYPE*,'UNACCEPTABLE VALUES. CHECK SYNTAX, PLEASE TRY AGAIN.'
      TYPE*,' '
140   TYPE*,'ERROR WHILE WRITING. REOPEN THE OUTPUT FILE.'
      TYPE*,' '
      RETURN
      END
C
C--SUBROUTINE SETUP ACCEPTS FILENAME
C
      SUBROUTINE SETUP(IARLOT,FILSPC)
      INTEGER IARLOT(3)
      BYTE FILSPC(15)
      TYPE 200,(IARLOT(I),I=1,3)
200   FORMAT(29H THE ARL OUTPUT FILENAME WAS ,3A2,/)
210   TYPE 220
220   FORMAT(52H ENTER NAME OF FILE TO BE CREATED AS DY?:??????.??? )
      ACCEPT 230,FILSPC
230   FORMAT(15A1)
      IF(FILSPC(1).EQ.' ') GOTO 210
      FILSPC(15)=0
      TYPE*,' '
      RETURN
      END
C
C--SUBROUTINE SELECT LETS YOU SELECT ANALYSES TO BE PUT OUT
C
      SUBROUTINE SELECT(LOWER,UPPER,INDFLG,JAVFLG,KNWFLG)
      INTEGER UPPER
      COMMON BFLAG1,BFLAG2,BFLAG3
      INDFLG=0
      JAVFLG=0
      KNWFLG=0
      LOWER=0
      UPPER=0
      IF(BFLAG1.EQ.'Y') GOTO 323
      TYPE 300
300   FORMAT(1H0,'CHOOSE ANALYSES TO BE PUT OUT.')
      TYPE 301
301   FORMAT(1H0,'YOU WILL BE ASKED TO CHOSE BETWEEN AVERAGED UNKNOWN,')
      TYPE 302
302   FORMAT(1H ,'ALL KNOWN-UNKNOWN, AND THEN INDIVIDUAL ANALYSES.')
305   TYPE 310
310   FORMAT(1H0,'ALL AVERAGES OF UNKNOWN ANALYSES (Y or N)?')
      ACCEPT 320, BFLAG1
320   FORMAT(A1)
      IF(BFLAG1.NE.'Y'.AND.BFLAG1.NE.'N') GOTO 305
      IF(BFLAG1.EQ.'Y') JAVFLG=1
      IF(BFLAG1.EQ.'Y') GOTO 360
323   IF(BFLAG2.EQ.'Y') GOTO 330
324   TYPE 325
325   FORMAT(1H0,'ALL THE KNOWN-UNKNOWNS (Y or N)?')
```

Appendix II. Listing of Program RDARL4. (continued)

```
      ACCEPT 320, BFLAG2
      IF(BFLAG2.NE.'Y'.AND.BFLAG2.NE.'N') GOTO 324
      IF(BFLAG2.EQ.'Y') KNWFLG=1
      IF(BFLAG2.EQ.'Y') GOTO 360
330   TYPE 340
340   FORMAT(1H0,'Enter FIRST and LAST analysis numbers in a group ',/,
     &1H ,'of CONSECUTIVE analyses (I3,I3). Just type <RET> to exit.')
      ACCEPT 350,LOWER,UPPER
350   FORMAT(I4,I3)
      IF(LOWER.EQ.0.OR.UPPER.EQ.0) GOTO 355
      IF(LOWER.GT.100.OR.LOWER.LT.0) TYPE*,'First number too large or
     & small. Try again.'
      IF(UPPER.GT.100.OR.UPPER.LT.0) TYPE*,'Second number too large or
     & small. Try again.'
      IF((UPPER-LOWER).LT.0) TYPE*,'Second number too small. Try again.'
      IF((UPPER-LOWER).LT.0) GOTO 330
      IF(LOWER.GT.100.OR.LOWER.LT.0) GOTO 330
      IF(UPPER.GT.100.OR.UPPER.LT.0) GOTO 330
      INDFLG=1
      GOTO 360
355   TYPE 370
370   FORMAT(1H0,'YOU SELECTED NOTHING OR TYPED A ZERO. TRY AGAIN (Y or N)?')
      ACCEPT 320, BFLAG3
      IF(BFLAG3.EQ.'Y') GOTO 330
360   RETURN
      END
C
C--SUBROUTINE CHOICE ENABLES A SECOND SELECTION
C
      SUBROUTINE CHOICE(ANSWER)
409   TYPE*,'SELECT MORE ANALYSES FOR THIS OUTPUT FILE (Y or N)?'
      ACCEPT 410,ANSWER
410   FORMAT(A1)
      IF(ANSWER.NE.'Y'.AND.ANSWER.NE.'N') GOTO 409
      RETURN
      END
```

Appendix III. Examples of Output created by Program RDARL4.

a. Hardcopy printout.

