PLUMBO:

A Hewlett-Packard Series 200 BASIC Language Program for

Version IV of Plumbotectonics

Sara M. Haines

and

Robert E. Zartman

U.S. Geological Survey
Mailstop 963, Denver Federal Center
Denver, Colorado 80225

UNITED STATES GEOLOGICAL SURVEY

OPEN-FILE REPORT 88-269

This report in preliminary and has not been reviewed for conformity with U.S. Geological Survey editorial standards and nomenclature. Any use of trade names is for descriptive purposes only, and does not imply endorsement by the U.S. Geological Survey.

TABLE OF CONTENTS

INTRODUCTION

In a concurrent paper, Zartman and Haines (in press) present version IV of Plumbotectonics, which pursues the objective of explaining Pb, Sr, and Nd isotopic systematics within major terrestial reservoirs. Version IV expands and refines the earlier versions of the model published previously by Doe and Zartman (1979), Zartman and Doe (1981), and Zartman (1984). The actual BASIC computer program for Plumbotectonics, called PLUMBO, is not included in Zartman and Haines (in press), but is made available here for persons wanting to repeat our calculations or devise their own modifications of the model. In this report we will document the current design, construction, and operation of PLUMBO. A full listing of the BASIC program is given in Appendix A.

For those who wish to change parameters set forth in version IV, we offer a step-by-step procedure for accessing and using PLUMBO. The program was designed to accomodate some variations of model assumptions, and to test those variations without editing the main program. This is accomplished by setting up a file using VisiCalc (Hewlett-Packard spreadsheet, copyright 1983, or any similar BDAT-form text file, such as CONTEXT MBA, also available from Hewlett-Packard) to store starting parameters. PLUMBO also allows changes in a few parameters by direct access through the keyboard of the computer. After the calculations are completed by PLUMBO, various datafiles can be saved for ISOPLOT200 (Ludwig, 1985) to print out and graph.

Plumbotectonics treats the interaction between major reservoirs, mantle, upper crust, lower crust, and subcrustal lithosphere (subcrust)[1], as cycles--called orogenies--of discrete, sequential operations that can be expressed by a series of BASIC program statements. These HP BASIC statements parallel the algebraic equations in the Appendix of Zartman and Haines (in press; see conversion tables 5(a)-5(d) in the Program Explanation section for a comparison between notational styles). Mathematically derived and stored in the form of numeric arrays, both mass distribution and isotopic contents of the reservoirs can be determined as a function of time. Significant changes over earlier versions of Plumbotectonics that have been adopted for version IV are (1) a greater time resolution, and (2) a subdivision of the orogene. Accordingly, by using 100 Ma instead of 400 Ma time intervals between orogenies, the model can better represent fine-structure in Earth history. By dividing the orogene into proximal, distal, and mantle wedge components capable of maintaining separate chemical and isotopic identities, the model can more realistically mimic plate tectonic processes.

A disk copy of PLUMBO and the VisiCalc template used by PLUMBO can be obtained by sending a 3.5" or 5.25" single-sided floppy disk to Branch of Isotope Geology, Mailstop 963, Denver Federal Center, Denver, CO 80225.

---

[1] In this report we adopt the convention of underlining the names of terrestrial reservoirs when referring to their model equivalents. Although in computer-generated tables and figures this convention may not be followed, it should be assumed that such labelled or listed reservoir are the model analogue, unless otherwise noted.

GETTING STARTED

WHAT'S REQUIRED TO RUN PLUMBO

Three main programs are needed to make the model calculations and view the results: commercially-available spreadsheet **VisiCalc** (Hewlett-Packard, copyright 1983), **PLUMBO** for the model calculations, and **ISOPLOT200** (Ludwig, 1985) for data retrieval and graphic capabilities. Individually, each program has its own minimum hardware and software requirements, but to run all three in succession, the following hardware is required:

--Hewlett-Packard (HP) Series 200 computer
--Dual single-sided 3.5" disk drive (Model HP-9121, assumed to be at address 700). 5.25" drives can be used for all three programs with minor adjustments to **PLUMBO**.
--Printer attached to HP-IB interface (assumed address, 701)
--HP7475 6-Pen or HP7470 2-pen plotter (assumed address, 705)

You should have at least 750 kilobytes of RAM (Random Access Memory) before loading the BASIC language. HP BASIC 2.1 or 3.x is required to run **ISOPLOT200** and **PLUMBO**. If you are using the BASIC 2.1 version of the language, you must have AP2_1 and GRAPH2_1 BASIC language extensions loaded into the memory. For BASIC 3.x version of the language, GRAPH, GRAPHX, MAT, CLOCK, KBD, ERR, KNB2_0, CS80, HPIB, and MS must be in memory.

In one run, **PLUMBO** generates approximately 30,000 output values, which may be saved as datafiles for subsequent use with **ISOPLOT200**. For saving all the datafiles, you will need two empty 3.5" single-sided floppy disks.

The next several sections outline the step-by-step procedure for accessing each program (**VisiCalc**, **PLUMBO**, and **ISOPLOT200**), and provides instruction on how to use each one for making your own model calculations.

VisiCalc

The procedure for accessing and using VisiCalc is outlined below. A VisiCalc spreadsheet, called TEMPLATE, was created to define and make changes in the starting parameters of the model. An ASCII file (text file) of TEMPLATE is saved using VisiCalc, which is then retrieved by PLUMBO for the calculations.

LOADING VisiCalc

Put the VisiCalc ENVIRONMENT disk in either drive and the VisiCalc PROGRAM disk in the other drive. If the computer is OFF, turn ON by pressing the black button at the back of the computer. If the computer in ON and has BASIC loaded into its memory, type SYSBOOT and press EXECUTE. (You can tell if BASIC is loaded by pressing SHIFT+RESET. The computer will respond by printing "BASIC reset" on the CRT if BASIC is already loaded.)

Wait until the computer is ready (about a minute). First, you will be asked to enter a new system date. Enter the date, so the files that are stored using VisiCalc will record the date they were stored. Next type N when asked "Extended system hardware configuration? (Y or N)." Finally, type VC:VC in response to the "Which Program?" query and wait until the VisiCalc spreadsheet appears.

LOADING A FILE

Now you are ready to load the template used for creating a file of starting parameters, which PLUMBO will use.

Remove the VisiCalc disks and place the disk containing the file TEMPLATE (or VERSIONIV) in one of the drives. Before loading either file, you will need to tell VisiCalc which disk to retrieve the files from. To do this, use the /SV command, and then tell VisiCalc what the new Volume Name is (e.g. VFILES). If you are not sure of the Volume Name of your disk, load BASIC (see following section for loading PLUMBO for instructions on how to load BASIC) and catalog the disk by typing CAT. Here BASIC will tell you the volume name of your disk along with the list of files stored on the disk. To return to VisiCalc, if you are in BASIC, follow the above instructions for loading VisiCalc.

To load the template you wish to edit for your own model starting parameters, type /SL and the filename (e.g. /SL TEMPLATE or /SL VERSIONIV).

FILE CONSTRAINTS

With TEMPLATE (Table 1) loaded, you can create your own starting parameters, or with VERSIONIV (Table 2) you can edit the starting parameters set up for version IV (Zartman and Haines, in press). Enter the values for enrichment factors, fractionation factors, decay constants, starting isotopic compositions, starting elemental abundances, and mantle contribution factors. For the time dependent values (mantle contribution and erosion factors and crustal formation), you will only need to fill in the column labelled STD MODEL. All the other columns will be computed for you. Even though allowance is made to change the starting variables of the model, it is imperative to edit and give values only as requested by the template in order for the file you stored using VisiCalc to be compatible with PLUMBO. You should be aware of the following basic constraints.

Table 1. Printout of TEMPLATE while in VisiCalc.

PARAMETERS FOR MODEL: TEMPLATE

E_a2 : r->d enrichment, MORidge to Dorogene
E_b1 : d->p enrichment, Dorogene to Porogene
E_b2 : d->w enrichment, Dorogene to Worogene
E_b3 : w->p enrichment, Worogene to Porogene
F_a3 : v->d,p fractionation of vertically-eroded Ucrust between Dorogene & Porogene
F_c3 : p->u,l fractionation of Porogene between new segments of Ucrust & Lcrust

Isotope Name	ENRICHMENT FACTORS E_a2(H)	E_b1(H)	E_b2(H)	E_b3(H)	FRACTIONATION FACTORS F_a3(H)	F_c3(H)
204 Pb						
206 Pb						
207 Pb						
208 Pb						
232 Th						
238 U						
86 Sr						
87 Sr						
87 Rb						
144 Nd						
143 Nd						
147 Sm						

Decay Constants	Start Iso Comps	Start El Abund
238 U	206/204	Mass
235 U	207/204	204 Pb
232 Th	208/204	232 Th
87 Rb	87/86	238 U
147 Sm	143/144	86 Sr
		87 Rb
		144 Nb
		147 Sm

Table 1. Continued.

Time	STAND MODEL	FUNC. MantContr	FOR Old Values	FRACTION Mantle Contrib.	FRACTION Horiz.ero U & L	FRACTION (E+24 g) Horiz.ero Subcrust	(E+24 g) Create Upper	(E+24 g) Create Lower	(E+24 g) Create Subcrust	ERO MASS FRACTION
4.5	0	0		ERROR	1	1ERROR	ERROR	ERROR		1
4.4	0	0		ERROR	1	1ERROR	ERROR	ERROR		1
4.3	0	0		ERROR	1	1ERROR	ERROR	ERROR		1
4.2	0	0		ERROR	1	1ERROR	ERROR	ERROR		1
4.1	0	0		ERROR	1	1ERROR	ERROR	ERROR		1
4	0	0	.0039063	ERROR	.5	.5ERROR	ERROR	ERROR		.525
3.9	0	0	.0039063	ERROR	.25	.25ERROR	ERROR	ERROR		.2875
3.8	0	0	.0039063	ERROR	.125	.125ERROR	ERROR	ERROR		.16875
3.7	0	0	.0039063	ERROR	.0625	.0625ERROR	ERROR	ERROR		.109375
3.6	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
3.5	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
3.4	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
3.3	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
3.2	0	0	.015625	ERROR	0	0ERROR	ERROR	ERROR		.05
3.1	0	0	.015625	ERROR	0	0ERROR	ERROR	ERROR		.05
3	0	0	.015625	ERROR	0	0ERROR	ERROR	ERROR		.05
2.9	0	0	.015625	ERROR	0	0ERROR	ERROR	ERROR		.05
2.8	0	0	.015625	ERROR	0	0ERROR	ERROR	ERROR		.05
2.7	0	0	.015625	ERROR	0	0ERROR	ERROR	ERROR		.05
2.6	0	0	.015625	ERROR	0	0ERROR	ERROR	ERROR		.05
2.5	0	0	.015625	ERROR	0	0ERROR	ERROR	ERROR		.05
2.4	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
2.3	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
2.2	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
2.1	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
2	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
1.9	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
1.8	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
1.7	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
1.6	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
1.5	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
1.4	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
1.3	0	0	.0078125	ERROR	0	0ERROR	ERROR	ERROR		.05
1.2	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
1.1	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
1	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
.9	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
.8	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
.7	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
.6	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
.5	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
.4	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
.3	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
.2	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
.1	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
0	0	0	.0039063	ERROR	0	0ERROR	ERROR	ERROR		.05
Total		0	0	.3164063	ERROR	5.9375	5.9375ERROR	ERROR	ERROR	
CFAC		ERROR	ERROR	ERROR		4.2E-5	4.2E-5ERROR	ERROR	ERROR	

Table 2. Printout of VERSIONIV while in VisiCalc.

PARAMETERS FOR MODEL: Version IV

E_a2 : r->d enrichment, MORidge to Dorogene
E_b1 : d->p enrichment, Dorogene to Porogene
E_b2 : d->w enrichment, Dorogene to Worogene
E_b3 : w->p enrichment, Worogene to Porogene
F_a3 : v->d,p fractionation of vertically-eroded Ucrust between Dorogene & Porogene
F_c3 : p->u,l fractionation of Porogene between new segments of Ucrust & Lcrust

Isotope Name	ENRICHMENT FACTORS E_a2(H)	E_b1(H)	E_b2(H)	E_b3(H)	FRACTIONATION FACTORS F_a3(H)	F_c3(H)
204 Pb	80.00	40.00	25.00	25.00	1.00	3.67
206 Pb	80.00	40.00	25.00	25.00	1.00	3.67
207 Pb	80.00	40.00	25.00	25.00	1.00	3.67
208 Pb	80.00	40.00	25.00	25.00	1.00	3.67
232 Th	100.00	47.00	25.00	25.00	0.67	4.00
238 U	100.00	52.00	27.00	27.00	1.50	6.25
86 Sr	12.00	8.00	5.00	5.00	1.00	1.00
87 Sr	12.00	8.00	5.00	5.00	1.00	1.00
87 Rb	50.00	23.00	10.00	10.00	1.00	9.00
144 Nd	10.00	6.00	5.00	5.00	1.00	2.33
143 Nd	10.00	6.00	5.00	5.00	1.00	2.33
147 Sm	8.00	5.00	3.00	3.00	1.00	2.25

Decay Constants		Start Iso Comps		Start El Abund	
238 U	.155125	206/204	9.21	Mass	1050
235 U	.98485	207/204	9.95	204 Pb	19
232 Th	.049475	208/204	28.85	232 Th	758
87 Rb	.0142	87/86	.69896	238 U	177
147 Sm	.00654	143/144	.50675	86 Sr	22000
				87 Rb	2140
				144 Nb	2100
				147 Sm	426

Table 2. Continued.

Time	STAND MODEL	FUNC. MantContr	FOR Old Values	Mantle Contrib.	FRACTION Horiz.ero U & L	FRACTION Horiz.ero Subcrust	FRACTION (E+24 g) Create Upper	(E+24 g) Create Lower	(E+24 g) Create Subcrust	ERO MASS FRACTION
4.5	0	0		0	1	1	0	0	0	1
4.4	0	0		0	1	1	0	0	0	1
4.3	0	0	.0039063	0	1	1	0	0	0	1
4.2	1.7	147.9	.0039063	.0719068	1	1	.1725762	.1725762	.295845	1
4.1	3.2	275.2	.0039063	.1337981	1	1	.3211155	.3211155	.5504837	1
4	4.5	382.5	.0039063	.1859658	.5	.5	.4463179	.4463179	.7651164	.525
3.9	5.6	470.4	.0039063	.2287014	.25	.25	.5488835	.5488835	.9409431	.2875
3.8	6.5	539.5	.0039063	.2622968	.125	.125	.6295124	.6295124	1.079164	.16875
3.7	7.2	590.4	.0039063	.2870437	.0625	.0625	.6889048	.6889048	1.18098	.109375
3.6	7.7	623.7	.0078125	.3032336	.0261954	.0261954	.7277607	.7277607	1.24759	.0748856
3.5	8	640	.0078125	.3111584	.02688	.02688	.7467802	.7467802	1.280195	.075536
3.4	8.1	639.9	.0078125	.3111098	.0268758	.0268758	.7466636	.7466636	1.279995	.0755320
3.3	8	624	.0078125	.3033795	.026208	.026208	.7281107	.7281107	1.24819	.0748976
3.2	7.8	600.6	.015625	.2920027	.0252252	.0252252	.7008066	.7008066	1.201383	.0739639
3.1	7.6	577.6	.015625	.2808205	.0242592	.0242592	.6739692	.6739692	1.155376	.0730462
3	7.5	562.5	.015625	.2734791	.023625	.023625	.6563498	.6563498	1.125171	.0724438
2.9	8.5	629	.015625	.3058104	.026418	.026418	.733945	.733945	1.258191	.0750971
2.8	11.5	839.5	.015625	.4081524	.035259	.035259	.9795656	.9795656	1.679255	.0834961
2.7	11.5	828	.015625	.4025612	.034776	.034776	.9661469	.9661469	1.656252	.0830372
2.6	10.5	745.5	.015625	.362451	.031311	.031311	.8698823	.8698823	1.491227	.0797455
2.5	9.5	665	.015625	.3233131	.02793	.02793	.7759513	.7759513	1.330202	.0765335
2.4	8	552	.0078125	.2683741	.023184	.023184	.644098	.644098	1.104168	.0720248
2.3	6.8	462.4	.0078125	.224812	.0194208	.0194208	.5395487	.5395487	.9249407	.0684498
2.2	6.6	442.2	.0078125	.2149910	.0185724	.0185724	.5159785	.5159785	.8845345	.0676438
2.1	6.3	415.8	.0078125	.2021557	.0174636	.0174636	.4851738	.4851738	.8317265	.0665904
2	6.7	435.5	.0078125	.2117336	.018291	.018291	.5081606	.5081606	.8711325	.0673765
1.9	10.5	672	.0078125	.3267164	.028224	.028224	.7841193	.7841193	1.344204	.0768128
1.8	12	756	.0078125	.3675559	.031752	.031752	.8821342	.8821342	1.51223	.0801644
1.7	10.5	651	.0078125	.3165065	.027342	.027342	.7596155	.7596155	1.302198	.0759749
1.6	8.5	518.5	.0078125	.2520869	.021777	.021777	.6050087	.6050087	1.037158	.0706882
1.5	6	360	.0078125	.1750266	.01512	.01512	.4200639	.4200639	.7201095	.064364
1.4	7.8	460.2	.0078125	.2237424	.0193284	.0193284	.5369817	.5369817	.92054	.068362
1.3	5.9	342.2	.0078125	.1663725	.0143724	.0143724	.3992941	.3992941	.6845041	.0636538
1.2	6.5	370.5	.0039063	.1801316	.015561	.015561	.4323157	.4323157	.7411127	.064783
1.1	7.2	403.2	.0039063	.1960298	.0169344	.0169344	.4704716	.4704716	.8065227	.0660877
1	6.3	346.5	.0039063	.1684631	.014553	.014553	.4043115	.4043115	.6931054	.0638254
.9	5.2	280.8	.0039063	.1365208	.0117936	.0117936	.3276498	.3276498	.5616854	.0612039
.8	5	265	.0039063	.1288390	.01113	.01113	.3092137	.3092137	.5300806	.0605735
.7	5.4	280.8	.0039063	.1365200	.0117936	.0117936	.3276498	.3276498	.5616054	.0612039
.6	5.8	295.8	.0039063	.1438135	.0124236	.0124236	.3451525	.3451525	.59169	.0618024
.5	5	250	.0039063	.1215463	.0105	.0105	.2917110	.2917110	.5000761	.059975
.4	7.2	352.8	.0039063	.1715261	.0148176	.0148176	.4116626	.4116626	.7057073	.0640767
.3	7.5	360	.0039063	.1750266	.01512	.01512	.4200639	.4200639	.7201095	.064364
.2	5.5	258.5	.0039063	.1256788	.010857	.010857	.3016292	.3016292	.5170786	.0603142
.1	7.4	340.4	.0039063	.1654974	.0142968	.0142968	.3971937	.3971937	.6809036	.063582
0	7	315	.0039063	.1531483	.01323	.01323	.3675559	.3675559	.6300958	.0625685
Total	311.5	20568.3	.3164063	10	6.700321	6.700321	24	24	41.14286	
CFAC		.3210273	.0010158	4.862E-4	4.2E-5	4.2E-5	.0011668	.0011668	.0020003	

1.) Most important is not to add or delete columns and rows unless you are able to make the appropriate changes in PLUMBO, because the values are read by their positioning in the spreadsheet. Also, do not erase functions within the cells. If you do decide to add or delete columns or rows and edit PLUMBO to accomodate these changes, no more than 100 rows or 50 columns are allowed.

2.) The column width must be 9 characters. This is the default column width that appears when you invoke VisiCalc, so be aware that you cannot change it and still have a data file compatible with PLUMBO.

3.) Column names or headings must appear above any data. The column names can occupy either one or two rows above the equals signs (see below). In other words, the column names can appear either as a single row of 9 characters per name, or as two stacked rows of 9 characters each. You shouldn't have to change any of the column names on TEMPLATE or VERSIONIV.

4.) The row directly underneath the column names must be filled with repeating equals-signs (==========). PLUMBO uses this row to tell it where to find the column names. There should be no breaks in this line in any column for the length of the of the longest row.

5.) The first two columns (A and B) and J, L, and N are used only for row names, so do not put any numeric data in these columns.

6.) PLUMBO will use the first row of the file as a file-title when it brings the file into memory. You can include a date, descriptive information, or whatever, but only 80 characters of the first row.