```
RDARL4 ENABLES RT-11 TO ACCESS CHEMICAL ANALYSES STORED IN ARL'ER FORMAT BY THE RESTON ARL/SEMQ MICROPROBE
IT WAS WRITTEN BY J S HUEBNER FOR THE RESTON MICROPROBE FACILITY AND COMPLETED JUNE 15, 1983
RT-11 RUN DATE AND TIME WERE  6-28-1983 AT 15:28
MICROPROBE OPERATOR: HUEBNER -  HD & GNT                          MICROPROBE ANALYSIS DATE 10-29-1982
THE TITLES OF THE *OSET INPUT AND *ANALZ OUTPUT FILES WERE *GARY  and D1027

   ELEMENT      X-LINE       STANDARD
     FE           KA           PXHB
     SI           KA           PXAB
     MG           KA           PXAB
     CA           KA           PXAB
     AL           KA           FSTA
     NA           KA           FSTA
     TI           KA           OXIL
     CR           KA           OXTB
     MN           KA           OLST
ANALYSES STORED AS OXIDE WEIGHT PERCENT
DATA REDUCTION BY METHOD OF BENCE & ALBEE

AVERAGES OF UNKNOWN ANALYSES ONLY:

 10 TITLE: HD BALLAL LAKE USNM R3005
      FE      SI      MG      CA      AL      NA      TI      CR      MN                                      TOTAL
WT % 3.405  54.037  16.697  25.389   0.864   0.261   0.005   0.018   0.334   0.674   0.000   0.000   0.000   0.000  101.009              AV OF  4
CAT  0.103   1.964   0.904   0.989   0.037   0.018   0.000   0.001   0.010   0.059   0.000   0.000   0.000   0.000    4.027 PER   6.000 ANIONS

 16 TITLE: HD BUCKTOWN USNM 119197
      FE      SI      MG      CA      AL      NA      TI      CR      MN                                      TOTAL
WT % 5.693  53.549  15.044  24.184   0.399   0.495   0.028   0.014   1.176   0.674   0.000   0.000   0.000   0.000  100.582              AV OF  5
CAT  0.176   1.978   0.828   0.957   0.017   0.035   0.001   0.000   0.037   0.037   0.000   0.000   0.000   0.000    4.030 PER   6.000 ANIONS

 24 TITLE: H UVAROVITE USNM 139715 ZONED
      FE      SI      MG      CA      AL      NA      TI      CR      MN                                      TOTAL
WT % 2.189  38.217   0.637  33.043   7.937   0.143   0.043  18.580   0.836   0.674   0.000   0.000   0.000   0.000  101.625              AV OF  9
CAT  0.072   1.513   0.038   1.401   0.370   0.011   0.001   0.581   0.028   0.021   0.000   0.000   0.000   0.000    4.016 PER   6.000 ANIONS

 29 TITLE: KAK GNT
      FE      SI      MG      CA      AL      NA      TI      CR      MN                                      TOTAL
WT %10.617  42.296  19.185   5.111  23.571   0.159   0.409   0.081   0.487   0.674   0.000   0.000   0.000   0.000  101.915              AV OF  4
CAT  0.312   1.489   1.006   0.193   0.978   0.011   0.011   0.002   0.015   0.020   0.000   0.000   0.000   0.000    4.016 PER   6.000 ANIONS

INDIVIDUALLY SELECTED ANALYSES OF ALL TYPES:

  1 TITLE: HD TEST                                                X =  8893.0     Y =  37153.5
      FE      SI      MG      CA      AL      NA      TI      CR      MN                                      TOTAL
WT %23.134  48.486   4.257  20.762   0.083   0.474   0.027   0.002   2.542   0.674   0.000   0.000   0.000   0.000   99.767
CAT  0.786   1.969   0.258   0.904   0.004   0.037   0.001   0.000   0.087   0.353   0.000   0.000   0.000   0.000    4.046 PER   6.000 ANIONS

  2 TITLE: HD TEST                                                X =  8893.0     Y =  37153.5
      FE      SI      MG      CA      AL      NA      TI      CR      MN                                      TOTAL
WT %22.820  49.072   4.206  20.753   0.499   0.469   0.005   0.005   2.530   0.674   0.000   0.000   0.000   0.000  100.359
CAT  0.767   1.973   0.252   0.894   0.024   0.037   0.000   0.000   0.086   0.224   0.000   0.000   0.000   0.000    4.033 PER   6.000 ANIONS

  3 TITLE: HD TEST                                                X =  8893.0     Y =  37153.5
      FE      SI      MG      CA      AL      NA      TI      CR      MN                                      TOTAL
WT %22.979  49.986   4.111  20.852   0.112   0.454   0.000   0.033   2.627   0.674   0.000   0.000   0.000   0.000  101.154
CAT  0.766   1.992   0.244   0.890   0.005   0.035   0.000   0.001   0.089   0.164   0.000   0.000   0.000   0.000    4.022 PER   6.000 ANIONS

  4 TITLE: HD TEST                                                X =  8893.0     Y =  37153.5
      FE      SI      MG      CA      AL      NA      TI      CR      MN                                      TOTAL
WT %23.137  49.733   3.881  20.754   0.141   0.452   0.008   0.000   2.622   0.674   0.000   0.000   0.000   0.000  100.727
CAT  0.775   1.993   0.232   0.891   0.007   0.035   0.000   0.000   0.089   0.128   0.000   0.000   0.000   0.000    4.021 PER   6.000 ANIONS

  5 TITLE: HD M12330 AV OF 4
      FE      SI      MG      CA      AL      NA      TI      CR      MN                                      TOTAL
WT %23.017  49.320   4.114  20.780   0.209   0.462   0.009   0.008   2.580   0.674   0.000   0.000   0.000   0.000  100.500              AV OF  4
CAT  0.774   1.982   0.246   0.895   0.010   0.036   0.000   0.000   0.088   0.105   0.000   0.000   0.000   0.000    4.031 PER   6.000 ANIONS
```

Appendix III. Examples of Output created by Program RDARL4.

b. Printout of sequential disk file for auxillary program MINCLC

```
****RDOXIONS                                                    6.        4.
 10 TITLE: HD BAILAL LAKE USNM R3005
FE  3.41SI 54.04MG 16.70CA 25.39AL  0.86NA  0.26TI  0.00CR  0.02
MN  0.33$$
 16 TITLE: HD DUCKTOWN USNM 119197
FE  5.69SI 53.55MG 15.04CA 24.18AL  0.40NA  0.50TI  0.03CR  0.01
MN  1.18$$
 24 TITLE: H UVAROVITE USNM 139715 ZONED
FE  2.19SI 38.22MG  0.64CA 33.04AL  7.94NA  0.14TI  0.04CR 18.58
MN  0.84$$
 29 TITLE: KAK GNT
FE 10.62SI 42.30MG 19.18CA  5.11AL 23.57NA  0.16TI  0.41CR  0.08
MN  0.49$$
  1 TITLE: HD TEST
FE 23.13SI 48.49MG  4.26CA 20.76AL  0.08NA  0.47TI  0.03CR  0.00
MN  2.54$$
```

c. Printout of sequential disk file for auxillary Program PXVOID

```
1
 10 HD BAILAL LAKE USNM R3005
SI 54.04AL  0.86TI  0.00CR  0.02FC  0.00FE  3.41
MG 16.70MN  0.33CA 25.39LI  0.00NA  0.26K   0.00
 16 HD DUCKTOWN USNM 119197
SI 53.55AL  0.40TI  0.03CR  0.01FC  0.00FE  5.69
MG 15.04MN  1.18CA 24.18LI  0.00NA  0.50K   0.00
 24 H UVAROVITE USNM 139715 ZONED
SI 38.22AL  7.94TI  0.04CR 18.58FC  0.00FE  2.19
MG  0.64MN  0.84CA 33.04LI  0.00NA  0.14K   0.00
 29 KAK GNT
SI 42.30AL 23.57TI  0.41CR  0.08FC  0.00FE 10.62
MG 19.18MN  0.49CA  5.11LI  0.00NA  0.16K   0.00
  5 HD M12330 AV OF 4
SI 49.32AL  0.21TI  0.01CR  0.01FC  0.00FE 23.02
MG  4.11MN  2.58CA 20.78LI  0.00NA  0.46K   0.00
 29 KAK GNT
SI 42.30AL 23.57TI  0.41CR  0.08FC  0.00FE 10.62
MG 19.18MN  0.49CA  5.11LI  0.00NA  0.16K   0.00
```

Appendix III. Examples of Output created by Program RDARL4.

d. Printout of sequential disk file for auxillary program RECAMP