SAVING FILES

When you have added all the data or made all the changes that you want, you must store the file in two different ways. (If you use a different disk, you should make sure that VisiCalc knows the new Volume Name, by giving the /SV command.)

First, use the /SS command for saving your file. The resulting file can then be accessed by the VisiCalc program itself, so you can add or make changes later.

Second, use the /PF command. To write an ASCII file (text file), which can be read directly by PLUMBO. Only use letters or numbers in the /PF file name, or PLUMBO will not be able to access it. To store file using /PF...

1.) Put the cursor at the upper-left hand corner (usually cell A1). Before storing, VisiCalc will ask you where the lower-right hand corner of the file is to be (e.g. for TEMPLATE the lower-right hand corner is AA66). Only cells included in a rectangle defined by these two cells will be stored on the disk.

2.) Be sure that the /PF file is stored with a PRINTER WIDTH of 80. This is the default value when VisiCalc is first invoked, but it

may be reset to 132 for printing out small type on the printer. To reset the PRINTER WIDTH to 80, type /PN, then enter 7 (the printer select code), then 1 (the HPIB address), then 80 (the PRINTER WIDTH).

3.) Type /PF to store the ASCII file for **PLUMBO** to read and enter the lower-right hand corner cell number.

PRINTOUT

Once you have edited and saved your files, you are ready to load and run **PLUMBO** to make the model calculations. However, you may wish to have a printout of your file while you are still in **VisiCalc**.

To print all the parameters on two sheets of paper, you will first need to set the PRINTER WIDTH to 132 by the /PN command. Position the cursor in the upper-left hand cell of the rectangle you want printed. Next, type /PC. In response to Print: Lower right cell or "Setup, type ". When Enter Setup String is displayed, type ^E&k2S (2=small print, 0=regular print). Finally, enter the lower-right hand cell of the rectangle you want printed. (NOTE: This Setup String is unique to Hewlett-Packard printers and cannot be used for another brand.)

PLUMBO

This section describes how to load BASIC and run **PLUMBO**. **PLUMBO** retrieves the datafile of starting parameters you stored in **VisiCalc**. It then calculates the mass and isotopic budget among major terrestial reservoirs. Finally, it stores ASCII datafiles of run-time and ending-values for **ISOPLOT200** to graph and printout.

LOADING BASIC

To load **PLUMBO**, the computer needs to have BASIC and the appropriate language extensions loaded into its memory. Place the BASIC SYSTEM disk in one of the disk drives. If the computer is OFF, turn it on by pressing the black button at the back of the CRT, ON. If you are in **VisiCalc**, hold the SHIFT down and press the PAUSE (SHFT+RESET). This will cause the computer to test its memory and re-BOOT, thus loading whichever system disk is in the drive. SHFT+RESET has the same effect as turning OFF the computer and then turning it back ON. After BASIC is loaded (about a minute), replace the BASIC SYSTEM disk with BASIC LANGUAGE EXTENSIONS & DRIVERS disk and load the appropriate language extensions for the BASIC language version you are using. If you are using BASIC 2.1, load AP2_1 and GRAPH2_1 language extensions by typing LOAD BIN "<filename>". If you are using BASIC 3.x version of the language, load GRAPH, GRAPHX, MAT, CLOCK, KBD, ERR, KNB2_1, CS80, HPIB, and MS, by typing LOAD BIN "<filename>".

LOADING PLUMBO

First, load **PLUMBO** by putting the **PLUMBO** disk in the left-hand drive and typing LOAD "PLUMBO:700,0",1. The ,1 just after the quotation marks indicates that the program is to be run as soon as it is loaded into memory. The program will take about a minute to load and run (NOTE: The drive address (700,0) is valid only if the DIP switches on the drive are set to 000.)

RETRIEVING VisiCalc FILES (the CATALOG screen)

The first CRT-display, the CATALOG screen, will look like the following:

```
<<<<<<<<<<<<< PLUMBOTECTONICS MODELLING PROGRAM >>>>>>>>>>>>>

Robert Zartman and Sara Haines                          (Rev. 02/04/88)

      Press  k0  for a catalog of the disk in the left-hand drive

      Press  k1  for a catalog of the disk in the right-hand drive

      Press  k4  to load a VisiCalc datafile into memory

 ---------------------------------------------------------------------------
|CATALOG Drv #0|CATALOG Drv #1|              |              |LOAD DATAFILE|
|--------------+--------------+--------------+--------------+-------------
|              |              |              |              |             |
 ---------------------------------------------------------------------------
```

Place the disk, with the file you saved while in VisiCalc using the /PF command, in either disk drive. Press softkey k4 to load your datafile. The program will prompt you for which datafile you wish to access and the drive it is in. If you cannot remember the datafile name, CATALOG either drive by pressing softkey k0 or k1. If the file was stored correctly in VisiCalc, the CRT-display screen will look something like the following:

```
    DATAFILES ON DRIVE #1                          LABEL:   VFILES

          FILES#      FILE NAME

            1    ---  VERSIONIV
            2    ---  TEMPLATE


    ENTER THE FILE-NUMBER THAT YOU WANT TO LOAD

 ---------------------------------------------------------------------------
|              |              |              |              |             |
|--------------+--------------+--------------+--------------+-------------|
|              |              |              |              |   ESCAPE    |
 ---------------------------------------------------------------------------
```

Enter the number of the file you choose to run, as shown on the screen. Once you have chosen a datafile, the program will search for and retrieve the data. Press k8 to return to the first screen, if you want to CATALOG the other drive or another disk.

After the data has been retrieved, the following CRT-display will appear:

KEY DESCRIPTION for **<filename>**, **<number of>** times upper mantle recycled

k0 Print starting parameters on printer.
k1 Print starting parameters on CRT.
k2 Change A2 or A4.
k3 Toggle with reseting B1.
k4 Toggle with reseting B2.
k5 Change (A3a) Vertical Erosion.
k6 Help screen that descrbes constants.

k7 Which isotope systematics do you want to calculate?
U-Th-Pb only, Rb-Sr and Sm-Nd only, or all three systematics.
Currently **U-Th-Pb only** will be calculated.

k8 Get a different set of parameters.
k9 Start calculating, RUN the program.

Printer	CRT	A4=.14	B1=1.0	B2=0.01
A3a= .05	Const DESCR.	U-Th-Pb only	Get Data	RUN

This screen allows you to change some of the constants used in the calculations without editing the program. Press Const DESC. (k6) for a description of these constants. The default values are the values that were used for the calculations of version IV (Zartman and Haines, in press).

It is a good idea to print the starting parameters, either on the Printer (k0) or on the CRT (k1), to be sure that you retrieved the file you chose and that the file was read correctly by PLUMBO. Table 3 is a printout by PLUMBO of the starting parameters of version IV. If you chose the wrong file, press softkey k7 (Get Data) to return to the CATALOG screen where you can load a different datafile. If all that is printed looks like garbage (either zeroes or incorrect values), a number of things could have happened while editing or storing (/PF) your particular datafile while using VisiCalc. Here are some common problems.

1.) Deleting or adding a column or row.
2.) Not positioning the cursor in cell A1 when using the /PF command.
3.) Not giving the correct lower-right hand cell designation and hence "chopping off" data.
4.) Storing the file when using the /PF command with a PRINTER WIDTH of 132 (or any width other than 80).

Table 3. Printout of version IV starting parameters using **PLUMBO.**

<<<<<<<<<<< PLUMBOTECTONICS MODELLING PROGRAM >>>>>>>>>>>

TABLE OF STARTING PARAMETERIC VALUES

Constants

The start time is 4.5 B.Y.A., with a time decrement of .1
and 46 cycles.

A3(Vertical erosion)= .05
Bs(Baseline, in km)= .001
A2(Fraction of MORidge to create Dorogene)= .2
Dp(Fraction of vertically-eroded Ucrust to go
to Dorogene)= .14
B1(Fraction of Porogene fill from Dorogene)= 1
B2(Fraction of Worogene filled from Dorogene)= .01
B3(Fraction of Porogene fill from Worogene) = 0

Decay Constants		Start Iso Comps		Start El Abund	
238 U	.155125	206/204	9.21	Mass	1050
235 U	.98485	207/204	9.95	204 Pb	19
232 Th	.049475	208/204	28.85	232 Th	758
87 Rb	.0142	87/86	.69896	238 U	177
147 Sm	.00654	143/144	.50675	86 Sr	22000
				87 Rb	2140
				144 Nd	2100
				147 Sm	426

Isotope Name	E_a2(H)	F_a3(H)	E_b1(H)	E_b2(H)	Eb3(H)	F_c3(H)
204 Pb	80.000	1.000	40.000	25.000	25.000	3.670
206 Pb	80.000	1.000	40.000	25.000	25.000	3.670
207 Pb	80.000	1.000	40.000	25.000	25.000	3.670
208 Pb	80.000	1.000	40.000	25.000	25.000	3.670
232 Th	100.000	.670	47.000	25.000	25.000	4.000
238 U	100.000	1.500	52.000	27.000	27.000	6.250
86 Sr	12.000	1.000	8.000	5.000	5.000	1.000
87 Sr	12.000	1.000	8.000	5.000	5.000	1.000
87 Rb	50.000	1.000	23.000	10.000	10.000	9.000
144 Nd	10.000	1.000	6.000	5.000	5.000	2.330
143 Nd	10.000	1.000	6.000	5.000	5.000	2.330
147 Sm	8.000	1.000	5.000	3.000	3.000	2.250

Table 3. Continued.

Time Dependent Values

Cycle	A1(K)	A5(K)	A6(K)	U(K)	L(K)	S(K)
1	0.000000	1.000000	1.000000	0.000000	0.000000	0.000000
2	0.000000	1.000000	1.000000	0.000000	0.000000	0.000000
3	0.000000	1.000000	1.000000	0.000000	0.000000	0.000000
4	.071909	1.000000	1.000000	.172570	.172570	.295844
5	.133802	1.000000	1.000000	.321103	.321103	.550483
6	.185972	.500000	.500000	.446301	.446301	.765115
7	.228708	.250000	.250000	.548863	.548863	.940941
8	.262305	.125000	.125000	.629489	.629489	1.079162
9	.287052	.062500	.062500	.688879	.688879	1.180977
10	.303243	.026195	.026195	.727733	.727733	1.247587
11	.311168	.026880	.026880	.746752	.746752	1.280192
12	.311119	.026876	.026876	.746635	.746635	1.279992
13	.303389	.026208	.026208	.728083	.728083	1.248187
14	.292012	.025225	.025225	.700780	.700780	1.201380
15	.280829	.024259	.024259	.673944	.673944	1.155373
16	.273487	.023625	.023625	.656325	.656325	1.125169
17	.305820	.026418	.026418	.733917	.733917	1.258189
18	.408165	.035259	.035259	.979529	.979529	1.679252
19	.402574	.034776	.034776	.966110	.966110	1.656248
20	.362462	.031311	.031311	.869849	.869849	1.491224
21	.323323	.027930	.027930	.775922	.775922	1.330199
22	.268382	.023184	.023184	.644074	.644074	1.104166
23	.224819	.019421	.019421	.539528	.539528	.924939
24	.214998	.018572	.018572	.515959	.515959	.884533
25	.202162	.017464	.017464	.485155	.485155	.831725
26	.211740	.018291	.018291	.508141	.508141	.871131
27	.326726	.028224	.028224	.784090	.784090	1.344202
28	.367567	.031752	.031752	.882101	.882101	1.512227
29	.316516	.027342	.027342	.759587	.759587	1.302195
30	.252095	.021777	.021777	.604986	.604986	1.037156
31	.175032	.015120	.015120	.420048	.420048	.720108
32	.223749	.019328	.019328	.536961	.536961	.920538
33	.166378	.014372	.014372	.399279	.399279	.684503
34	.180137	.015561	.015561	.432299	.432299	.741111
35	.196036	.016934	.016934	.470454	.470454	.806521
36	.168468	.014553	.014553	.404296	.404296	.693104
37	.136525	.011794	.011794	.327637	.327637	.561684
38	.128843	.011130	.011130	.309202	.309202	.530080
39	.136525	.011794	.011794	.327637	.327637	.561684
40	.143818	.012424	.012424	.345139	.345139	.591689
41	.121550	.010500	.010500	.291700	.291700	.500075
42	.171531	.014818	.014818	.411647	.411647	.705706
43	.175032	.015120	.015120	.420048	.420048	.720108
44	.125683	.010857	.010857	.301618	.301618	.517078
45	.165502	.014297	.014297	.397179	.397179	.680902
46	.153153	.013230	.013230	.367542	.367542	.630095

Return to VisiCalc and you file and re-store or edit you file paying particular attention to these problems.

RUN-TIME

It takes about 6 minutes (approximately 1.07 X 10^6 calculations) for PLUMBO to make calculations for all three isotope systematics (U-Th-Pb, Rb-Sr, and Sm-Nd) when the program finally reaches the Output Options screen. Since the primary objective of version IV was the interpretation of Pb isotopic systematics, it was desirable to only calculate the U-Th-Pb budget among the major terrestial reservoirs, thereby cutting the calculation-time in half. Press softkey k7 to choose the isotope systematics you would like to have calculated.

When you have determined that all the starting parameters are to your liking, press softkey k9 (RUN) for PLUMBO to calculate. The following will appear on the CRT, so that you can monitor PLUMBO's status:

RUNNING

PLEASE DO NOT DISTURB

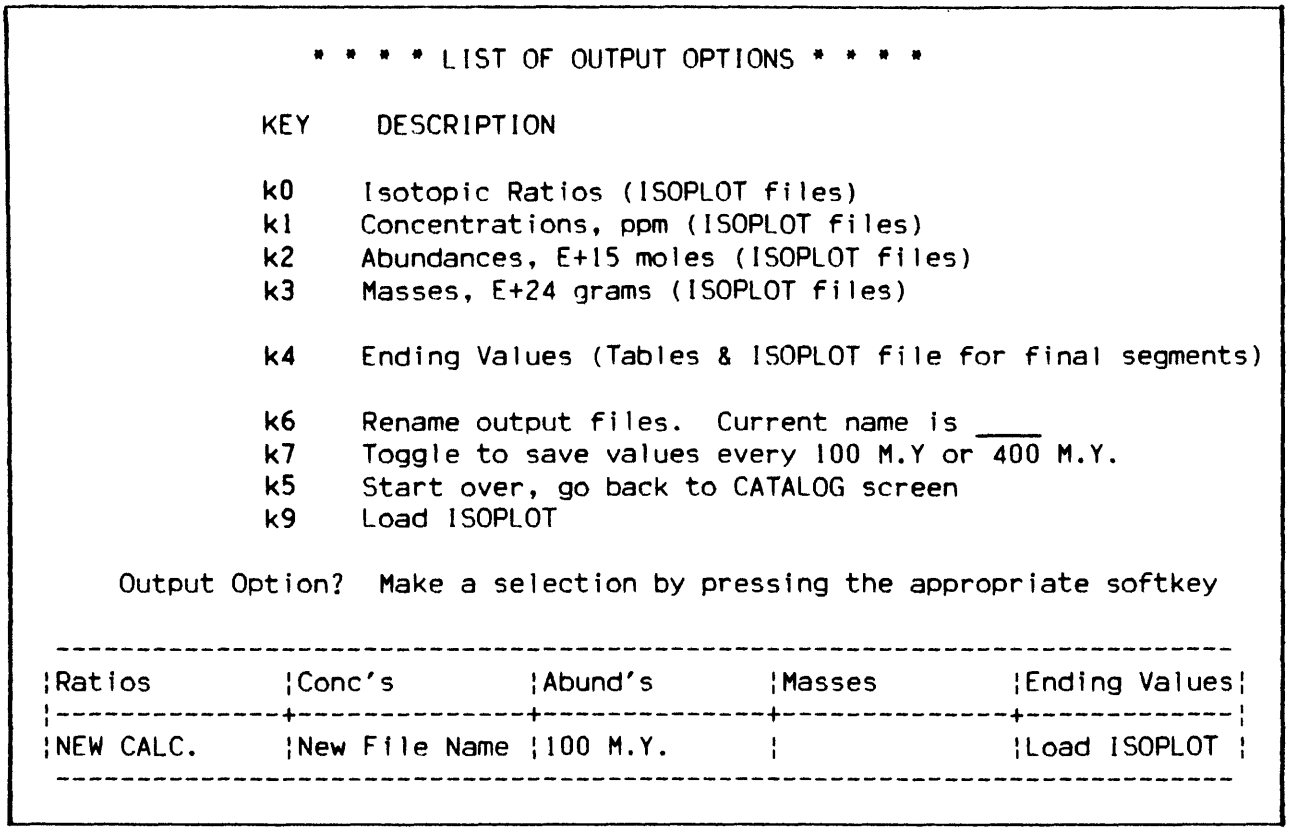
Calculating mass and isotopic abundances
for 3 cycle of 43 cycles.

Press k9 to abort RUN and return to the main screen (CATALOG)

				ESCAPE

OUTPUT OPTIONS

After PLUMBO is done calculating, the program will request a file name that is exactly 4 characters long which will represent this run (e.g. "STD1" or "A2__"). This file name will prefix any files saved for ISOPLOT200. The default file name is "____". At this time, place an empty disk in the drive from which you initially retrieved your datafile. PLUMBO will store all files requested to the disk in this drive. To save all the files, it will take 2 empty disks for your particular run. After choosing a file name, the Output Options screen will appear on the CRT:



Press softkeys k1 through k3 to save the indicated files. After PLUMBO has saved the type of file you selected, the program will return to this CRT-display. The program attempts to keep you informed of its status if it requires time to perform the operation it is on (either more calculations or storing files). For example, if you choose Conc's (k1), the program will first inform you:

Please be patient, I'm thinking.

Then the program will begin storing a file for each element (Pb, Th, U, Sr, Rb, Sm, and Nd). PLUMBO will indicate which file is being stored for each elemental concentration by printing on the CRT, for example:

Writing file for concentrations of **Pb**

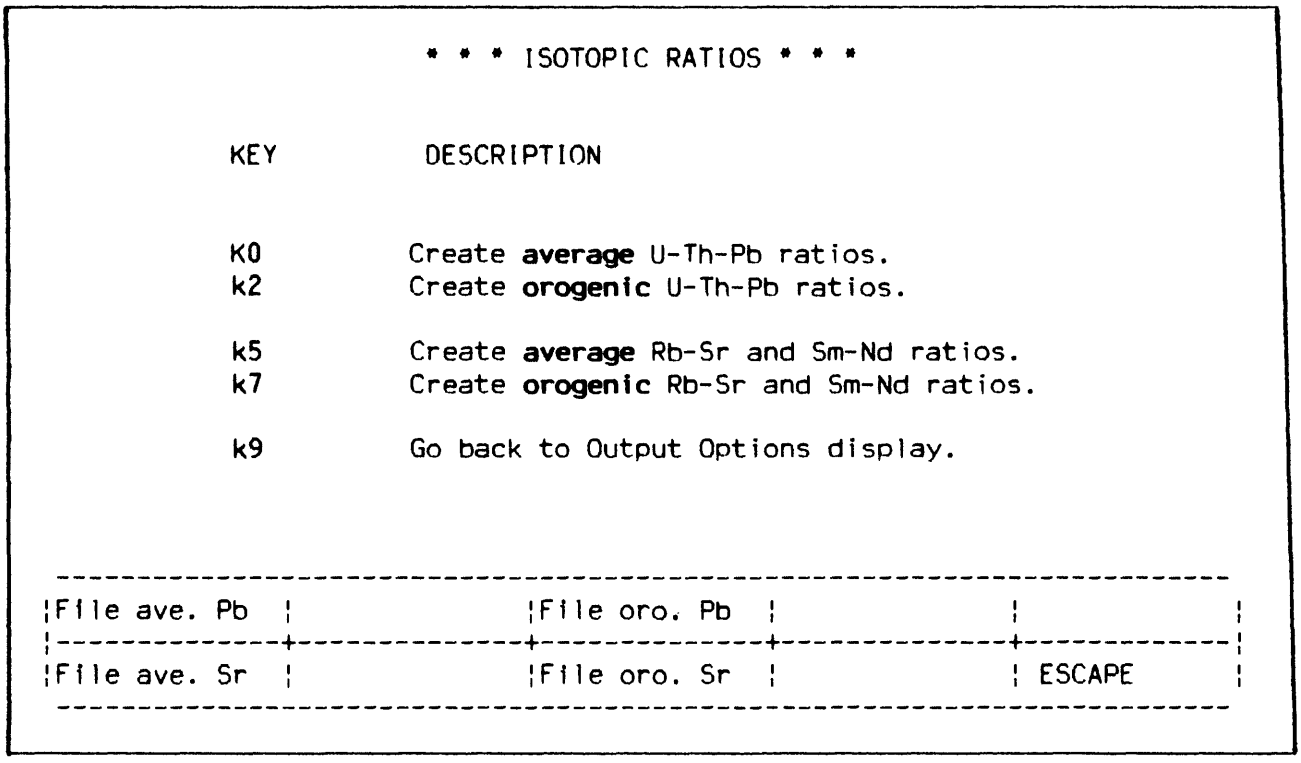
Likewise, if you chose Abund's (k2) or Masses (k3), the CRT will keep you informed of the program's status, whether it is writing files or doing more calculations.

Your storage disk may become full before you have saved all of the files you want. The computer will beep and print at the bottom of the screen:

ERROR 64 Mass storage media overflow.

Simply replace the full disk with an empty (blank) one and press CONTINUE.

Press softkey k0 (Ratios) of the Output Options screen to display the screen for saving files of isotopic ratios:

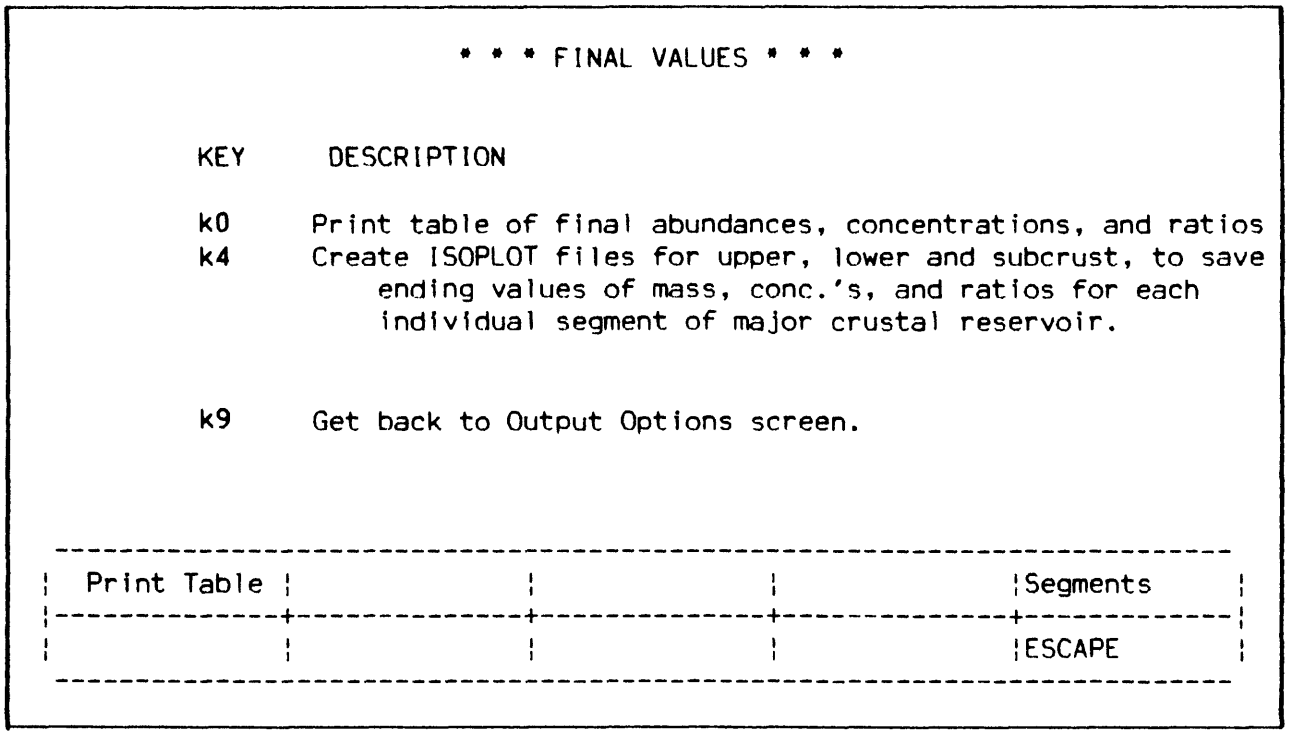


Press File ave. Pb (k0) to save average crustal U-Th-Pb ratios. Since the upper, lower, and subcrust of the model contain more than one segment after the first orogeny, the value that is stored represents an average ratio for all segments of that reservoir. Likewise for File ave. Sr, press softkey k5 to save files for average Rb-Sr and Sm-Nd isotopic ratios.

Orogenic reservoirs are considered to be instantaneous during an orogeny, for example the distal, proximal, and mantle wedge components. These reservoirs do not survive from one orogeny to the next in the calculations and do not go through the decay calculations. Since the orogene of version IV acts not only as a mixer but a discriminator, you may want to see values of the orogene (contribution to, functions of, and dispersion of the orogene). If so, press File oro. Pb (k2) or File oro. Sr (k7) to save files for ISOPLOT200 of model orogenic values.

Press softkey k9 to ESCAPE back to the Output Options screen.

Press k4 (Ending Values) of the Output Options screen to display the following screen:



Press Print Table (k0) to print out a table, either to the printer or the CRT, of the final mass, concentration, and isotopic ratio configuration of average upper crust, lower crust, subcrust, and mantle. This output option allows you to view the final distribution of mass and isotopic abundance without having to store all the files and get into ISOPLOT200 and search and extract this pertinent information from all the files. Table 4 is a table printed by PLUMBO for version IV by enabling the Print Table function.

Press Segments (k4) to save a file for ISOPLOT200 of the final mass, concentration, and isotopic ratios of each individual segment of upper crust, lower crust, and subcrust as it would be seen today. This output option allows you to see how the average of each major crust form is weighted.

Press softkey k9 to return to the Output Options screen.

Once you have stored the files you desire, press Load ISOPLOT (k9) of the Output Options screen if you wish to graph or printout the files you have stored.

Table 4. Table of ending values of version IV as printed by PLUMBO.

* * * ENDING MASS AND RESERVOIR ABUNDANCES * * *
(AVERAGES, in E+15 moles)

A2(Fraction of MORidge to create Orogene)= .2
Dp(Fraction of vertically-eroded Ucrust to go to Orogene)= .14
B1(Fraction of Porogene fill from Orogene)= 1
B2(Fraction of Worogene filled from Orogene)= .01
B3(Fraction of Porogene fill from Worogene) = 0

Bs(Baselevel of vertical erosion)= .001 Km
A3(Vertical erosion)= .05
Final Continental Area = 151.5 E+06 (Km^2)
FOR VERSION IV MODEL 10 times upper mantle

ISOTOPES	MANTLE	TOTAL ORO	UPPER	LOWER	SUB	TOTAL
MASS	1000.100	--	6.703	15.910	27.280	1049.990
204 Pb	2.000	--	9.950	6.660	.386	19.000
206 Pb	37.000	--	192.000	117.000	7.070	353.734
207 Pb	31.000	--	156.000	102.000	5.960	295.708
208 Pb	75.600	--	389.000	258.000	14.700	737.171
232 Th	54.700	--	428.000	262.000	13.700	758.000
238 U	20.100	--	110.000	43.300	3.530	177.000
86 Sr	11700.000	--	2850.000	6960.000	461.000	22000.000
87 Sr	8250.000	--	2040.000	4910.000	325.000	15518.300
87 Rb	469.000	--	1300.000	337.000	35.500	2140.000
144 Nd	1340.000	--	356.000	356.000	48.000	2100.000
143 Nd	687.000	--	182.000	182.000	24.600	1076.900
147 Sm	297.000	--	58.600	60.600	9.290	426.000

* * * ENDING RESERVOIR CONCENTRATIONS * * *
(AVERAGES, in ppm)

ELEMENT	MANTLE	TOTAL ORO	UPPER	LOWER	SUB
Pb	.0302	--	23.1000	6.3020	.2136
Th	.0127	--	14.8200	3.8210	.1165
U	.0048	--	3.9350	.6525	.0310
Sr	10.4000	--	377.8000	388.7000	15.0200
Rb	.1439	--	59.5200	6.5010	.3994
Nd	.8096	--	32.0900	13.5200	1.0630
Sm	.2963	--	8.7220	3.8000	.3398

Table 4. Continued.

* * * ENDING OROGENIC ISOTOPIC RATIOS * * *

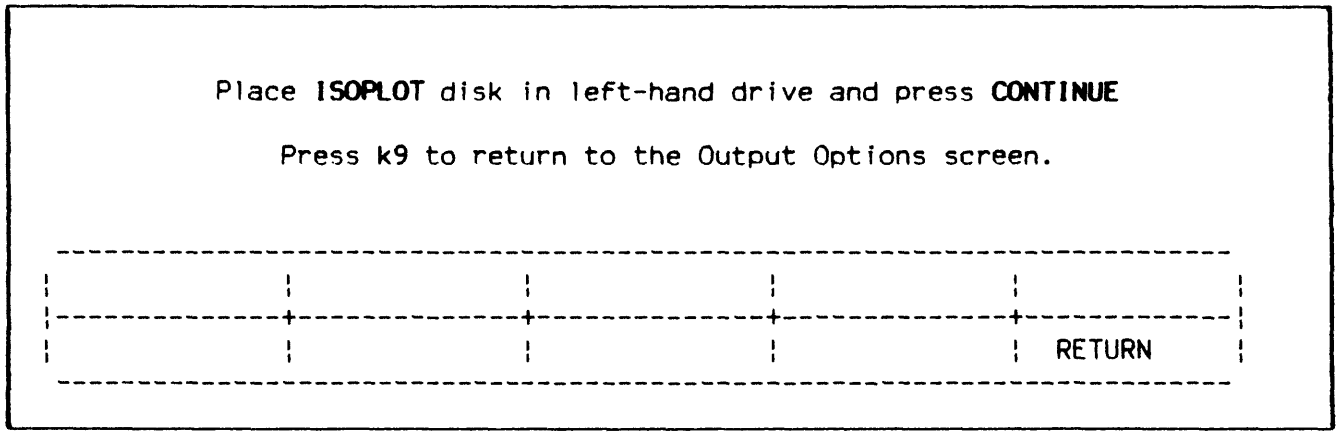
	MORB	EROSION		BEFORE TRUCKING		AFTER TRUCKING			
RATIOS		Vert	Horiz	Porogene	Orogene	Porogene	Orogene	Worogene	Total_or
206Pb/204Pb	18.44700	19.34100	18.62300	19.10300	18.62100	19.06200	18.62100	18.40100	18.92700
207Pb/204Pb	15.47400	15.73100	15.57200	15.67800	15.52400	15.66500	15.52400	15.46300	15.62200
208Pb/204Pb	37.68900	39.07900	38.94400	39.03400	37.95900	38.94300	37.95900	38.04000	38.64500
238U/204Pb	10.02100	11.06600	9.19070	9.96060	11.08700	10.27900	10.55900	9.87740	10.35900
232Th/238U	2.75410	3.88410	4.51070	4.40410	2.53460	4.17120	2.58660	3.36050	3.68620
232Th/204Pb	27.59800	42.97900	41.45600	43.86800	28.10200	42.87400	27.31200	33.19300	38.18400
87Sr/86Sr	.70290	.71518	.70831	.71169	.70308	.71021	.70308	.70371	.70441
87Rb/86Sr	.04910	.45290	.16320	.30590	.05503	.27870	.05131	.07091	.09370
143Nd/144Nd	.51314	.51218	.51218	.51218	.51313	.51232	.51313	.51290	.51299
147Sm/144Nd	.20720	.16470	.16750	.16570	.20650	.16680	.20750	.19280	.20070

* * * * ENDING ISOTOPIC RATIOS FOR THE MAJOR RESERVOIRS * * * *

RATIOS	Mantle	Upper	Lower	Subcrust
206Pb/204Pb	18.47300	19.32500	17.62200	18.31800
207Pb/204Pb	15.48200	15.72700	15.35100	15.44000
208Pb/204Pb	37.72900	39.07100	38.75300	38.10100
238U/204Pb	10.00600	11.07700	6.49030	9.12920
232Th/238U	2.72760	3.88330	6.05180	3.89320
232Th/204Pb	27.29400	43.01600	39.27800	35.54200
87Sr/86Sr	.70292	.71491	.70569	.70422
87Rb/86Sr	.03999	.45550	.04846	.07700
143Nd/144Nd	.51314	.51219	.51219	.51272
147Sm/144Nd	.22200	.16470	.17010	.19340

ISOPLOT200

While still in the Output Options screen of PLUMBO press k9 (Load ISOPLOT). The CRT will display:

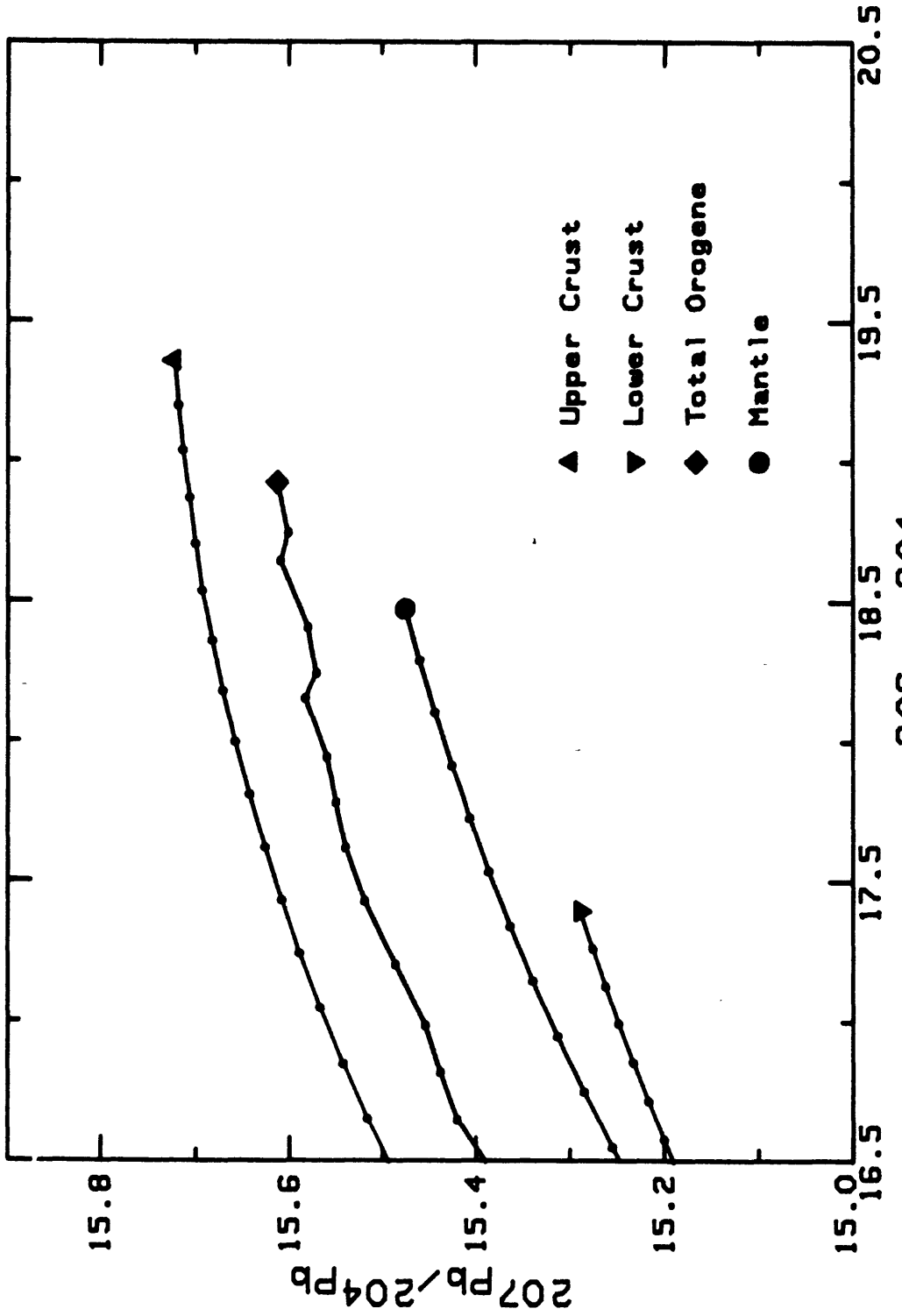


Once ISOPLOT is running, catalog the disks which contain the PLUMBO datafiles. These are the file names you should read, with the four character prefix you chose in PLUMBO. For example, version IV datafile names appear as follows:

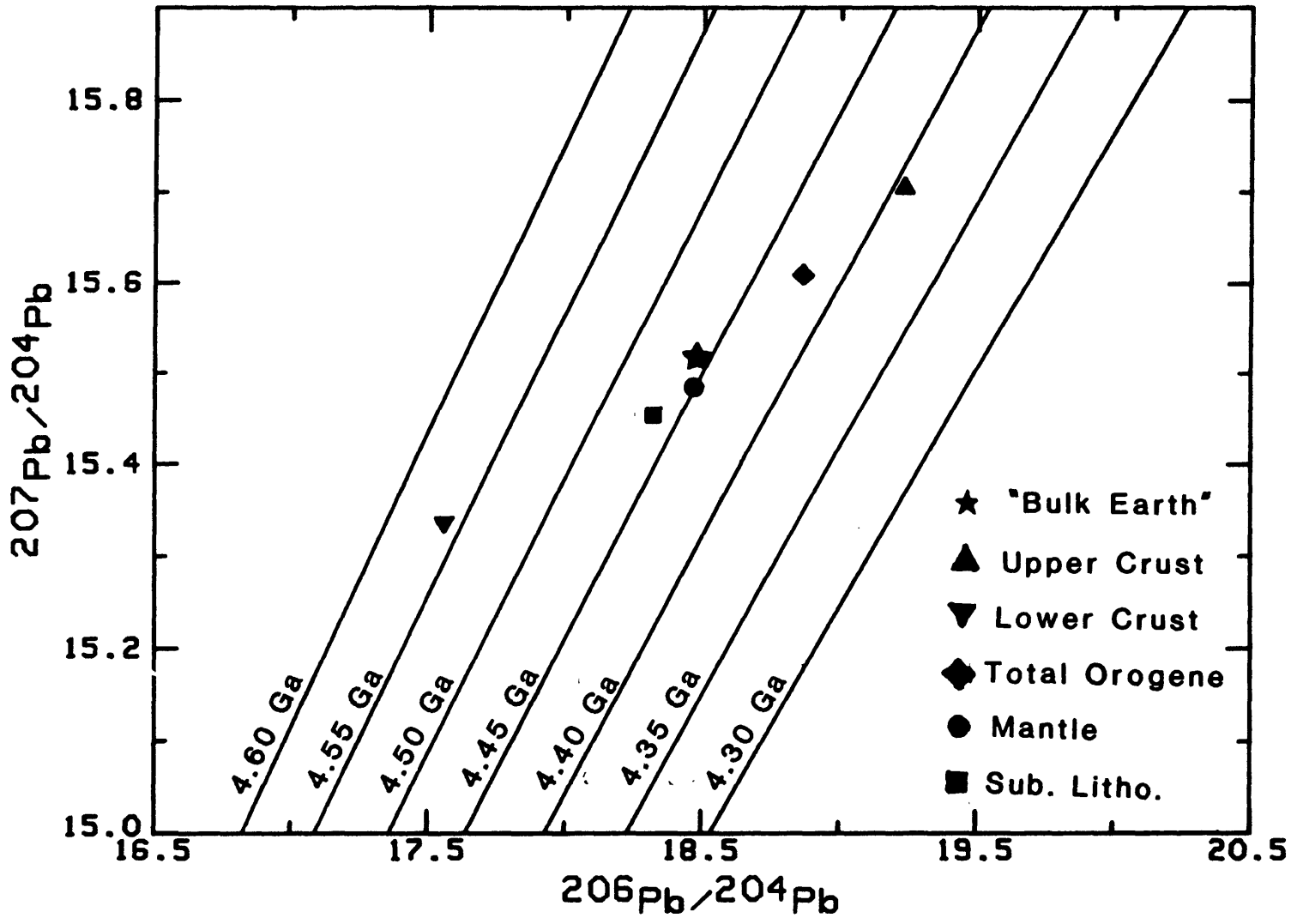
IV__Mass_	IV__204Pb
IV__AvePb	IV__206Pb
IV__AveSr	IV__207Pb
IV__OroPb	IV__208Pb
IV__OroSr	IV__232Th
IV__conPb	IV___238U
IV__conTh	IV___86Sr
IV__con_U	IV___87Sr
IV__conSr	IV___87Rb
IV__conRb	IV__144Nd
IV__conNd	IV__143Nd
IV__conSm	IV__147Sm
IV__upseg	
IV__loseg	
IV__suseg	

Since PLUMBO generates so many values in one run, it is highly desirable to view some of these values graphically and have a way to select information. ISOPLOT200 (Ludwig, 1985) offers a simple, rapid, and flexible way of generating X-Y plots and retrieving data from the files saved while in PLUMBO. Because ISOPLOT200 is so easy to use and is provided with documentation of its own, it is pointless to go through the steps for generating plots and data retrieval. Graphs 1 through 3 are plots made using ISOPLOT200 just as a few examples of the wide variety of symbols and plot-types that can be made.