```
  1 HD TEST
NA  0.47MG  4.26AL  0.08SI 48.49K   0.00CA 20.76TI  0.03MN  2.54FE 23.13
  2 HD TEST
NA  0.47MG  4.21AL  0.50SI 49.07K   0.00CA 20.75TI  0.01MN  2.53FE 22.82
  3 HD TEST
NA  0.45MG  4.11AL  0.11SI 49.99K   0.00CA 20.85TI  0.00MN  2.63FE 22.98
  4 HD TEST
NA  0.45MG  3.88AL  0.14SI 49.73K   0.00CA 20.75TI  0.01MN  2.62FE 23.14
  5 HD M12330 AV OF 4
NA  0.46MG  4.11AL  0.21SI 49.32K   0.00CA 20.78TI  0.01MN  2.58FE 23.02
 25 KAK GNT
NA  0.15MG 19.06AL 23.61SI 42.38K   0.00CA  4.99TI  0.48MN  0.50FE 10.63
  1 HD TEST
NA  0.47MG  4.26AL  0.08SI 48.49K   0.00CA 20.76TI  0.03MN  2.54FE 23.13
```

e. Printout of sequential disk file for auxillary Program OXCALC

```
1 ·
10, 4,HD BAILAL LAKE USNM R3005
SI, 54.04
TI,  0.00
AL,  0.86
V ,  0.00
CR,  0.02
FC,  0.00
FE,  3.41
MN,  0.33
MG, 16.70
CA, 25.39
ZN,  0.00
NI,  0.00
NB,  0.00
16, 5,HD DUCKTOWN USNM 119197
SI, 53.55
TI,  0.03
AL,  0.40
V ,  0.00
CR,  0.01
FC,  0.00
FE,  5.69
MN,  1.18
MG, 15.04
CA, 24.18
ZN,  0.00
NI,  0.00
NB,  0.00
24, 9,H UVAROVITE USNM 139715 ZONED
```

Appendix IIIf. Description of BASIC virtual array file.

The direct access file or BASIC virtual array created by Option 5 is written in internal (binary) computer format; its contents cannot readily be examined or edited. The record length in 2-byte words is

LENGTH = 4* (NELMTS + 1) + 3

Each reecord includes the number of analyses in the file (KOUNTR) and the number of elements analyzed (NELMTS). Thus, the length of the file can be computed once the first record is read using a default record size of 63, the maximum size that can be written. Data for each record is contained in the following sequence:

KOUNTR	
NANAL (I)	
NELMTS	
WTPCT	up to 14 value depending upon NELMTS
WTPCT	total of weight precent values
CATION	up to 14 values
CATION	sum of cation numbers

g. BASIC Sequential File for statistics

```
 5, 9
 1
23.13
48.49
 4.26
20.76
 0.08
 0.47
 0.03
 0.00
 2.54
 2
22.82
49.07
 4.21
20.75
 0.50
 0.47
 0.01
 0.00
 2.53
 3
22.98
49.99
 4.11
20.85
 0.11
 0.45
 0.00
 0.03
 2.63
 4
23.14
```

Appendix IIIh. BASIC Sequential File or auxillary programs used to calculate mineral formula of amphiboles, micas, cordierite, feldspars, chlorite, and garnets.

```
1
SI
TI
AL
FC
CR
FE
ZN
MN
NI
MG
CA
SR
BA
NA
K
P
F
CL
 10,HD BAILAL LAKE USNM R3005
*GAPY , 6.000
 54.04, 1.964
  0.00, 0.000
  0.86, 0.037
  0.00, 0.000
  0.02, 0.001
  3.41, 0.103
  0.00, 0.000
  0.33, 0.010
  0.00, 0.000
 16.70, 0.904
 25.39, 0.989
  0.00, 0.000
  0.00, 0.000
  0.26, 0.018
  0.00, 0.000
  0.00, 0.000
  0.00, 0.000
  0.00, 0.000
 16,HD DUCKTOWN USNM 119197
*GAPY , 6.000
 53.55, 1.978
  0.03, 0.001
  0.40, 0.017
  0.00, 0.000
  0.01, 0.000
  5.69, 0.176
  0.00, 0.000
  1.18, 0.037
  0.00, 0.000
 15.04, 0.828
 24.18, 0.957
```

Appendix IIIi. Printout of disk file for storing "known-unknown" microprobe analyses with Option 8 of Program RDARL4.