Graph 1.



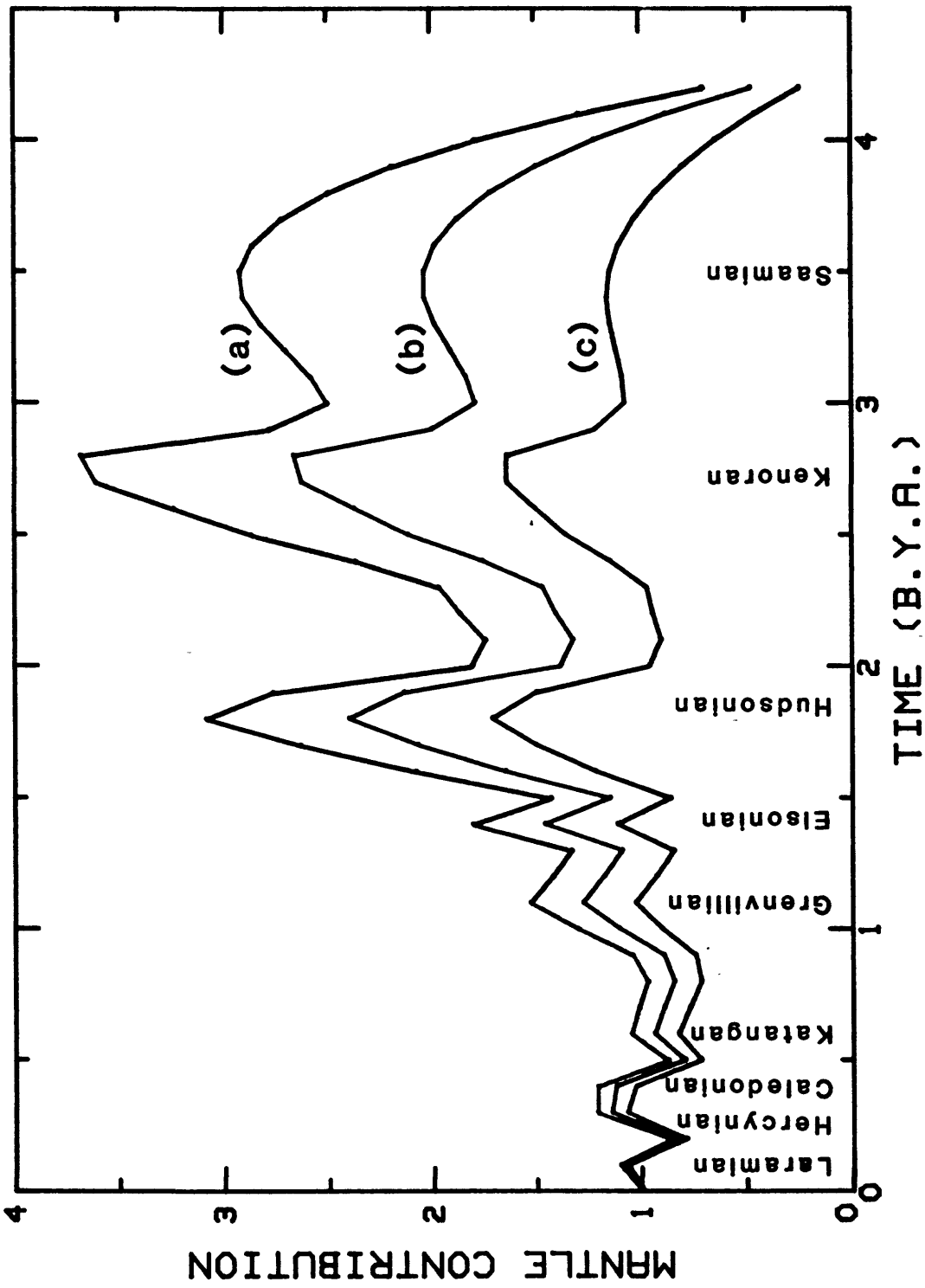
Graph 1. Pb isotope growth curves for major reservoirs as generated by PLUMBO and plotted using ISOPLOT200 (values are for version IV).



Graph 2.

Graph 2. Ending Pb isotope ratios for version IV of the major reservoirs. Isochrons were generated using ISOPLOT200 and initial ratios of Canyon Diablo Pb.

Graph 3.



Graph 3. Mantle contribution to the orogene as a function of time.

PROGRAM EXPLANATION

For those who wish to repeat our calculations or make modifications to the program, this section describes the design of the program and the sequence of calculations within the program. Included in this section is a conversion table from the generalized algebraic notation to HP BASIC (Table 5), a program outline (Table 6), and an outline of the transport equations in the program (Table 7). Appendix A lists the entire PLUMBO program.

Table 5(a). Conversion table from algebraic notation to HP BASIC.

ALGEBRAIC	HP BASIC	DESCRIPTION
$M(t)^{\mu}$	M(0,I,K,J)	Mass of reservoir μ
	I=1	Immediately prior to an orogeny.
	I=2	Immediately after an orogeny.
	J=0	Mantle
	J=1	Upper Crust
	J=2	Lower Crust
	J=3	Subcrustal Lithosphere
	K=1 to 46	Orogenic cycle, from 4.5 Ga until present in 0.1 b.y. intervals.
$^{\alpha}N(t)^{\mu}$	M(H,I,K,J)	Number of moles in reservoir μ.
	H=1 to 12	Isotopic species α
	I	(same as above)
	J	(same as above)
	K	(same as above)
$\Delta M(t)^{\mu\to\nu}$	Table 5(b)	Increment of mass being transported from reservoir μ to reservoir ν
$\Delta^{\alpha}N(t)^{\mu\to\nu}$	Table 5(b)	Number of moles of species ∇ being transported from reservoir μ to reservoir ν.
$\Phi(t)^{\mu\to\nu}$	Table 5(c)	Mass fraction of reservoir μ removed.
$\alpha_E{}^{\mu\to\nu}$	Table 5(d)	Partitioning coefficient for species α between removed increment and contributing reservoir μ.
$\alpha_F{}^{\mu\to\nu'(,\nu'')}$	Table 5(d)	Partitioning coefficient for species α between the two receiving reservoirs ν′ and ν″.

Table 5(b). Mass and isotopic transport functions (for $\Delta M(t)^{\Sigma\mu\rightarrow\nu'(,\nu'')}$, $\alpha_{N(t)}{}^{\Sigma\mu\rightarrow\nu'(,\nu'')}$).

		HP BASIC		
GATE	ALGEBRAIC	CONTRIBUTING RESERVOIR	VEHICLE	RECEIVING RESERVOIR
A1	m→r	M(H,1,K,0)	Mantle(H,K)	Mor(H,K)
A2	r→d	Mor(H,K)	→	D_oro(H,K)
A3	Σv→d,p	M(H,1,K,1)	Vert(H,K)	{ D_oro(H,K) ; P_oro(H,K)
A4	Σu→p	M(H,1,K,1)	U_hz(H,K)	P_oro(H,K)
A5	Σl→p	M(H,1,K,2)	L_hz(H,K)	P_oro(H,K)
A6	Σs→w	M(H,1,K,3)	Sub_re(H,K)	W_oro(H,K)
A7	m→w	M(H,1,K,0)	Sub_mant(H,K)	W_oro(H,K)
B1	d→p	D_oro(H,K,1)	Gateb1(H,K)	P_oro(H,K,0)
B2	d→w	D_oro(H,K,0)	Gateb2(H,K)	W_oro(H,K,1)
B3	w→p	W_oro(H,K,1)	Gateb3(H,K)	P_oro(H,K,1)
C1	r→m	Rmor(H,K)	→	M(H,2,K,0)
C2	d→m	D_oro(H,K,2)	→	M(H,2,K,0)
C3	p→u,l	P_oro(H,K,1)	→	{ M(H,2,K,1) ; M(H,2,K,2)
C4	w→s	W_oro(H,K,2)	→	M(H,2,K,3)

H=0 to 12 (Mass and each isotopic species α), K=1 to 46 (Orogenic cycle).

m - mantle, r - mid-ocean ridge, d - distal component of orogene, p - proximal component of orogene, w - mantle wedge component of orogene, u - upper crust, l - lower crust, s - subcrustal lithosphere, v - vertically recycled component of upper crust.

Table 5(c). Mass fraction of reservoir μ removed ($\Phi(t)^{\mu\rightarrow\nu'(,\nu'')}$).

GATE	ALGEBRAIC	HP BASIC	VERSION IV
A1	m→r	A1(K)	Time dependent
A2	r→d	A2	0.20
A3	Σv→d,p	A3	0.05
A4	Σu→p	A4(K)	Time dependent
A5	Σl→p	A5(K)	Time dependent
A6	Σs→w	A6(K)	Time dependent
A7	m→w	*	Time dependent
B1	d→p	*	Time dependent
B2	d→w	*	Time dependent
B3	w→p	*	Time dependent
C1	r→m	----	1.00
C2	d→m	----	1.00
C3	p→u,l	**	0.50(u),0.50(l)
C4	w→s	----	1.00

K=1 to 46 (Orogenic cycle)

m - mantle, r - mid-ocean ridge, d - distal component of orogene, p - proximal component of orogene, w - mantle wedge component of orogene, u - upper crust, l - lower crust, s - subcrust, v - vertically recycled component of upper crust, * - controlled by receiving reservoir, ** - partitioned between upper and lower crust in proportion to U(K) and L(K).

Table 5(d). Partitioning coefficients ($^{\alpha}E^{u\rightarrow v}$, and $^{\alpha}F^{u\rightarrow v'(,v'')}$).

GATE	ALGEBRAIC	HP BASIC
A2	r→d	E_a2(H)
A3	v→d,p	F_a3(H)
B1	d→p	E_b1(H)
B2	d→w	E_b2(H)
B3	w→p	E_b3(H)
C3	p→u,l	F_c3(H)

H= 1 to 12 (Each isotopic species α).

r - mid-ocean ridge, d - distal component of orogene, p - proximal component of orogene, w - mantle wedge component of orogene, u - upper crust, l - lower crust, v - vertically eroded component of upper crust.

Table 6. General outline of PLUMBO.

LINES		FUNCTIONS
100-560	Start:	Dimension arrays and initialize values and strings
570-1380	Convert_vc:	Retrieve VisiCalc file and convert to arrays used in calculations.
1390-3470	St_options:	CRT-display to edit constants, print starting parameters.
3480-5270	Main_program:	Calculate mass and isotope budget.
5280-5930	Options:	CRT-display of output options.
5940-6730	Ave_pb:	Save file for average U-Th-Pb ratios.
6740-7460	Ave_sr:	Save file for average Rb-Sr and Sm-Nd ratios.
7470-8630	Oro_pb:	Save file for orogenic U-Th-Pb ratios.
8640-9750	Oro_sr:	Save file for orogenic Rb-Sr and Sm-Nd ratios.
9760-10450	Abundances:	Save files for isotopic abundances.
10460-11280	Concentrations:	Save files of elemental concentrations.
11290-11960	Masses:	Save file for masses.
11970-15500	Ending_values:	Printout of ending concentrations, abundances, masses, and isotopic ratios.
15510-16200	Segments:	Save files for crustal segments at 0.0 Ga.
16210-16690	Quit:	Load ISOPLOT200.
16700-22610	Subprograms:	Run ISOPLOT200 plotting and regression programs.

Table 7. Outline of Main_prog (sequence of transport equations in PLUMBO.

LINES	FUNCTIONS		
		Gate	Function
3530-4110	Mass:		
		A3	Σv→d,p
		A4	Σu→p
		A5	Σl→p
		A6	Σs→w
		A1	m→r
		A2	r→d
		B3	w→p
		B2	d→w
		B1	d→p
		A7	m→w
		C1	r→m
		C2	d→m
		C3	p→u,l
		C4	w→s
4120-4570	Isotopes:		
		A3	Σv→d,p
		A4	Σu→p
		A5	Σl→p
		A6	Σs→w
		A1	m→r
		A2	r→d
		B2	d→w
		B1	d→p
		A7	m→w
		B3	w→p
		C1	r→m
		C2	d→m
		C3	p→u,l
		C4	w→s
4580-5270	Increment: Redistribute values for next orogeny, decay equations, and increment counters.		

ACKNOWLEDGMENTS

Throughout the course of this work we had many occasions to consult with Kenneth R. Ludwig about aspects of computer hardware and software. His readiness to share his expertise in programming the Hewlett-Packard Series 200/300 computer, and the access permitted to subprograms of ISOPLOT200, such as Convert_vc, Krlinput, FNH$, and Clunk, etc., are greatly appreciated.

REFERENCES

Doe, B. R. and Zartman, R. E., 1979, Chapter 2, Plumbotectonics I, The Phanerozoic, in Geochemistry of Hydrothermal Ore Deposits, 2nd Ed., H. L. Barnes (ed.), New York, Wiley Interscience, p. 22-70.

Hewlett-Packard, 1983, HP98810A/98811A VisiCalc for Series 200 Computers, Hewlett-Packard, Fort Collins, CO.

Ludwig, K. R., 1985, ISOPLOT200 A plotting and regression program for isotope geochemists, for use with HP Series 200 Computers: U.S. Geological Survey Open-File Report 85-513.

Zartman, R. E., 1984, Lead, strontium, and neodymium isotopic characterization of mineral deposits to their geologic environments: Proc. 27th Internat'l Geol. Cong., Moscow 4-14 August, 1984, v. 12, Metallogenesis and Mineral Ore Deposits: VNU Science Press, Utrecht, p. 83-106.

Zartman, R. E. and Doe, B. R., 1981, Plumbotectonics--The model: Tectonophysics, v. 75, p. 135-162.

Zartman, R. E. and Haines, S. M., in press, The plumbotectonic model for Pb isotopic systematics among major terrestrial reservoirs--a case for bi-directional transport.

APPENDIX A

Program Listing of PLUMBO

```
10    !                        ***** PLUMBO *****
20 Rev_date$="02/10/88"
30    !    This program accesses an ASCII file created by VISICALC which stores
40    ! starting parameters for the Plumbotectonic Model (Zartman and Haines,
50    ! 1988). By employing the VISICALC template VERSIONIV and progam toggles,
60    ! the user can vary the starting parameters and, therefore, some of the
70    ! assumptions of the Model. The program calculates abundances, concentra-
80    ! tions, and isotopic ratios for the Pb, Sr, and Nd systems, and masses of
90    ! major terrestial reservoirs, including intermediate and ending values.
100 Start:OPTION BASE 1
110       DIM M(0:12,2,47,0:3),Vc_cc(47,0:12,30)
120       DIM Gateb1(0:12,47),Gateb2(0:12,47),Gateb3(0:12,47)
130       DIM P_oro(0:12,47,0:2),D_oro(0:12,47,0:2),W_oro(0:12,47,0:2)
140       DIM Total_oro(0:12,47,0:1),Sub_re(0:12,47),Sub_mant(0:12,47)
150       DIM Vert(0:12,47),U_hz(0:12,47),L_hz(0:12,47),Hz(0:12,47)
160       DIM Mantle(0:12,47),D_mant(0:12,47),Mor(0:12,47),Rmor(0:12,47)
170       DIM Outboard(0:12,47),Inboard(0:12,47),A1(47),A4(47),A5(47),A6(47)
180       DIM E_a2(12),F_a3(12),E_b1(12),E_b2(12),E_b3(12),F_c3(12)
190       DIM Sum(0:12,6),Up_seg(0:12,47),Low_seg(0:12,47),Sub_seg(0:12,47)
200       DIM E_up(0:12),E_low(0:12),E_sub(0:12),End(47,47),Mass(47,47)
210       DIM F(47),U(47),L(47),S(47),Conc(47,0:12,30),Abund(47,0:12,30)
220       DIM Sum_up(0:12,47),Sum_low(0:12,47),Sum_sub(0:12,47)
230       DIM Sum_low_c(0:12,47),Low_hz_c(0:12,47),End_seg(47,3,25)
240       DIM Time(47),Age(47),Ratio(47,50),U_eroded(47),V0(47)
250       DIM St1$[5],St2$[5],St3$[5],Cname1$[600],Cname2$[600],Heading$[600]
260       DIM File1$[10],File$[40],Add_file$(0:12)[6],Value$[15]
270       DIM Col(50),Temp1$(23)[18],End_col$(23)[18]
280       DIM Decay_const(8),St_comp(8),St_abund(8)
290       COM F$[30],Ms$(0:1)[13],Drive_string$[13],Vc(100,50)
300       Ms$(0)=":HP9121,700,0"    ! Left drive  3.5"
310       Ms$(1)=":HP9121,700,1"    ! Right drive  3.5"
320       INTEGER G,H,I,J,K,L1,P,Q,R,Cycles,Step,Lo,Go,Int
330       INTEGER Rowlength,Ncolumn,Filesize
340       REAL T,Td,Bias,Uv,Uh,Lh,Sh,T_start,T_ratio,A2,A3,Dp,B1,B2,B3
350       STATUS 1,9;Screen
360       Center=(Screen-60)/2      ! Center of screen
370       Clear                     ! Clear screen
380       OUTPUT 2;"SCRATCH KEY";CHR$(255)&CHR$(88);
390       MAT M= (0)
400       MAT Vc= (0)
410       Step=1
420       Warn_neg=1
430       System$(1)="Pb-Th-U only"
440       System$(2)="Rb-Sr, Sm-Nd only"
450       System$(3)="ALL Isotopes"
460 Iso_abundances:   DATA 1.00,.9928,.0986,.2785,.2387,.1507
470     RESTORE 460
480      FOR H=2 TO 7
490       READ Iso_abund(H)
500      NEXT H
510 Molecular_wts:   DATA 232.038,238.03,87.62,85.47,144.24,150.35
520     RESTORE 510
530       FOR H=2 TO 7
540        READ El_mw(H)
550       NEXT H
```

```
560   !
570       Convert_vc(Vc(*),Ms$(*),Drive_string$,F$,Rev_date$) ! Retrieve data fil
e to be used
580       IF F$[7,9]="End" THEN GOTO End_values
590 Convert_data:  ! This section takes values from Vc(rows,cols) and puts them
into various variables and arrays used by the main program
600       L1=Vc(1,11)                       ! Decay constants
610       L2=Vc(2,11)
620       L3=Vc(3,11)
630       L4=Vc(4,11)
640       L5=Vc(5,11)
650       X0=Vc(1,13)                       ! Starting isotopes ratios
660       X1=Vc(2,13)
670       X2=Vc(3,13)
680       Y0=Vc(4,13)
690       Z0=Vc(5,13)
700       M(0,1,1,0)=Vc(1,15)               ! Starting elmental abundances in Mantle
710       M(1,1,1,0)=Vc(2,15)
720       M(5,1,1,0)=Vc(3,15)
730       M(6,1,1,0)=Vc(4,15)
740       M(7,1,1,0)=Vc(5,15)
750       M(9,1,1,0)=Vc(6,15)
760       M(10,1,1,0)=Vc(7,15)
770       M(12,1,1,0)=Vc(8,15)
780       M(2,1,1,0)=M(1,1,1,0)*X0
790       M(3,1,1,0)=M(1,1,1,0)*X1
800       M(4,1,1,0)=M(1,1,1,0)*X2
810       M(8,1,1,0)=M(7,1,1,0)*Y0
820       M(11,1,1,0)=M(10,1,1,0)*Z0
830       !
840       T=DROUND(Vc(1,17),2)                    ! Time to start
850       Td=DROUND((Vc(1,17)-Vc(2,17)),2)        ! Time interval
860       Cycles=INT((T+Td)*(1/Td))               ! Number of cycles
870       T_start=T
880       Hh=6
890       Gg=1
900       Jj=1
910       Yy=1
920       Zz=3
930      IF Vc(16,11)=1 THEN Model$="VERSION IV"
940      IF Vc(16,11)=2 THEN Model$="EARLY"
950      IF Vc(16,11)=3 THEN Model$="LATE"
960      IF Vc(16,11)<1 OR Vc(16,11)>3 THEN
970        Clear
980        Clunk
990        PRINT TABXY(1,15);FNH$("You didn't specify model in Visicalc file")
1000       PAUSE
1010     END IF
1020 !
1030      Bs=.001              ! Bs is baseline to vertical erosion in kilometers
1040      H0=35/Bs
1050      A3=.0500             ! Fraction of each Ucrust segment above baseline ver
tically eroded (sum u->v)
1060        FOR I=1 TO Cycles
1070         F(I)=Vc(I,19)
```

```
1080       NEXT I
1090     Model_num$=VAL$(Vc(48,21))
1100     Ero$=VAL$(A3)
1110     MAT A1= F*(Vc(50,21))       ! Fraction of Mantle to create MORidge
1120      FOR K=1 TO Cycles
1130        A4(K)=Vc(K,22)           ! Horizontal erosion of Ucrust
1140        A5(K)=Vc(K,22)           ! Horizontal erosion of Lcrust
1150        A6(K)=Vc(K,23)           ! Horizontal erosion of Scrust
1160      NEXT K
1170     MAT U= F*(Vc(50,24))        ! Ucrust mass to be created at each cycle
1180     MAT U_eroded= F*(Vc(50,24))
1190     MAT L= F*(Vc(50,25))        ! Lcrust mass to be created at each cycle
1200     MAT S= F*(Vc(50,26))        ! Scrust mass to be created at each cycle
1210 !
1220     A2=.20          ! Fraction of MORidge to the Dorogene (m->r)
1230     Dp=.14          ! Fraction of vertically-eroded Ucrust to Dorogene (v->d)
1240     B1=1            ! Fraction of Porogene fill to come from Dorogene (d->p)
1250     B3=1-B1         ! Fraction of Porogene fill to come from Worogene (w->p)
1260     B2=.01          ! Fraction of Worogene fill to come from Dorogene (d->w)
1270     FOR H=1 TO 12
1280       E_a2(H)=Vc(H,3)           ! Enrichment through gate A2 (r->d)
1290       E_b1(H)=Vc(H,4)           ! Enrichment through gate B1 (d->p)
1300       E_b2(H)=Vc(H,5)           ! Enrichment through gate B2 (d->w)
1310       E_b3(H)=Vc(H,6)           ! Enrichment through gate B3 (w->p)
1320       F_a3(H)=Vc(H,7)           ! Fractionation through gate A3 (d/p)
1330       F_c3(H)=Vc(H,8)           ! Fractionation through gate C3 (u/l)
1340     NEXT H
1350     MAT E_up= (1.0)
1360     MAT E_low= (1.0)
1370     MAT E_sub= (1.0)
1380 !
1390 St_options: ! Option to print parameters read by convert_vc
1400     Clear
1410     PRINT TABXY(1,1)
1420     PRINT TAB(Center);" KEY    DESCRIPTION for ";FNH$(Model$);",  ";Model_n
um$;"  times upper mantle"
1430     PRINT
1440     PRINT TAB(Center);"  k0      Print starting parameters on printer."
1450     PRINT TAB(Center);"  k1      Print starting parameters on CRT."
1460     PRINT TAB(Center);"  k2      Change A2 or Dp."
1470     PRINT TAB(Center);"  k3      Toggle with reseting B1 fraction value."
1480     PRINT TAB(Center);"  k4      Toggle with reseting B2 value."
1490     PRINT TAB(Center);"  k5      Change (A3) Vertical Erosion."
1500     PRINT TAB(Center);"  k6      HELP screen. Variable descriptions."
1510     PRINT
1520     PRINT TAB(Center);"  k7      Which isotope systematics do you want to c
alculate?"
1530     PRINT TAB(Center);"          U-Th-Pb only, Rb-Sr and Sm-Nd only, or all
 Isotopes"
1540     PRINT TAB(Center);"          Currently ";FNH$(System$(Jj));"is going to
 be calculated."
1550     PRINT
1560     PRINT TAB(Center);"  k8      Return to CATALOG screen."
1570     PRINT TAB(Center);"  k9      Start calculations, RUN the program."
1580     OFF KEY
```

```
1590     ON KEY 0 LABEL "Printer" GOTO Printer
1600     ON KEY 1 LABEL "  CRT" GOTO Crt
1610     ON KEY 5 LABEL " A3=  "&Ero$ GOTO Set_vert_ero
1620     ON KEY 6 LABEL "Const DESCR." GOTO Const_descr
1630     ON KEY 8 LABEL "Get Data" GOTO Start
1640     ON KEY 9 LABEL "RUN" GOTO Begin
1650       ON KEY 2 LABEL "A2, Dp" GOTO Dp_new
1660       IF B1=1 THEN ON KEY 3 LABEL "B1=1.0" GOTO B1_part
1670       IF B1<1 THEN ON KEY 3 LABEL "B1= "&St2$ GOTO B1_all
1680       IF B2=.01 THEN ON KEY 4 LABEL "B2=0.01" GOTO B2_new
1690       IF B2<>.01 THEN ON KEY 4 LABEL "B2= "&St3$ GOTO B2_old
1700       IF Gg=1 AND Hh=6 THEN ON KEY 7 LABEL System$(1) GOTO Hh_sys
1710       IF Gg=7 AND Hh=12 THEN ON KEY 7 LABEL System$(2) GOTO Hh_sys
1720       IF Gg=1 AND Hh=12 THEN ON KEY 7 LABEL System$(3) GOTO Hh_sys
1730     GOTO 1730
1740 Const_descr: Clear
1750              OFF KEY
1760              PRINT TABXY(1,10)
1770     PRINT "               * * * CONSTANT DESCRIPTIONS * * *"
1780     PRINT
1790     PRINT "          Constant       DESCRIPTION"
1800     PRINT
1810     PRINT "            A2           Fraction of MORidge to Dorogene (r->d)"
1820     PRINT "                           A2=:  "&FNH$(VAL$(A2))
1830     PRINT "            Dp           Fraction of vertically-eroded Ucrust to"
1840     PRINT "                           Dorogene (v->p). Dp=:  "&FNH$(VAL$(Dp))
1850     PRINT "            B1           Fraction of Porogene fill to come from"
1860     PRINT "                           Dorogene (d->p). (B1 + B3 = 1.0)."
1870     PRINT "                           B1=: "&FNH$(VAL$(B1))
1880     PRINT "            B2           Fraction of Worogene fill to come from"
1890     PRINT "                           Dorogene (d->w). (B1 + B3 = 1.0)."
1900     PRINT "                           B2=: "&FNH$(VAL$(B2))
1910     PRINT "            A3          Fraction of each Ucrust segment above"
1920     PRINT "                           baseline vertically eroded (sum u->v)."
1930     PRINT "                           A3=: "&FNH$(VAL$(A3))
1940     ON KEY 9 LABEL "  RETURN" GOTO St_options
1950     GOTO 1950
1960 Hh_sys: Clear ! calculate all isotopes
1970         PRINT TABXY(1,10);"Choose which systematics you wish to have calcul
ated:"
1980         PRINT
1990         PRINT TAB(Center);"     1--  Pb-Th-U only"
2000         PRINT TAB(Center);"     2--  Rb-Sr, Sm-Nd only"
2010         PRINT TAB(Center);"     3--  All isotopes"
2020         PRINT
2030         DISP "Enter the number of the choice you've made."
2040         INPUT Answer
2050         SELECT Answer
2060         CASE 1
2070           Hh=6
2080           Gg=1
2090           Yy=1
2100           Zz=3
2110           Jj=1
2120         CASE 2
```

```
2130           Hh=12
2140           Gg=7
2150           Yy=4
2160           Zz=7
2170           Jj=2
2180         CASE 3
2190           Hh=12
2200           Gg=1
2210           Yy=1
2220           Zz=7
2230           Jj=3
2240         CASE ELSE
2250         GOTO Hh_sys
2260         END SELECT
2270         GOTO St_options
2280 Set_vert_ero: Clear
2290               PRINT TAB(Center);"Enter the fractional value of vertical ero
sion (e.g. 0.04 for 4%):"
2300               INPUT A3
2310               Ero$=VAL$(A3)
2320               GOTO St_options
2330 Dp_new: Clear
2340         PRINT "Enter the fraction of MORidge to create Dorogene"
2350         PRINT "Currently, A2= "&FNH$(VAL$(A2))
2360         PRINT "(Press ENTER, if you don't want to change the value.)"
2370         INPUT A2
2380         Clear
2390           PRINT "Enter the fraction of Dp that you wish to use"
2400           PRINT "(Remember Dp is the fraction of vertically-eroded Ucrust"
2410           PRINT " to go to Dorogene)"
2420           PRINT "Currently, Dp= "&FNH$(VAL$(Dp))
2430         PRINT "(Press ENTER, if you don't want to change the value.)"
2440           INPUT Dp
2450         GOTO St_options
2460 B1_all: Clear
2470          B1=1
2480          B3=1-B1
2490      GOTO St_options
2500 B1_part: Clear
2510           PRINT "Enter the fraction of B1 that you wish to use"
2520           PRINT "(Remember B1 is the fraction of Porogene fill to come"
2530           PRINT " from Dorogene so that B1+B3=1)"
2540           PRINT "Currently, B1=  "&FNH$(VAL$(B1))
2550         PRINT "(Press ENTER, if you don't want to change the value.)"
2560           INPUT B1
2570           B3=1-B1
2580           Ch=DROUND(B1,3)
2590           St2$=VAL$(Ch)
2600         GOTO St_options
2610 B2_old: Clear
2620          B2=.01
2630          GOTO St_options
2640 B2_new: Clear
2650           PRINT "Enter the fraction of B2 that you wish to use"
2660           PRINT "(Remember B2 is the fraction of Worogene fill to come"
```

```
2670           PRINT " from Dorogene)"
2680           PRINT "Currently, B2=  "&FNH$(VAL$(B2))
2690         PRINT "(Press ENTER, if you don't want to change the value.)"
2700           INPUT B2
2710           Ch=DROUND(B2,3)
2720           St3$=VAL$(Ch)
2730         GOTO St_options
2740 Printer:PRINTER IS 701;WIDTH 80
2750     PRINT ""
2760     GOTO Print_param
2770 Crt:PRINTER IS CRT
2780 Print_param:!
2790             Clear
2800     PRINT "        <<<<<<<<<<<  PLUMBOTECTONICS MODELLING PROGRAM  >>>>>>>>
>>>"
2810     PRINT
2820     PRINT "                    "&FNUn$("TABLE OF STARTING PARAMETERIC VALUE
S")
2830     PRINT
2840     PRINT "                                  Constants"
2850     PRINT
2860     PRINT "       The start time is ";T_start;" B.Y.A., with a time decreme
nt of ";Td
2870     PRINT "                         and ";Cycles;" cycles."
2880     PRINT
2890     PRINT "             A3(Vertical erosion)= ";A3
2900     PRINT "             Bs(Baseline, in km)= ";Bs
2910     PRINT "             A2(Fraction of MORidge to create Dorogene)= ";A2
2920     PRINT "             Dp(Fraction of vertically-eroded Ucrust to go"
2930     PRINT "               to Dorogene)= ";Dp
2940     PRINT "             B1(Fraction of Porogene fill from Dorogene)= ";B1
2950     PRINT "             B2(Fraction of Worogene filled from Dorogene)= ";B2
2960     PRINT "             B3(Fraction of Porogene fill from Worogene) = ";B3
2970     PRINT
2980     PRINT "             Decay Constants  Start Iso Comps  Start El Abund"
2990     PRINT "            [=============================================]"
3000     PRINT
3010     PRINT "              238 U ";L1;"   206/204";X0;"    Mass  ";M(0,1,1,0)
3020     PRINT "              235 U ";L2;"    207/204";X1;"    204 Pb";M(1,1,1,0)
3030     PRINT "              232 Th";L3;"   208/204";X2;"   232 Th";M(5,1,1,0)
3040     PRINT "              87 Rb ";L4;"     87/86  ";Y0;"  238 U ";M(6,1,1,0)
3050     PRINT "              147 Sm";L5;"    143/144";Z0;"  86 Sr ";M(7,1,1,0)
3060     PRINT "                                          87 Rb ";M(9,1,1,
0)
3070     PRINT "                                          144 Nd";M(10,1,1
,0)
3080     PRINT "                                          147 Sm";M(12,1,1
,0)
3090     PRINT
3100     PRINT
3110     DATA "204 Pb","206 Pb","207 Pb","208 Pb","232 Th","238 U","86 Sr","87 S
r","87 Rb","144 Nd","143 Nd","147 Sm"
3120     RESTORE 3110
3130     FOR I=1 TO 12
3140       READ End_col$(I)
```

```
3150      NEXT I
3160      PRINT "Isotope Name          E_a2(H)     F_a3(H)    E_b1(H)    E_b2(H)    Eb
3(H)    F_c3(H)"
3170      PRINT "=============================================================
==============="
3180      PRINT
3190      FOR H=6g TO Hh