```
#    1 OPERATOR: HUEBNER -= HD & GNT          DATE: 10-29-1982
 *QSET=*GAPY  DATA STORED IN D1027   NELMTS= 9   NREDCT=2
 HD TEST                                          AVERAGE OF  0
 FE 23.13SI 48.49MG  4.26CA 20.76AL  0.08NA  0.47TI  0.03CR  0.00MN  2.54
 SUM= 99.77
 PXHD PXAD PXAD PXAD FSTA FSTA OXIL OXTB OLST
#    2 OPERATOR: HUEBNER -= HD & GNT          DATE: 10-29-1982
 *QSET=*GAPY  DATA STORED IN D1027   NELMTS= 9   NREDCT=2
 HD TEST                                          AVERAGE OF  0
 FE 22.82SI 49.07MG  4.21CA 20.75AL  0.50NA  0.47TI  0.01CR  0.00MN  2.53
 SUM=100.36
 PXHD PXAD PXAD PXAD FSTA FSTA OXIL OXTB OLST
#    3 OPERATOR: HUEBNER -= HD & GNT          DATE: 10-29-1982
 *QSET=*GAPY  DATA STORED IN D1027   NELMTS= 9   NREDCT=2
 HD TEST                                          AVERAGE OF  0
 FE 22.98SI 49.99MG  4.11CA 20.85AL  0.11NA  0.45TI  0.00CR  0.03MN  2.63
 SUM=101.15
 PXHD PXAD PXAD PXAD FSTA FSTA OXIL OXTB OLST
#    4 OPERATOR: HUEBNER -= HD & GNT          DATE: 10-29-1982
 *QSET=*GAPY  DATA STORED IN D1027   NELMTS= 9   NREDCT=2
 HD TEST                                          AVERAGE OF  0
 FE 23.14SI 49.73MG  3.88CA 20.75AL  0.14NA  0.45TI  0.01CR  0.00MN  2.62
 SUM=100.73
 PXHD PXAD PXAD PXAD FSTA FSTA OXIL OXTB OLST
#    5 OPERATOR: HUEBNER -= HD & GNT          DATE: 10-29-1982
 *QSET=*GAPY  DATA STORED IN D1027   NELMTS= 9   NREDCT=2
 HD M12330 AV OF 4                                AVERAGE OF  4
 FE 23.02SI 49.32MG  4.11CA 20.78AL  0.21NA  0.46TI  0.01CR  0.01MN  2.58
 SUM=100.50
 PXHD PXAD PXAD PXAD FSTA FSTA OXIL OXTB OLST
#   25 OPERATOR: HUEBNER -= HD & GNT          DATE: 10-29-1982
 *QSET=*GAPY  DATA STORED IN D1027   NELMTS= 9   NREDCT=2
 KAK GNT                                          AVERAGE OF  0
 FE 10.63SI 42.38MG 19.06CA  4.99AL 23.61NA  0.15TI  0.48CR  0.10MN  0.50
 SUM=101.89
 PXHD PXAD PXAD PXAD FSTA FSTA OXIL OXTB OLST
```

Appendix IIIj: Printout of sequential file for archiving, concatenating, and editing analyses