3200        PRINT USING "18A,2X,6(0000.000,2X)";End_col$(H),E_a2(H),F_a3(H),E_b1(
H),E_b2(H),E_b3(H),F_c3(H)
3210      NEXT H
3220      PRINT CHR$(12)  !form feed paper
3230      PRINT "                                Time Dependent Values"
3240      PRINT
3250      PRINT "    Cycle     A1(K)      A5(K)      A6(K)      U(K)       L(K)       S
(K)"
3260      PRINT "   [============================================================
====]"
3270      PRINT
3280      FOR K=1 TO Cycles
3290        PRINT USING "7X,K,2X,6(D.6D,2X)";K,A1(K),A4(K),A5(K),U(K),L(K),S(K)
3300      NEXT K
3310      PRINT CHR$(12)
3320      PRINTER IS CRT
3330      GOTO St_options
3340 Begin:!
3350      K=1
3360 !
3370      Clear
3380      OFF KEY
3390      PRINT TABXY(1,7)
3400      PRINT TAB(Center+25);CHR$(129)&" RUNNING "&CHR$(128)
3410      PRINT
3420      PRINT TAB(Center+20);CHR$(129)&" PLEASE DO NOT DISTURB "&CHR$(128)
3430      PRINT
3440      PRINT
3450      PRINT TABXY(10,29);"Press "&FNH$("k9")&" to abort RUN and return to the
 CATALOG screen."
3460      ON KEY 9 LABEL "  ESCAPE" GOTO Start
3470  !
3480 Main_prog:    !Main program to calculate matrix M and other matrices
3490      LOOP
3500                PRINT TABXY(1,12)
3510                PRINT TAB(Center+10);"Calculating mass and isotope abundance"
3520                PRINT TAB(Center+10);"for ";K;" cycle of ";Cycles;" cycles."
3530 Mass:          ! Calculate mass (H=0) of all matrices
3540      FOR J=1 TO K-1
3550        IF M(0,1,J,1)=0 THEN
3560          V0(J)=0
3570        ELSE
3580          V0(J)=(M(0,1,J,1)-U_eroded(J)/H0)/M(0,1,J,1)
3590        END IF
3600        Uv=M(0,1,J,1)*V0(J)*A3  ! Vertical erosion of Ucrust segment to basel
evel
3610        Uh=M(0,1,J,1)*(V0(J)*A4(K)*(1-A3)+(1-V0(J))*A4(K))  ! Horizontal eros
ion of Ucrust segment
```

```
3620       U_eroded(J)=U_eroded(J)*(1-A4(K))  ! Ucrust segment remaining
3630       M(0,2,J,1)=M(0,1,J,1)-Uv-Uh
3640       Vert(0,K)=Vert(0,K)+Uv  ! Vertical erosion of total Ucrust
3650       U_hz(0,K)=U_hz(0,K)+Uh  ! Horizontal erosion of total Ucrust
3660     NEXT J
3670     FOR J=1 TO K-1
3680       Lh=M(0,1,J,2)*A5(K)
3690       M(0,2,J,2)=M(0,1,J,2)-Lh
3700       L_hz(0,K)=L_hz(0,K)+Lh  ! Horizontal erosion of total Lcrust
3710     NEXT J
3720     FOR J=1 TO K
3730       Low_hz_c(0,K)=Low_hz_c(0,K)+L_hz(0,K)
3740     NEXT J
3750     FOR J=1 TO K
3760         Sum_low_c(0,K)=Sum_low_c(0,K)+L(J)
3770     NEXT J
3780     Outboard(0,K)=Dp*Vert(0,K)  ! Fraction of vertically-eroded Ucrust to D
orogene
3790     Inboard(0,K)=(1-Dp)*(Vert(0,K))  ! Fraction of vertically-eroded Ucrust
 to Porogene
3800     Hz(0,K)=U_hz(0,K)+L_hz(0,K) ! Total horizontal erosion of Ucrust and Lc
rust
3810     P_oro(0,K,0)=Inboard(0,K)+Hz(0,K) ! Mass of Porogene before communicati
on
3820     FOR J=1 TO K-1
3830       Sh=M(0,1,J,3)*A6(K)
3840       Sub_re(0,K)=Sh+Sub_re(0,K)
3850       M(0,2,J,3)=M(0,1,J,3)-Sh
3860     NEXT J
3870     Mantle(0,K)=M(0,1,K,0)*A1(K)  ! Mass of Mantle to MOR
3880     Mor(0,K)=A2*Mantle(0,K)  ! Mass of MOR to Dorogene
3890     Rmor(0,K)=(1-A2)*Mantle(0,K)  ! Mass of MOR back to Mantle
3900     D_oro(0,K,0)=Mor(0,K)+Outboard(0,K)  ! Mass of Dorogene before communic
ation
3910     Gateb3(0,K)=B3*(U(K)+L(K)-P_oro(0,K,0)) ! Mass through Gateb3 needed to
 fill Porogene
3920 IF Gateb3(0,K)<0 THEN GOSUB Illegit_neg   ! Beep warning if calc neg #
3930     Gateb2(0,K)=B2*S(K)+Gateb3(0,K)  ! Mass through Gateb2 is B2*100 % of S
(K) plus any mass through Gateb3
3940     D_oro(0,K,1)=D_oro(0,K,0)-Gateb2(0,K)  ! Mass of Dorogene after materia
l leaves through Gateb2
3950     Gateb1(0,K)=B1*(U(K)+L(K)-P_oro(0,K,0))  ! Mass needed to fill Porogene
3960 IF Gateb1(0,K)<0 THEN GOSUB Illegit_neg  ! Beep warning if calc neg #
3970     D_oro(0,K,2)=D_oro(0,K,1)-Gateb1(0,K)  ! Mass of Dorogene after materia
l leaves through Gateb1
3980 !
3990     Sub_mant(0,K)=S(K)-Sub_re(0,K)-Gateb2(0,K)+Gateb3(0,K)  ! Mass of Mantl
e meeded to fill Worogene
4000 IF Sub_mant(0,K)<0 THEN GOSUB Illegit_neg  ! Beep warning if calc neg #
4010     W_oro(0,K,0)=Sub_re(0,K)+Sub_mant(0,K)  ! Mass Worogene before communic
ation
4020     Total_oro(0,K,0)=D_oro(0,K,0)+P_oro(0,K,0)+W_oro(0,K,0)
4030     W_oro(0,K,1)=W_oro(0,K,0)+Gateb2(0,K)  ! Mass of Worogene after materia
l arrives through Gateb2
4040     W_oro(0,K,2)=W_oro(0,K,1)-Gateb3(0,K)  ! Mass of Worogene after materia
```

```
l leaves through Gateb3
4050     P_oro(0,K,1)=P_oro(0,K,0)+Gateb1(0,K)+Gateb3(0,K)  ! Final mass of Poro
gene
4060     Total_oro(0,K,1)=D_oro(0,K,2)+P_oro(0,K,1)+W_oro(0,K,2)  ! Final mass o
f total orogene
4070     M(0,2,K,0)=M(0,1,K,0)-Mantle(0,K)+D_oro(0,K,2)-Sub_mant(0,K)+Rmor(0,K)
 ! Mass of Mantle after orogeny
4080     M(0,2,K,1)=U(K)                          ! Mass of new Ucrust segment
4090     M(0,2,K,2)=L(K)                          ! Mass of new Lcrust segment
4100     M(0,2,K,3)=S(K)                          ! Mass of new Scrust segment
4110 !
4120 Isotopes:   ! This loop calculates Pb, Sr, & Nd isotope systematics
4130     FOR H=Gg TO Hh
4140       FOR J=1 TO K-1
4150         Uv=M(H,1,J,1)*V0(J)*A3*E_up(H)  ! Moles removed from Ucrust by vert
ical erosion
4160         Uh=M(H,1,J,1)*(V0(J)*A4(K)*(1-A3)+(1-V0(J))*A4(K))*E_up(H) ! Moles
removed from Ucrust by horizontal erosion
4170         U_eroded(J)=U_eroded(J)*(1-A4(K))
4180         M(H,2,J,1)=M(H,1,J,1)-Uv-Uh  ! Moles remaining in Ucrust
4190         Vert(H,K)=Vert(H,K)+Uv
4200         U_hz(H,K)=U_hz(H,K)+Uh
4210       NEXT J
4220       FOR J=1 TO K-1
4230         Lh=M(H,1,J,2)*A5(K)*E_low(H)  ! Moles removed from Lcrust by horizo
ntal erosion
4240         M(H,2,J,2)=M(H,1,J,2)-Lh  ! Moles remaining in Lcrust
4250         L_hz(H,K)=L_hz(H,K)+Lh
4260       NEXT J
4270       Hz(H,K)=U_hz(H,K)+L_hz(H,K)
4280       FOR J=1 TO K-1
4290         Sh=M(H,1,J,3)*A6(K)*E_sub(H)  ! Moles removed from Scrust by horizo
ntal erosion
4300         Sub_re(H,K)=Sh+Sub_re(H,K)
4310         M(H,2,J,3)=M(H,1,J,3)-Sh  ! Moles remaining in Scrust
4320       NEXT J
4330       Outboard(H,K)=FNEmoles(Vert(H,K),Outboard(0,K),Inboard(0,K),F_a3(H))
 ! Moles of vertically-eroded material to Dorogene
4340       Inboard(H,K)=Vert(H,K)-Outboard(H,K)
4350       P_oro(H,K,0)=Inboard(H,K)+Hz(H,K) ! Moles in Porogene before communic
ation
4360       Mantle(H,K)=M(H,1,K,0)*A1(K)  ! Mass fraction of Mantle to MOR
4370       Mor(H,K)=FNEmoles(Mantle(H,K),Mor(0,K),Rmor(0,K),E_a2(H))  ! Moles fr
om Mantle to MOR
4380       Rmor(H,K)=Mantle(H,K)-Mor(H,K)  ! Moles returned to Mantle from MOR
4390       D_oro(H,K,0)=Mor(H,K)+Outboard(H,K) ! Moles in Dorogene before commun
ication
4400       Gateb2(H,K)=FNEmoles(D_oro(H,K,0),Gateb2(0,K),D_oro(0,K,1),E_b2(H))
! Gateb2 (d->w)
4410       D_oro(H,K,1)=D_oro(H,K,0)-Gateb2(H,K)  ! Moles remaining in Dorogene
after material leaves through Gateb2
4420       Gateb1(H,K)=FNEmoles(D_oro(H,K,1),Gateb1(0,K),D_oro(0,K,2),E_b1(H))
! Gateb1 (d->p)
4430       D_oro(H,K,2)=D_oro(H,K,1)-Gateb1(H,K)  ! Moles remaining in Dorogene
after material leaves through Gateb1
```

```
4440        Sub_mant(H,K)=(Sub_mant(0,K)/(M(0,1,K,0)-Mor(0,K)))*(M(H,1,K,0)-Mor(H
,K))  ! Gatea7 (m->w)
4450        W_oro(H,K,0)=Sub_re(H,K)+Sub_mant(H,K)  ! Moles in Worogene before co
mmunication
4460        Total_oro(H,K,0)=D_oro(H,K,0)+P_oro(H,K,0)+W_oro(H,K,0)
4470        W_oro(H,K,1)=W_oro(H,K,0)+Gateb2(H,K)  ! Moles in Worogene after mate
rial arrives through Gateb2
4480        Gateb3(H,K)=FNEmoles(W_oro(H,K,1),Gateb3(0,K),W_oro(0,K,2),E_b3(H))
! Gateb3 (w->p)
4490        W_oro(H,K,2)=W_oro(H,K,1)-Gateb3(H,K)  ! Moles remaining in Worogene
after material leaves through Gateb3
4500        P_oro(H,K,1)=P_oro(H,K,0)+Gateb1(H,K)+Gateb3(H,K)  ! Moles in Porogen
e after communication
4510        M(H,2,K,1)=FNEmoles(P_oro(H,K,1),U(K),L(K),F_c3(H)) ! Moles from Poro
gene through Gatec3 to new Ucrust
4520        M(H,2,K,2)=P_oro(H,K,1)-M(H,2,K,1)  ! Moles from Porogene through Gat
ec3 to new Lcrust
4530        M(H,2,K,3)=W_oro(H,K,2)  ! Moles from Worogene through Gatec4 to new
Subcrust
4540        M(H,2,K,0)=M(H,1,K,0)-Mantle(H,K)+D_oro(H,K,2)-Sub_mant(H,K)+Rmor(H,K
)  ! Moles in Mantle after orogeny
4550        Total_oro(H,K,1)=D_oro(H,K,2)+P_oro(H,K,1)+W_oro(H,K,2)
4560      NEXT H
4570 !
4580 Increment:   !Radioactive decay during time increment k
4590      FOR H=0 TO Hh  ! Reservoirs after orogeny
4600        D_mant(H,K)=M(H,2,K,0)  ! Mass and moles in Mantle after the orogeny
4610        FOR J=1 TO K
4620          Sum_up(H,K)=M(H,2,J,1)+Sum_up(H,K) ! Summation of material in Ucrus
t segments
4630          Sum_low(H,K)=M(H,2,J,2)+Sum_low(H,K)  ! Summation of material in Lc
rust segments
4640          Sum_sub(H,K)=M(H,2,J,3)+Sum_sub(H,K)  ! Summation of material in Sc
rust segments
4650        NEXT J
4660      NEXT H
4670   IF T<=4.5 THEN Td=.1
4680        M(0,1,K+1,0)=M(0,2,K,0)  ! Mass in Mantle ready for next cycle
4690          IF Gg<>7 OR Hh<12 THEN
4700        M(1,1,K+1,0)=M(1,2,K,0)  ! Moles in Mantle ready for next cycle
4710        M(5,1,K+1,0)=M(5,2,K,0)
4720        M(6,1,K+1,0)=M(6,2,K,0)
4730        M(2,1,K+1,0)=M(2,2,K,0)+M(6,2,K,0)*(EXP(L1*T)-EXP(L1*(T-(Td))))
4740        M(3,1,K+1,0)=M(3,2,K,0)+(M(6,2,K,0)/137.88)*(EXP(L2*T)-EXP(L2*(T-(Td)
)))
4750        M(4,1,K+1,0)=M(4,2,K,0)+M(5,2,K,0)*(EXP(L3*T)-EXP(L3*(T-(Td))))
4760        END IF
4770          IF Gg>1 OR Hh<>6 THEN
4780        M(7,1,K+1,0)=M(7,2,K,0)
4790        M(9,1,K+1,0)=M(9,2,K,0)
4800        M(8,1,K+1,0)=M(8,2,K,0)+M(9,2,K,0)*(EXP(L4*T)-EXP(L4*(T-(Td))))
4810        M(10,1,K+1,0)=M(10,2,K,0)
4820        M(12,1,K+1,0)=M(12,2,K,0)
4830        M(11,1,K+1,0)=M(11,2,K,0)+M(12,2,K,0)*(EXP(L5*T)-EXP(L5*(T-(Td))))
4840        END IF
```

```
4850     FOR J=1 TO K
4860      FOR L0=1 TO 3
4870       M(0,1,J,L0)=M(0,2,J,L0)
4880         IF Gg=7 AND Hh=12 THEN 4960
4890       M(1,1,J,L0)=M(1,2,J,L0)
4900       M(5,1,J,L0)=M(5,2,J,L0)
4910       M(6,1,J,L0)=M(6,2,J,L0)
4920       M(2,1,J,L0)=M(2,2,J,L0)+M(6,2,J,L0)*(EXP(L1*T)-EXP(L1*(T-(Td))))
4930       M(3,1,J,L0)=M(3,2,J,L0)+(M(6,2,J,L0)/137.88)*(EXP(L2*T)-EXP(L2*(T-(Td
))))
4940       M(4,1,J,L0)=M(4,2,J,L0)+M(5,2,J,L0)*(EXP(L3*T)-EXP(L3*(T-(Td))))
4950         IF Gg=1 AND Hh=6 THEN GOTO 5020
4960       M(7,1,J,L0)=M(7,2,J,L0)
4970       M(9,1,J,L0)=M(9,2,J,L0)
4980       M(8,1,J,L0)=M(8,2,J,L0)+M(9,2,J,L0)*(EXP(L4*T)-EXP(L4*(T-(Td))))
4990       M(10,1,J,L0)=M(10,2,J,L0)
5000       M(12,1,J,L0)=M(12,2,J,L0)
5010       M(11,1,J,L0)=M(11,2,J,L0)+M(12,2,J,L0)*(EXP(L5*T)-EXP(L5*(T-(Td))))
5020      NEXT L0
5030     NEXT J
5040     EXIT IF K>=Cycles  ! Exit Main_prog when done
5050     T=DROUND(T-Td,3)
5060     K=K+1
5070     END LOOP
5080 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
5090 Seg_input:  Clear
5100   FOR H=0 TO Hh
5110     FOR J=1 TO K
5120       Up_seg(H,J)=M(H,1,J,1)  ! Final mass and moles in each Ucrust segment
5130       Low_seg(H,J)=M(H,1,J,2) ! Final mass and moles in each Lcrust segment
5140       Sub_seg(H,J)=M(H,1,J,3) ! Final mass and moles in each Scrust segment
5150     NEXT J
5160   NEXT H
5170 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
5180 !                             File Name
5190 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
5200 File_name: Clear
5210     PRINT TABXY(1,10)
5220     PRINT TAB(Center);" Choose a file name exactly 4 characters long which
"
5230     PRINT TAB(Center);" characterizes this particular run (e.g. A___ or Std
1)!"
5240     PRINT TAB(Center);" The default file name is ""____""."
5250     INPUT File1$
5260     IF File1$="" THEN File1$="____"
5270 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
5280 !                         Output Options
5290 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
5300 Options: Clear
5310     PRINT TABXY(1,10)
5320     PRINT TAB(Center);"    * * * * LIST OF OUTPUT OPTIONS * * * *"
5330     PRINT
5340     PRINT TAB(Center);" KEY     DESCRIPTION"
5350     PRINT
5360     PRINT TAB(Center);"  k0     Isotopic Ratios (ISOPLOT files)"
```

```
5370     PRINT TAB(Center);"  k1     Concentrations, ppm (ISOPLOT files)"
5380     PRINT TAB(Center);"  k2     Abundances, E+15 moles (ISOPLOT files)"
5390     PRINT TAB(Center);"  k3     Masses, E+24 grams (ISOPLOT files)"
5400     PRINT
5410     PRINT TAB(Center);"  k4     Ending Values (Tables and ISOPLOT file for
final segments)"
5420     PRINT
5430     PRINT TAB(Center);"  k5     Start over.  Go back to CATALOG screen."
5440     PRINT TAB(Center);"  k6     Rename output files.  Current name is "&FNH
$(File1$)
5450     PRINT TAB(Center);"  k7     Toggle to save values every 100 M.Y. or 400
 M.Y."
5460     PRINT TAB(Center);"  k9     Load ISOPLOT"
5470     PRINT
5480     PRINT TAB(Center);"Output option?  Make a selection by pressing appropr
iate soft key."
5490     PRINT
5500     IF Warn_neg=0 THEN
5510         PRINT TAB(Center);FNHb1$("WARNING NEGATIVE VALUES WERE CALCULATED")
5520         Clunk
5530     END IF
5540     OFF KEY
5550     ON KEY 0 LABEL "Ratios" GOTO Ratios
5560     ON KEY 1 LABEL "Conc's" GOTO Concentrations
5570     ON KEY 2 LABEL "Abund's" GOTO Abundances
5580     ON KEY 3 LABEL "Masses" GOTO Masses
5590     ON KEY 4 LABEL "Ending Values" GOTO Ending_values
5600     ON KEY 5 LABEL "NEW" GOTO Start
5610     ON KEY 6 LABEL "New name" GOTO File_name
5620     ON KEY 9 LABEL "Load ISOPLOT" GOTO Quit
5630       IF Step=1 THEN ON KEY 7 LABEL "100 M.Y." GOTO Step_fcn4
5640       IF Step=4 THEN ON KEY 7 LABEL "400 M.Y." GOTO Step_fcn1
5650 Spin:GOTO Spin
5660 Step_fcn4: Clear
5670            Step=4
5680            GOTO Options
5690 Step_fcn1: Clear
5700            Step=1
5710            GOTO Options
5720 Ratios:   !
5730    !  This subroutine calculates the isotopic ratios
5740     MAT Ratio= (0)
5750     Clear
5760     PRINT TABXY(1,10);"       * * * ISOTOPE RATIOS * * *"
5770     PRINT
5780     PRINT TAB(Center);" KEY    DESCRIPTION"
5790     PRINT
5800     PRINT TAB(Center);"  k0    Create ";FNH$("average");" U-Th-Pb ratios"
5810     IF Hh=12 THEN PRINT TAB(Center);"  k5    Create ";FNH$("average");" Rb-
Sr and Sm-Nd ratios"
5820     PRINT TAB(Center);"  K2    Create ";FNH$("orogenic");" U-Th-Pb ratios"
5830     IF Hh=12 THEN PRINT TAB(Center);"  k7    Create ";FNH$("orogenic");" Rb
-Sr and Sm-Nd ratios"
5840     PRINT
5850     PRINT TAB(Center);"  k9    Go back to output options"
```

```
5860      OFF KEY
5870      ON KEY 0 LABEL "File ave. Pb" GOTO Ave_pb
5880      IF Hh=12 THEN ON KEY 5 LABEL "File ave. Sr" GOTO Ave_sr
5890      ON KEY 2 LABEL "File oro. Pb" GOTO Oro_pb
5900      IF Hh=12 THEN ON KEY 7 LABEL "File oro. Sr" GOTO Oro_sr
5910      ON KEY 9 LABEL "Escape" GOTO Options
5920      GOTO 5920
5930 !
5940 Ave_pb:     Clear
5950      PRINT TABXY(10,10);FNH$("Please be patient. I'm working on it!!")
5960      H=1
5970      FOR K=1 TO Cycles STEP Step
5980        T_ratio=T_start-((K-1)*Td)
5990          IF T_ratio<0 THEN T_ratio=0
6000        Ratio(H,1)=DROUND(T_ratio,2)
6010        IF D_mant(1,K)=0 THEN GOTO 6060
6020        Ratio(H,2)=D_mant(2,K)/D_mant(1,K)
6030        Ratio(H,3)=D_mant(3,K)/D_mant(1,K)
6040        Ratio(H,4)=D_mant(4,K)/D_mant(1,K)
6050        Ratio(H,5)=D_mant(6,K)/D_mant(1,K)
6060        IF D_mant(6,K)=0 THEN GOTO 6500
6070        Ratio(H,6)=D_mant(5,K)/D_mant(6,K)
6080        IF Mor(1,K)=0 THEN GOTO 6130
6090        Ratio(H,7)=Mor(2,K)/Mor(1,K)
6100        Ratio(H,8)=Mor(3,K)/Mor(1,K)
6110        Ratio(H,9)=Mor(4,K)/Mor(1,K)
6120        Ratio(H,10)=Mor(6,K)/Mor(1,K)
6130        IF Mor(6,K)=0 THEN GOTO 6580
6140        Ratio(H,11)=Mor(5,K)/Mor(6,K)
6150        IF Sum_up(1,K)=0 THEN GOTO 6200
6160        Ratio(H,12)=Sum_up(2,K)/Sum_up(1,K)  ! Note that sum is average
6170        Ratio(H,13)=Sum_up(3,K)/Sum_up(1,K)
6180        Ratio(H,14)=Sum_up(4,K)/Sum_up(1,K)
6190        Ratio(H,15)=Sum_up(6,K)/Sum_up(1,K)
6200        IF Sum_up(6,K)=0 THEN GOTO 6660
6210        Ratio(H,16)=Sum_up(5,K)/Sum_up(6,K)
6220        IF Sum_low(1,K)=0 THEN GOTO 6270
6230        Ratio(H,17)=Sum_low(2,K)/Sum_low(1,K)
6240        Ratio(H,18)=Sum_low(3,K)/Sum_low(1,K)
6250        Ratio(H,19)=Sum_low(4,K)/Sum_low(1,K)
6260        Ratio(H,20)=Sum_low(6,K)/Sum_low(1,K)
6270        IF Sum_low(6,K)=0 THEN GOTO 6740
6280        Ratio(H,21)=Sum_low(5,K)/Sum_low(6,K)
6290        IF Sum_sub(1,K)=0 THEN GOTO 6340
6300        Ratio(H,22)=Sum_sub(2,K)/Sum_sub(1,K)
6310        Ratio(H,23)=Sum_sub(3,K)/Sum_sub(1,K)
6320        Ratio(H,24)=Sum_sub(4,K)/Sum_sub(1,K)
6330        Ratio(H,25)=Sum_sub(6,K)/Sum_sub(1,K)
6340        IF Sum_sub(6,K)=0 THEN GOTO 6820
6350        Ratio(H,26)=Sum_sub(5,K)/Sum_sub(6,K)
6360      IF K=1 AND Step=4 THEN K=2
6370      H=H+1
6380      NEXT K
6390      Go=H
6400 Ave_pb_file: !
```

```
6410                Clear
6420      PRINT TABXY(1,10)
6430      PRINT TAB(Center);"Writing file for average Pb isotopic ratios for: ";F
NH$(File1$)
6440      Cname1$[1,252]=""
6450      Cname2$[1,252]=""
6460      DATA " "," "," "," "
6470      DATA " ","TIME","Mantle","206/204","Mantle","207/204","Mantle","208/204
"
6480      DATA "Mantle","238U/204","Mantle","232/238U"
6490      DATA "MORB","206/204","MORB","207/204","MORB","208/204"
6500      DATA "MORB","238U/204","MORB","232/238U"
6510      DATA "Upp_ave","206/204","Upp_ave","207/204","Upp_ave","208/204"
6520      DATA "Upp_ave","238U/204","Upp_ave","232/238U"
6530      DATA "Low_ave","206/204","Low_ave","207/204","Low_ave","208/204"
6540      DATA "Low_ave","238U/204","Low_ave","232/238U"
6550      DATA "Sub_ave","206/204","Sub_ave","207/204","Sub_ave","208/204"
6560      DATA "Sub_ave","238U/204","Sub_ave","232/238U"
6570      RESTORE 6460
6580      FOR I=1 TO 28
6590        S1=9*I-8
6600        S2=S1+8
6610        READ C1$,C2$
6620        Cname1$[S1,S2]=C1$
6630        Cname2$[S1,S2]=C2$
6640      NEXT I
6650      Rowlength=252
6660      Heading$[1,252]=""
6670      File$=File1$&"AvePbA"&Drive_string$
6680      Heading$[1,252]="Average PB isotopic ratios for file "&File1$
6690      Ncolumn=28
6700      Filesize=100
6710    Writefile1(Ratio(*),Heading$,File$,Cname1$,Cname2$,50,Go,252,28)
6720      GOTO Ratios
6730    !
6740  Ave_sr: !
6750      Clear
6760      PRINT TABXY(1,10);FNH$("Please be patient, I'm thinking!!")
6770      H=1
6780      FOR K=1 TO Cycles STEP Step
6790        T_ratio=T_start-((K-1)*Td)
6800          IF T_ratio<0 THEN T_ratio=0
6810        Ratio(H,1)=DROUND(T_ratio,2)
6820        IF D_mant(7,K)=0 THEN GOTO 6850
6830        Ratio(H,2)=D_mant(8,K)/D_mant(7,K)
6840        Ratio(H,3)=D_mant(9,K)/D_mant(7,K)
6850        IF D_mant(10,K)=0 THEN GOTO 7390
6860        Ratio(H,4)=D_mant(11,K)/D_mant(10,K)
6870        Ratio(H,5)=D_mant(12,K)/D_mant(10,K)
6880        IF Mor(7,K)=0 THEN GOTO 6910
6890        Ratio(H,6)=Mor(8,K)/Mor(7,K)
6900        Ratio(H,7)=Mor(9,K)/Mor(7,K)
6910        IF Mor(10,K)=0 THEN GOTO 7460
6920        Ratio(H,8)=Mor(11,K)/Mor(10,K)
6930        Ratio(H,9)=Mor(12,K)/Mor(10,K)
```

```
6940        IF Sum_up(7,K)=0 THEN GOTO 6970
6950        Ratio(H,10)=Sum_up(8,K)/Sum_up(7,K)
6960        Ratio(H,11)=Sum_up(9,K)/Sum_up(7,K)
6970        IF Sum_up(10,K)=0 THEN GOTO 7530
6980        Ratio(H,12)=Sum_up(11,K)/Sum_up(10,K)
6990        Ratio(H,13)=Sum_up(12,K)/Sum_up(10,K)
7000        IF Sum_low(7,K)=0 THEN GOTO 7030
7010        Ratio(H,14)=Sum_low(8,K)/Sum_low(7,K)
7020        Ratio(H,15)=Sum_low(9,K)/Sum_low(7,K)
7030        IF Sum_low(10,K)=0 THEN GOTO 7600
7040        Ratio(H,16)=Sum_low(11,K)/Sum_low(10,K)
7050        Ratio(H,17)=Sum_low(12,K)/Sum_low(10,K)
7060        IF Sum_sub(7,K)=0 THEN GOTO 7090
7070        Ratio(H,18)=Sum_sub(8,K)/Sum_sub(7,K)
7080        Ratio(H,19)=Sum_sub(9,K)/Sum_sub(7,K)
7090        IF Sum_sub(10,K)=0 THEN GOTO 7670
7100        Ratio(H,20)=Sum_sub(11,K)/Sum_sub(10,K)
7110        Ratio(H,21)=Sum_sub(12,K)/Sum_sub(10,K)
7120      IF K=1 AND Step=4 THEN K=2
7130      H=H+1
7140      NEXT K
7150      Go=H
7160 Ave_sr_file:        !
7170      Clear
7180      PRINT TABXY(1,10)
7190      PRINT TAB(Center);"Writing file for average Sr isotopic ratios for: "&F
NH$(File1$)
7200      Cname1$[1,207]=""
7210      Cname2$[1,207]=""
7220      DATA " "," "," "," "
7230      DATA " ","TIME","Mantle", "87Sr/Sr"
7240      DATA "Mantle","87Rb/86","Mantle","143/144","Mantle","147/144"
7250      DATA "MORB", "87Sr/Sr"
7260      DATA "MORB","87Rb/86","MORB","143/144","MORB","147/144"
7270      DATA "Upp_ave","87Sr/86"
7280      DATA "Upp_ave","87Rb/86","Upp_ave","143/144","Upp_ave","147/144"
7290      DATA "Low_ave","87Sr/86"
7300      DATA "Low_ave","87Rb/86","Low_ave","143/144","Low_ave","147/144"
7310      DATA "Sub_ave","87Sr/86"
7320      DATA "Sub_ave","87Rb/86","Sub_ave","143/144","Sub_ave","147/144"
7330      RESTORE 7220
7340      FOR I=1 TO 23
7350        S1=9*I-8
7360        S2=S1+8
7370        READ C1$,C2$
7380        Cname1$[S1,S2]=C1$
7390        Cname2$[S1,S2]=C2$
7400      NEXT I
7410      Heading$[1,207]=""
7420      File$=File1$&"AveSrA"&Drive_string$
7430      Heading$[1,207]="Average Sr and Nd isotopic ratios for file "&File1$
7440   Writefile1(Ratio(*),Heading$,File$,Cname1$,Cname2$,50,Go,207,23)
7450      GOTO Ratios
7460   !
7470 Oro_pb: Clear
```

```
7480     PRINT TABXY(10,10);FNH$("Please be patient, I'm thinking!!")
7490     H=1
7500     FOR K=1 TO Cycles STEP Step
7510       T_ratio=T_start-((K-1)*Td)
7520         IF T_ratio<0 THEN T_ratio=0
7530       Ratio(H,1)=DROUND(T_ratio,2)
7540       IF Vert(1,K)=0 THEN GOTO 7590
7550       Ratio(H,2)=Vert(2,K)/Vert(1,K)
7560       Ratio(H,3)=Vert(3,K)/Vert(1,K)
7570       Ratio(H,4)=Vert(4,K)/Vert(1,K)
7580       Ratio(H,5)=Vert(6,K)/Vert(1,K)
7590       IF Vert(6,K)=0 THEN GOTO 8230
7600       Ratio(H,6)=Vert(5,K)/Vert(6,K)
7610       IF L_hz(1,K)=0 THEN GOTO 7660
7620       Ratio(H,7)=L_hz(2,K)/L_hz(1,K)
7630       Ratio(H,8)=L_hz(3,K)/L_hz(1,K)
7640       Ratio(H,9)=L_hz(4,K)/L_hz(1,K)
7650       Ratio(H,10)=L_hz(6,K)/L_hz(1,K)
7660       IF L_hz(6,K)=0 THEN GOTO 8310
7670       Ratio(H,11)=L_hz(5,K)/L_hz(6,K)
7680       IF W_oro(1,K,2)=0 THEN GOTO 7730  ! Ratios in Worogene after communic
ation at time k
7690       Ratio(H,12)=W_oro(2,K,2)/W_oro(1,K,2)
7700       Ratio(H,13)=W_oro(3,K,2)/W_oro(1,K,2)
7710       Ratio(H,14)=W_oro(4,K,2)/W_oro(1,K,2)
7720       Ratio(H,15)=W_oro(6,K,2)/W_oro(1,K,2)
7730       IF W_oro(6,K,2)=0 THEN GOTO 8390
7740       Ratio(H,16)=W_oro(5,K,2)/W_oro(6,K,2)
7750       IF P_oro(1,K,1)=0 THEN GOTO 7800   ! Ratios in Porogene after communi
cation at time k
7760       Ratio(H,17)=P_oro(2,K,1)/P_oro(1,K,1)
7770       Ratio(H,18)=P_oro(3,K,1)/P_oro(1,K,1)
7780       Ratio(H,19)=P_oro(4,K,1)/P_oro(1,K,1)
7790       Ratio(H,20)=P_oro(6,K,1)/P_oro(1,K,1)
7800       IF P_oro(6,K,1)=0 THEN GOTO 8470
7810       Ratio(H,21)=P_oro(5,K,1)/P_oro(6,K,1)
7820       IF D_oro(1,K,2)=0 THEN GOTO 7870   ! Ratios in Dorogene after communi
cation at time k
7830       Ratio(H,22)=D_oro(2,K,2)/D_oro(1,K,2)
7840       Ratio(H,23)=D_oro(3,K,2)/D_oro(1,K,2)
7850       Ratio(H,24)=D_oro(4,K,2)/D_oro(1,K,2)
7860       Ratio(H,25)=D_oro(6,K,2)/D_oro(1,K,2)
7870       IF D_oro(6,K,2)=0 THEN GOTO 8550
7880       Ratio(H,26)=D_oro(5,K,2)/D_oro(6,K,2)
7890       IF Total_oro(1,K,1)=0 THEN GOTO 7940
7900       Ratio(H,27)=Total_oro(2,K,1)/Total_oro(1,K,1)
7910       Ratio(H,28)=Total_oro(3,K,1)/Total_oro(1,K,1)
7920       Ratio(H,29)=Total_oro(4,K,1)/Total_oro(1,K,1)
7930       Ratio(H,30)=Total_oro(6,K,1)/Total_oro(1,K,1)
7940       IF Total_oro(6,K,1)=0 THEN GOTO 8630
7950       Ratio(H,31)=Total_oro(5,K,1)/Total_oro(6,K,1)
7960       IF Up_seg(1,K)=0 THEN GOTO 8000
7970       Ratio(H,32)=Up_seg(2,K)/Up_seg(1,K)
7980       Ratio(H,33)=Up_seg(3,K)/Up_seg(1,K)
7990       Ratio(H,34)=Up_seg(4,K)/Up_seg(1,K)
```

```
8000        IF Up_seg(1,K)=0 THEN 8020
8010        Ratio(H,35)=Up_seg(6,K)/Up_seg(1,K)  ! Seg gives value of particular
segment
8020        IF Up_seg(6,K)=0 THEN GOTO 8730
8030        Ratio(H,36)=Up_seg(5,K)/Up_seg(6,K)
8040        IF Low_seg(1,K)=0 THEN GOTO 8080
8050        Ratio(H,37)=Low_seg(2,K)/Low_seg(1,K)
8060        Ratio(H,38)=Low_seg(3,K)/Low_seg(1,K)
8070        Ratio(H,39)=Low_seg(4,K)/Low_seg(1,K)
8080        IF Low_seg(1,K)=0 THEN 8100
8090        Ratio(H,40)=Low_seg(6,K)/Low_seg(1,K)
8100        IF Low_seg(6,K)=0 THEN GOTO 8820
8110        Ratio(H,41)=Low_seg(5,K)/Low_seg(6,K)
8120        IF Sub_seg(1,K)=0 THEN GOTO 8160
8130        Ratio(H,42)=Sub_seg(2,K)/Sub_seg(1,K)
8140        Ratio(H,43)=Sub_seg(3,K)/Sub_seg(1,K)
8150        Ratio(H,44)=Sub_seg(4,K)/Sub_seg(1,K)
8160        IF Sub_seg(1,K)=0 THEN 8180
8170        Ratio(H,45)=Sub_seg(6,K)/Sub_seg(1,K)
8180        IF Sub_seg(6,K)=0 THEN GOTO 8910
8190        Ratio(H,46)=Sub_seg(5,K)/Sub_seg(6,K)
8200      IF K=1 AND Step=4 THEN K=2
8210      H=H+1
8220      NEXT K
8230      Go=H
8240 Oro_pb_file: !
8250      Clear
8260      PRINT TABXY(1,10)
8270      PRINT TAB(Center);"Writing file for orogenic Pb isotopic ratios for:  "
&FNH$(File1$)
8280      Cname1$[1,432]=""
8290      Cname2$[1,432]=""
8300      DATA " "," "," "," "
8310      DATA " ","TIME"
8320      DATA "Vert","206/204","Vert","207/204","Vert","208/204"
8330      DATA "Vert","238U/204","Vert","232/238U"
8340      DATA "LHoriz","206/204","LHoriz","207/204","LHoriz","208/204"
8350      DATA "LHoriz","238U/204","LHoriz","232/238U"
8360      DATA "Worogene","206/204","Worogene","207/204","Worogene","208/204"
8370      DATA "Worogene","238U/204","Worogene","232/238U"
8380      DATA "Porogene","206/204","Porogene","207/204","Porogene","208/204"
8390      DATA "Porogene","238U/204","Porogene","232/238U"
8400      DATA "Dorogene","206/204","Dorogene","207/204","Dorogene","208/204"
8410      DATA "Dorogene","238U/204","Dorogene","232/238U"
8420      DATA "Total_or","206/204","Total_or","207/204","Total_or","208/204"
8430      DATA "Total_or","238U/204","Total_or","232/238U"
8440      DATA "Up_seg","206/204","Up_seg","207/204","Up_seg","208/204"
8450      DATA "Up_seg","238U/204","Up_seg","232/238U"
8460      DATA "Low_seg","206/204","Low_seg","207/204","Low_seg","208/204"
8470      DATA "Low_seg","238U/204","Low_seg","232/238U"
8480      DATA "Sub_seg","206/204","Sub_seg","207/204","Sub_seg","208/204"
8490      DATA "Sub_seg","238U/204","Sub_seg","232/238U"
8500      RESTORE 8300
8510      FOR I=1 TO 48
8520        S1=9*I-8
```

```
8530        S2=S1+8
8540        READ C1$,C2$
8550        Cname1$[S1,S2]=C1$
8560        Cname2$[S1,S2]=C2$
8570      NEXT I
8580      Heading$[1,432]=""
8590      File$=File1$&"OroPbA"&Drive_string$
8600      Heading$[1,432]="Orogenic Pb isotopic ratios for file "&File1$
8610   Writefile1(Ratio(*),Heading$,File$,Cname1$,Cname2$,85,Go,432,48)
8620      GOTO Ratios
8630 !
8640 Oro_sr: !
8650      Clear
8660      PRINT TABXY(1,10);FNH$("Please be patient, I'm thinking!!")
8670      H=1
8680      FOR K=1 TO Cycles STEP Step
8690        T_ratio=T_start-((K-1)*Td)
8700          IF T_ratio<0 THEN T_ratio=0
8710        Ratio(H,1)=DROUND(T_ratio,2)
8720        IF Vert(7,K)=0 THEN GOTO 8750
8730        Ratio(H,2)=Vert(8,K)/Vert(7,K)
8740        Ratio(H,3)=Vert(9,K)/Vert(7,K)
8750        IF Vert(10,K)=0 THEN GOTO 9540
8760        Ratio(H,4)=Vert(11,K)/Vert(10,K)
8770        Ratio(H,5)=Vert(12,K)/Vert(10,K)
8780        IF L_hz(7,K)=0 THEN GOTO 8810
8790        Ratio(H,6)=L_hz(8,K)/L_hz(7,K)
8800        Ratio(H,7)=L_hz(9,K)/L_hz(7,K)
8810        IF L_hz(10,K)=0 THEN GOTO 9610
8820        Ratio(H,8)=L_hz(11,K)/L_hz(10,K)
8830        Ratio(H,9)=L_hz(12,K)/L_hz(10,K)
8840        IF W_oro(7,K,2)=0 THEN GOTO 8870
8850        Ratio(H,10)=W_oro(8,K,2)/W_oro(7,K,2)
8860        Ratio(H,11)=W_oro(9,K,2)/W_oro(7,K,2)
8870        IF W_oro(10,K,2)=0 THEN GOTO 9680
8880        Ratio(H,12)=W_oro(11,K,2)/W_oro(10,K,2)
8890        Ratio(H,13)=W_oro(12,K,2)/W_oro(10,K,2)
8900        IF P_oro(7,K,1)=0 THEN GOTO 8930
8910        Ratio(H,14)=P_oro(8,K,1)/P_oro(7,K,1)
8920        Ratio(H,15)=P_oro(9,K,1)/P_oro(7,K,1)
8930        IF P_oro(10,K,1)=0 THEN GOTO 9750
8940        Ratio(H,16)=P_oro(11,K,1)/P_oro(10,K,1)
8950        Ratio(H,17)=P_oro(12,K,1)/P_oro(10,K,1)
8960        IF D_oro(7,K,2)=0 THEN GOTO 8990
8970        Ratio(H,18)=D_oro(8,K,2)/D_oro(7,K,2)
8980        Ratio(H,19)=D_oro(9,K,2)/D_oro(7,K,2)
8990        IF D_oro(10,K,2)=0 THEN GOTO 9820
9000        Ratio(H,20)=D_oro(11,K,2)/D_oro(10,K,2)
9010        Ratio(H,21)=D_oro(12,K,2)/D_oro(10,K,2)
9020        IF Total_oro(7,K,1)=0 THEN GOTO 9050
9030        Ratio(H,22)=Total_oro(8,K,1)/Total_oro(7,K,1)
9040        Ratio(H,23)=Total_oro(9,K,1)/Total_oro(7,K,1)
9050        IF Total_oro(10,K,1)=0 THEN GOTO 9890
9060        Ratio(H,24)=Total_oro(11,K,1)/Total_oro(10,K,1)
9070        Ratio(H,25)=Total_oro(12,K,1)/Total_oro(10,K,1)
```

```
9080        IF Up_seg(7,K)=0 THEN GOTO 9100
9090        Ratio(H,26)=Up_seg(8,K)/Up_seg(7,K)
9100        IF Up_seg(7,K)=0 THEN 9120
9110        Ratio(H,27)=Up_seg(9,K)/Up_seg(7,K)
9120        IF Up_seg(10,K)=0 THEN GOTO 9980
9130        Ratio(H,28)=Up_seg(11,K)/Up_seg(10,K)
9140        IF Up_seg(10,K)=0 THEN 9980
9150        Ratio(H,29)=Up_seg(12,K)/Up_seg(10,K)
9160        IF Low_seg(7,K)=0 THEN GOTO 9180
9170        Ratio(H,30)=Low_seg(8,K)/Low_seg(7,K)
9180        IF Low_seg(7,K)=0 THEN 9200
9190        Ratio(H,31)=Low_seg(9,K)/Low_seg(7,K)
9200        IF Low_seg(10,K)=0 THEN GOTO 10070
9210        Ratio(H,32)=Low_seg(11,K)/Low_seg(10,K)
9220        IF Low_seg(10,K)=0 THEN 10070
9230        Ratio(H,33)=Low_seg(12,K)/Low_seg(10,K)
9240        IF Sub_seg(7,K)=0 THEN GOTO 9260
9250        Ratio(H,34)=Sub_seg(8,K)/Sub_seg(7,K)
9260        IF Sub_seg(7,K)=0 THEN 9280
9270        Ratio(H,35)=Sub_seg(9,K)/Sub_seg(7,K)
9280        IF Sub_seg(10,K)=0 THEN GOTO 10160
9290        Ratio(H,36)=Sub_seg(11,K)/Sub_seg(10,K)
9300        IF Sub_seg(10,K)=0 THEN 10160
9310        Ratio(H,37)=Sub_seg(12,K)/Sub_seg(10,K)
9320      IF K=1 AND Step=4 THEN K=2
9330      H=H+1
9340      NEXT K
9350      Go=H
9360 Oro_sr_file: !
9370      Clear
9380      PRINT TABXY(1,10)
9390      PRINT TAB(Center);"Writing file for orogenic Sr and Nd isotopic ratios
for:  "&FNH$(File1$)
9400      Cname1$[1,351]=""
9410      Cname2$[1,351]=""
9420      DATA " "," "," "," "
9430      DATA " ","TIME"
9440      DATA "Vert","87Sr/86"
9450      DATA "Vert","87Rb/86","Vert","143/144","Vert","147/144"
9460      DATA "LHoriz","87Sr/86"
9470      DATA "LHoriz","87Rb/86","LHoriz","143/144","LHoriz","147/144"
9480      DATA "Worogene","87Sr/86"