```
   9,***, 6.000
 FE
 SI
 MG
 CA
 AL
 NA
 TI
 CR
 MN
 TOTAL
  10,HD BAILAL LAKE USNM R3005                          , 6.000,  4
   3.41, 0.103
  54.04, 1.964
  16.70, 0.904
  25.39, 0.989
   0.86, 0.037
   0.26, 0.018
   0.00, 0.000
   0.02, 0.001
   0.33, 0.010
 101.01, 4.027
  16,HD DUCKTOWN USNM 119197                            , 6.000,  5
   5.69, 0.176
  53.55, 1.978
  15.04, 0.828
  24.18, 0.957
   0.40, 0.017
   0.50, 0.035
   0.03, 0.001
   0.01, 0.000
   1.18, 0.037
 100.58, 4.030
  24,H UVAROVITE USNM 139715 ZONED                      , 6.000,  9
   2.19, 0.072
  38.22, 1.513
   0.64, 0.038
  33.04, 1.401
   7.94, 0.370
   0.14, 0.011
   0.04, 0.001
  18.58, 0.581
   0.84, 0.028
 101.63, 4.016
  29,KAK GNT                                            , 6.000,  4
  10.62, 0.312
  42.30, 1.489
  19.18, 1.006
   5.11, 0.193
  23.57, 0.978
```

Appendix IV. Printout of instructions contained in Reston Disk File

(Press the NO SCROLL key at the lower left of the keyboard to halt the motion of the display; press the key again to advance the display.)

WELCOME TO RDARL4!

RDARL4 is a Fortran program that enables the MINC computer to read the data files written by the ARLEB operating system in the ARL-SEMQ 'probe. For input you must use a disk that contains data written in the abbreviated storage format coded by Jim McGee that appears in an SEMQ operating program such as $ANBA. (The lengthy storage format provided by ARL is not compatible with RDARL4!) RDARL4 will coach you by asking a number of questions; your responses will be alphabetic characters (Y, N, or C), integer numbers (e.g., I2 calls for a one- or two-digit number without decimal point), rarely a floating point number (6.0 or 4.0 are acceptable for F12.7 format), and file names. For disk output, you will need a formatted MINC disk (see the INITIALIZE command, MINC Programming reference, volume 3). You will also need to have made some decisions in advance. Specifically, if you elect to create an output file on disk, you will be asked what use will be made of the disk output (RDARL4 takes care of preparing the correct format for that use). All options will ask you to select the analyses that you wish included in the disk output file. In addition, the option that writes a disk file for MINC regressions needs to know how many analyses you will select. Your current choices for output are:

Option#	
	HARD-COPY PRINTOUT
1	MINCLC FORTRAN INPUT DISK FILE
2	PXVOID " " "
3	RECAMP " " "
5	MINC-BASIC VIRTUAL ARRAY ON DISK
	MINC BASIC SEQUENTIAL FILES ON DISK for:
6	REGRESSION PROGRAMS on the MINC APPLICATIONS PACKAGE disk)
7	SILICATE FORMULA RECALCULATION
4	OXIDE FORMULA RECALCULATION
8	FORTRAN SEQUENTIAL DISK FILE OF KNOWN-UNKNOWN ANALYSES
9	BASIC SEQUENTIAL DISK FILE FOR EDITING, CONCATENATING, ARCHIVING

If you plan to select more than one kind of output file, you will need to specify different output file names. One way to do this is to use a file name that embeds a mnemonic abbreviation (MIN, PXV, REC, OXD, etc.) for the kind of file. For example, DY1:DA709.MIN would be a file created for the mineral formula program MINCLC.

Now set the computer system's current date and time by using the following format after the monitor's prompt (.):

.DATE 19-AUG-82<return>

.TIME 13:49<return>

If you intend to display the data on this video screen, you can prevent 'wraparound' by pressing the keys SETUP, 9, SETUP to change the number of characters per line from 78 to 130. However, in so doing you will erase the current screen display, so read the next three lines first!

To run the program, type RUN RDARL4<return>.
To interrupt the program, type ^C twice. (^C means 'control C')

Have Fun!

Steve Huebner
June 10, 1983

Table 1. Structure of a file of chemical analyses created by microprobe

First Record Contains Parameters Common to All Analyses

Entry Number	Contents	RDARL4 Variable or Array Name
1 - 4	Operator's Name	IOPNAM
5 - 7	ARL System Date	IARDT
8	Name of *QFILE	IQFILE
9	Name of Data Output File	IARLOT
10	Data Reduction Flag	NREDCT
11	Number of Elements Analyzed	NELMTS
12	Oxide or Element Wt. % Flag	NTYPE
13	Temporary Counter Used by ARL	NCOUNT
14	Future Use	NOUSED
15 - 28	Element Names, X-ray Lines (up to 14)	BELNAM,BXRLIN
29 - 42	Standard Names (up to 14)	BSTAND

Subsequent Records Contain One Analysis Each

Entry Number	Contents	RDARL4 Array Name
1 - 8	Title	ITITLE
9	Stage Position X	X
10	Stage Position Y	Y
11	Averaging & Known-Unknown Flag	IAVFLG
12 - 25	Weight Percent Values (14)	WTPCT
26	Total Weight Percent	WTPCT
27 - 40	Cation Numbers (14)	CATION
41	Sum of Cations	CATION
42	Number Anions Per Formula Unit	PEROXY