9490      DATA "Worogene","87Rb/86","Worogene","143/144","Worogene","147/144"
9500      DATA "Porogene","87Sr/86"
9510      DATA "Porogene","87Rb/86","Porogene","143/144","Porogene","147/144"
9520      DATA "Dorogene","87Sr/86"
9530      DATA "Dorogene","87Rb/86","Dorogene","143/144","Dorogene","147/144"
9540      DATA "Total_or","87Sr/86"
9550      DATA "Total_or","87Rb/86","Total_or","143/144","Total_or","147/144"
9560      DATA "Up_seg","87Sr/86"
9570      DATA "Up_seg","87Rb/86","Up_seg","143/144","Up_seg","147/144"
9580      DATA "Low_seg","87Sr/86"
9590      DATA "Low_seg","87Rb/86","Low_seg","143/144","Low_seg","147/144"
9600      DATA "Sub_seg","87Sr/86"
9610      DATA "Sub_seg","87Rb/86","Sub_seg","143/144","Sub_seg","147/144"
```

```
9620     RESTORE 9420
9630     FOR I=1 TO 39
9640       S1=9*I-8
9650       S2=S1+8
9660       READ C1$,C2$
9670       Cname1$[S1,S2]=C1$
9680       Cname2$[S1,S2]=C2$
9690     NEXT I
9700     Heading$[1,351]=""
9710     File$=File1$&"OroSrA"&Drive_string$
9720     Heading$[1,351]="Orogenic Sr and Nd isotopic ratios for file "&File1$
9730   Writefile1(Ratio(*),Heading$,File$,Cname1$,Cname2$,75,Go,351,39)
9740     GOTO Ratios
9750   !
9760 Abundances: ! This subroutine for reservoir abundances
9770     MAT Abund= (0)
9780     Clear
9790     PRINT TABXY(1,10);FNH$("Please be patient, I'm thinking!!")
9800     Lo=1
9810     FOR K=1 TO Cycles STEP Step
9820       T_conc=T_start-((K-1)*Td)
9830         IF T_conc<0 THEN T_conc=0
9840       Abund(Lo,0,1)=DROUND(T_conc,2)
9850       FOR H=Gg TO Hh
9860         Abund(Lo,H,1)=Abund(Lo,0,1)
9870         Abund(Lo,H,2)=Vert(H,K)
9880         Abund(Lo,H,3)=Hz(H,K)
9890         Abund(Lo,H,4)=Sub_re(H,K)
9900         Abund(Lo,H,5)=Sub_mant(H,K)
9910         Abund(Lo,H,6)=Outboard(H,K)
9920         Abund(Lo,H,7)=Inboard(H,K)
9930         Abund(Lo,H,8)=Mantle(H,K)
9940         Abund(Lo,H,9)=Mor(H,K)
9950         Abund(Lo,H,10)=Rmor(H,K)
9960         Abund(Lo,H,11)=P_oro(H,K,0)
9970         Abund(Lo,H,12)=D_oro(H,K,0)
9980         Abund(Lo,H,13)=W_oro(H,K,0)
9990         Abund(Lo,H,14)=P_oro(H,K,1)
10000        Abund(Lo,H,15)=D_oro(H,K,2)
10010        Abund(Lo,H,16)=W_oro(H,K,2)
10020        Abund(Lo,H,17)=Total_oro(H,K,1)
10030        Abund(Lo,H,18)=Gateb1(H,K)
10040        Abund(Lo,H,19)=Gateb2(H,K)
10050        Abund(Lo,H,20)=Gateb3(H,K)
10060        Abund(Lo,H,21)=Sum_up(H,K)
10070        Abund(Lo,H,22)=Sum_low(H,K)
10080        Abund(Lo,H,23)=Sum_sub(H,K)
10090        Abund(Lo,H,24)=D_mant(H,K)
10100      NEXT H
10110    IF K=1 AND Step=4 THEN K=2
10120    Lo=Lo+1
10130    NEXT K
10140    Go=Lo
10150 Abund_file: Clear
10160    Cname1$[1,234]=""
```

```
10170     Cname2$[1,234]=""
10180     DATA " "," ","TIME","Vertical","Horizont","Sub_re","Sub_mant","Outbaord
","Inboard","Mantle","MORB","Rmor"
10190     DATA "Prox_b","Dist_b","Worogene_b","Prox_a","Dist_a","Worogene_a","Tot
al_or","Gateb1","Gateb2","Gateb3"
10200     DATA "Sum_up","Sum_low","Sum_sub","Dmant"
10210     RESTORE 10180
10220     FOR I=1 TO 26
10230       S1=9*I-8
10240       S2=S1+8
10250       READ C2$
10260       Cname2$[S1,S2]=C2$
10270     NEXT I
10280     DATA "204Pb","206Pb","207Pb","208Pb","232Th","_238U","_86Sr","_87Sr","_
87Rb","144Nd","143Nd","147Sm"
10290     RESTORE 10280
10300     FOR H=1 TO 12
10310       READ Add_file$(H)
10320     NEXT H
10330     FOR H=1 TO Hh
10340       Rowlength=234
10350       File$=File1$&Add_file$(H)&"A"&Drive_string$
10360       Heading$[1,Rowlength]=""
10370       Heading$[1,Rowlength]="Reservoir Abundances (E+15 moles) and masses (
E+24 grams) "&File1$&Add_file$(H)
10380       Clear
10390       PRINT TABXY(1,10)
10400       PRINT TAB(Center);"Writing file for abundances in each reservoir"
10410       PRINT TAB(Center);"for file:  "&FNH$(Add_file$(H))
10420 Writefile2(Abund(*),Heading$,File$,Cname1$,Cname2$,50,Go,234,26,H)
10430     NEXT H
10440     GOTO Options
10450      !
10460 Concentrations:! This subroutine for reservoir abundances
10470     MAT Vc_cc= (0)
10480     MAT Conc= (0)
10490     Clear
10500     PRINT TABXY(1,10);FNH$("Please be patient, I'm thinking!!")
10510     Lo=1
10520     FOR K=1 TO Cycles STEP Step
10530       T_conc=T_start-((K-1)*Td)
10540         IF T_conc<.1 THEN T_conc=0
10550       FOR Element=1 TO 7
10560         Conc(Lo,Element,1)=DROUND(T_conc,2)
10570       NEXT Element
10580       FOR H=0 TO Hh
10590         Vc_cc(Lo,H,2)=Vert(H,K)
10600         Vc_cc(Lo,H,3)=Hz(H,K)
10610         Vc_cc(Lo,H,4)=Sub_re(H,K)
10620         Vc_cc(Lo,H,5)=Sub_mant(H,K)
10630         Vc_cc(Lo,H,6)=Outboard(H,K)
10640         Vc_cc(Lo,H,7)=Inboard(H,K)
10650         Vc_cc(Lo,H,8)=Mantle(H,K)  ! Moles in Mantle before orogeny
10660         Vc_cc(Lo,H,9)=Mor(H,K)
10670         Vc_cc(Lo,H,10)=Rmor(H,K)
```

```
10680         Vc_cc(Lo,H,11)=P_oro(H,K,0)  ! Moles in Porogene before communicati
on (Horiz_low + Inboard)
10690         Vc_cc(Lo,H,12)=D_oro(H,K,0)  ! Moles in Dorogene before communicati
on (Mor + Outboard)
10700         Vc_cc(Lo,H,13)=W_oro(H,K,0)  ! Moles in Worogene before communicati
on (Sub_re + Sub_mant)
10710         Vc_cc(Lo,H,14)=P_oro(H,K,1)  ! Moles in Porogene after communicatio
n
10720         Vc_cc(Lo,H,15)=D_oro(H,K,2)  ! Moles in Dorogene after communicatio
n
10730         Vc_cc(Lo,H,16)=W_oro(H,K,2)  ! Moles in Worogene after communicatio
n
10740         Vc_cc(Lo,H,17)=Total_oro(H,K,1)  ! Moles in total orogene after com
munication
10750         Vc_cc(Lo,H,18)=Gateb1(H,K)
10760         Vc_cc(Lo,H,19)=Gateb2(H,K)
10770         Vc_cc(Lo,H,20)=Gateb3(H,K)
10780         Vc_cc(Lo,H,21)=Sum_up(H,K)
10790         Vc_cc(Lo,H,22)=Sum_low(H,K)
10800         Vc_cc(Lo,H,23)=Sum_sub(H,K)
10810         Vc_cc(Lo,H,24)=D_mant(H,K)  ! Moles in Mantle after orogeny
10820       NEXT H
10830     IF K=1 AND Step=4 THEN K=2
10840     Lo=Lo+1
10850     NEXT K
10860     Go=Lo
10870 Ppm_all:!
10880     FOR Lo=1 TO Go
10890      FOR I=2 TO 24
10900        Conc(Lo,1,I)=FNPpm_pb(Vc_cc(Lo,0,I),Vc_cc(Lo,1,I),Vc_cc(Lo,2,I),Vc_c
c(Lo,3,I),Vc_cc(Lo,4,I))
10910         Conc(Lo,2,I)=FNPpm_others(Vc_cc(Lo,0,I),Vc_cc(Lo,5,I),Iso_abund(2),E
l_mw(2))
10920         Conc(Lo,3,I)=FNPpm_others(Vc_cc(Lo,0,I),Vc_cc(Lo,6,I),Iso_abund(3),E
l_mw(3))
10930         Conc(Lo,4,I)=FNPpm_others(Vc_cc(Lo,0,I),Vc_cc(Lo,7,I),Iso_abund(4),E
l_mw(4))
10940         Conc(Lo,5,I)=FNPpm_others(Vc_cc(Lo,0,I),Vc_cc(Lo,9,I),Iso_abund(5),E
l_mw(5))
10950         Conc(Lo,6,I)=FNPpm_others(Vc_cc(Lo,0,I),Vc_cc(Lo,10,I),Iso_abund(6),
El_mw(6))
10960         Conc(Lo,7,I)=FNPpm_others(Vc_cc(Lo,0,I),Vc_cc(Lo,12,I),Iso_abund(7),
El_mw(7))
10970      NEXT I
10980     NEXT Lo
10990 Ppm_end_all:  !
11000     DATA "Pb","Th","_U","Sr","Rb","Nd","Sm"
11010     RESTORE 11000
11020     FOR H=1 TO 7
11030       READ Elem$(H)
11040     NEXT H
11050 Conc_file: Clear
11060     Cname1$[1,234]=""
11070     Cname2$[1,234]=""
11080     DATA " "," ","TIME","Vertical","Horizont","Sub_re","Sub_mant","Outboard
```

```
","Inboard","Mantle","Mor","Rmor","Prox_b","Dist_b","Worogene_b"
11090     DATA "Prox_a", "Dist_a","Worogene_a","Tot_oro_a","Gateb1","Gateb2","Gat
eb3","Sum_up","Sum_low","Sum_sub","D_mant"
11100     RESTORE 11080
11110     FOR I=1 TO 26
11120       S1=9*I-8
11130       S2=S1+8
11140       READ C2$
11150       Cname2$[S1,S2]=C2$
11160     NEXT I
11170     FOR H=1 TO 7
11180       Rowlength=234
11190       File$=File1$&"con"&Elem$(H)&"A"&Drive_string$
11200       Heading$[1,Rowlength]=""
11210       Heading$[1,Rowlength]="Concentrations (ppm) "&File1$&Elem$(H)
11220       Clear
11230       PRINT TABXY(1,10)
11240       PRINT TAB(Center);"Writing file for concentrations for "&FNH$(Elem$(H
))
11250 Writefile2(Conc(*),Heading$,File$,Cname1$,Cname2$,50,Go,234,26,H)
11260     NEXT H
11270     GOTO Options
11280      !
11290 Masses:! This subroutine for MASSES
11300     MAT Mass= (0)
11310     Clear
11320     PRINT TABXY(1,10);FNH$("Please be patient, I'm thinking!!")
11330     Lo=1
11340     FOR K=1 TO Cycles STEP Step
11350       T_conc=T_start-((K-1)*Td)
11360         IF T_conc<0 THEN T_conc=0
11370       Mass(Lo,1)=DROUND(T_conc,2)
11380         Mass(Lo,2)=Vert(0,K)
11390         Mass(Lo,3)=Hz(0,K)
11400         Mass(Lo,4)=Sub_re(0,K)
11410         Mass(Lo,5)=(Sub_mant(0,K)/(Mantle(0,K)-Mor(0,K)))*10   !okay  10 TI
MES  mass fraction of m->w
11420         Mass(Lo,6)=Outboard(0,K)
11430         Mass(Lo,7)=Inboard(0,K)
11440         Mass(Lo,8)=M(0,1,K,0)  ! Mass in Mantle before orogeny
11450         Mass(Lo,9)=Mor(0,K)
11460         Mass(Lo,10)=Rmor(0,K)
11470         Mass(Lo,11)=P_oro(0,K,0)  ! Mass in Porogene before communication
11480         Mass(Lo,12)=D_oro(0,K,0)  ! Mass in Dorogene before communication
11490         Mass(Lo,13)=W_oro(0,K,0)  ! Mass in Worogene before communication
11500         Mass(Lo,14)=P_oro(0,K,1)  ! Mass in Porogene after communication
11510         Mass(Lo,15)=D_oro(0,K,2)  ! Mass in Dorogene after communication
11520         Mass(Lo,16)=W_oro(0,K,2)  ! Mass in Worogene after communication
11530         Mass(Lo,17)=Total_oro(0,K,1)  ! Mass in total orogene after communi
cation
11540         Mass(Lo,18)=Gateb1(0,K)/D_oro(0,K,1)  ! Mass fraction of d->p
11550         Mass(Lo,19)=(Gateb2(0,K)/D_oro(0,K,0))*100  ! 100 x Mass fraction o
f d->w
11560         Mass(Lo,20)=Gateb3(0,K)/W_oro(0,K,1)  ! Mass fraction of w->p
11570         Mass(Lo,21)=Sum_up(0,K)
```

```
11580         Mass(Lo,22)=Sum_low(0,K)
11590         Mass(Lo,23)=Sum_sub(0,K)
11600         Mass(Lo,24)=D_mant(0,K)
11610         Mass(Lo,25)=FNArea(Sum_low(0,K),35.0,3.0)  !CONTINENTAL AREA KM^2
11620         Mass(Lo,26)=FNArea(Sum_low_c(0,K),35.0,3.0) !CREATE CONT AREA
11630         Mass(Lo,27)=Mass(Lo,22)-Mass(Lo,21)          !AREA DESTROYED
11640        IF L(K)=0 THEN GOTO 11670
11650         Mass(Lo,28)=(1-(L_hz(0,K)/L(K)))*100  ! % continental growth
11660         Mass(Lo,29)=100*((Gateb1(0,K)+Gateb3(0,K))/(U(K)+L(K)))  ! % unfill
ed Porogene
11670         Mass(Lo,30)=100*(Sub_mant(0,K)/S(K))  ! % of unfilled Worogene
11680      F_area=FNArea(Sum_low(0,46),35.0,3.0)  ! FINAL AREA IN km^2
11690         Mass(Lo,31)=(FNArea(Sum_low(0,K),35.0,3.0)/F_area)*100  ! % CONTINE
NTAL AREA NORMALIZED TO TODAY (average for total Lcrust at each orogeny
11700     IF K=1 AND Step=4 THEN K=2
11710     Lo=Lo+1
11720     NEXT K
11730     Go=Lo
11740 Masses_file: Clear
11750     PRINT TABXY(1,10)
11760     PRINT TAB(Center);"Writing for masses for "&FNH$(File1$)
11770     Cname1$[1,297]=""
11780     Cname2$[1,297]=""
11790     DATA " "," ","TIME","Vertical","Horizont","Sub_re","m->w_MF","Outboard"
,"Inboard","Mantle","MORB","Rmor"
11800     DATA "Prox_b","Dist_b","Worogene_b","Prox_a","Dist_a","Worogene_a","Tot
al_or_a","d->p_MF","d->w_MF","w->p_MF"
11810     DATA "Sum_up","Sum_low","Sum_sub","Dmant"
11820     DATA "Cont area","CreatArea","DestrArea","%growth","%Sm_unfil","%Sub_un
fil","%ContArea"
11830     RESTORE 11790
11840     FOR I=1 TO 33
11850       S1=9*I-8
11860       S2=S1+8
11870       READ C2$
11880       Cname2$[S1,S2]=C2$
11890     NEXT I
11900       Rowlength=297
11910       File$=File1$&"Mass_"&"A"&Drive_string$
11920       Heading$[1,Rowlength]=""
11930       Heading$[1,Rowlength]="Masses (E+24 grams) for file "&File1$
11940 Writefile1(Mass(*),Heading$,File$,Cname1$,Cname2$,65,Go,297,33)
11950     GOTO Options
11960 !
11970 Ending_values:IF F$[7,9]="End" THEN GOTO Keys
11980 Keys:!
11990       Clear
12000       PRINT TABXY(1,10)
12010       PRINT TAB(Center);"KEY     DESCRIPTION"
12020       PRINT
12030       PRINT TAB(Center);"K0      Print table of final abund, conc., and rat
ios"
12040       PRINT TAB(Center);"k4      Create file for ISOPLOT for crustal segmen
ts"
12050       PRINT TAB(Center);"         of upper, lower, and subcrust and their r
```

```
espective"
12060       PRINT TAB(Center);"          mass, conc.'s, and ratios, as they would
be seen today."
12070       PRINT
12080       PRINT TAB(Center);"k9       Go back to options screen."
12090       OFF KEY
12100       ON KEY 0 LABEL "Print table" GOTO Convert_values
12110       ON KEY 4 LABEL "Segments" GOTO Segments
12120       ON KEY 9 LABEL "ESCAPE" GOTO Options
12130       GOTO 12130
12140 Convert_values:K=Cycles
12150     MAT End= (0)
12160     End(1,4)=DROUND(Sum_up(0,K),4)
12170     End(1,5)=DROUND(Sum_low(0,K),4)
12180     End(1,6)=DROUND(Sum_sub(0,K),4)
12190     End(1,3)=DROUND(D_mant(0,K),5)
12200     Add=End(1,3)+End(1,4)+End(1,5)+End(1,6)
12210     End(1,7)=DROUND(Add,6)
12220     FOR H=Gg TO Hh
12230       End(H+1,4)=DROUND(Sum_up(H,K),3)
12240       IF Sum_low(0,K)=0 THEN 12260
12250       End(H+1,5)=DROUND(Sum_low(H,K),3)
12260       IF Sum_sub(0,K)=0 THEN 12270
12270       End(H+1,6)=DROUND(Sum_sub(H,K),3)
12280       End(H+1,3)=DROUND(D_mant(H,K),3)
12290       Add=Sum_up(H,K)+Sum_low(H,K)+Sum_sub(H,K)+D_mant(H,K)
12300       End(H+1,7)=DROUND(Add,6)
12310     NEXT H
12320 Ppm: !
12330      FOR I=3 TO 6
12340        Ppm(1,I)=FNPpm_pb(End(1,I),End(2,I),End(3,I),End(4,I),End(5,I))
12350        Ppm(2,I)=FNPpm_others(End(1,I),End(6,I),Iso_abund(2),El_mw(2))
12360        Ppm(3,I)=FNPpm_others(End(1,I),End(7,I),Iso_abund(3),El_mw(3))
12370        Ppm(4,I)=FNPpm_others(End(1,I),End(8,I),Iso_abund(4),El_mw(4))
12380        Ppm(5,I)=FNPpm_others(End(1,I),End(10,I),Iso_abund(5),El_mw(5))
12390        Ppm(6,I)=FNPpm_others(End(1,I),End(11,I),Iso_abund(6),El_mw(6))
12400        Ppm(7,I)=FNPpm_others(End(1,I),End(13,I),Iso_abund(7),El_mw(7))
12410      NEXT I
12420      FOR H=Yy TO Zz
12430       FOR I=3 TO 6
12440        Ppm(H,I)=DROUND(Ppm(H,I),4)
12450       NEXT I
12460     NEXT H
12470 Ppm_end:  !
12480     IF Mantle(1,K)=0 THEN GOTO 12530
12490     End(14,3)=DROUND(Mantle(2,K)/Mantle(1,K),5)
12500     End(15,3)=DROUND(Mantle(3,K)/Mantle(1,K),5)
12510     End(16,3)=DROUND(Mantle(4,K)/Mantle(1,K),5)
12520     End(17,3)=DROUND(Mantle(6,K)/Mantle(1,K),5)
12530     IF Mantle(6,K)=0 THEN GOTO 12570
12540     End(18,3)=DROUND(Mantle(5,K)/Mantle(6,K),5)
12550     End(19,3)=DROUND(Mantle(5,K)/Mantle(1,K),5)
12560     IF Hh=6 THEN GOTO 13800
12570     IF Mantle(7,K)=0 THEN GOTO 12600
12580     End(20,3)=DROUND(Mantle(8,K)/Mantle(7,K),5)
```

```
12590    End(21,3)=DROUND(Mantle(9,K)/Mantle(7,K),4)
12600    IF Mantle(10,K)=0 THEN GOTO 13800
12610    End(22,3)=DROUND(Mantle(11,K)/Mantle(10,K),5)
12620    End(23,3)=DROUND(Mantle(12,K)/Mantle(10,K),4)
12630    IF Vert(1,K)=0 THEN GOTO 12680
12640    End(14,4)=DROUND(Vert(2,K)/Vert(1,K),5)
12650    End(15,4)=DROUND(Vert(3,K)/Vert(1,K),5)
12660    End(16,4)=DROUND(Vert(4,K)/Vert(1,K),5)
12670    End(17,4)=DROUND(Vert(6,K)/Vert(1,K),5)
12680    IF Vert(6,K)=0 THEN GOTO 12720
12690    End(18,4)=DROUND(Vert(5,K)/Vert(6,K),5)
12700    End(19,4)=DROUND(Vert(5,K)/Vert(1,K),5)
12710    IF Hh=6 THEN GOTO 13960
12720    IF Vert(7,K)=0 THEN GOTO 12750
12730    End(20,4)=DROUND(Vert(8,K)/Vert(7,K),5)
12740    End(21,4)=DROUND(Vert(9,K)/Vert(7,K),4)
12750    IF Vert(10,K)=0 THEN GOTO 13960
12760    End(22,4)=DROUND(Vert(11,K)/Vert(10,K),5)
12770    End(23,4)=DROUND(Vert(12,K)/Vert(10,K),4)
12780    IF Hz(1,K)=0 THEN GOTO 12830
12790    End(14,5)=DROUND(Hz(2,K)/Hz(1,K),5)
12800    End(15,5)=DROUND(Hz(3,K)/Hz(1,K),5)
12810    End(16,5)=DROUND(Hz(4,K)/Hz(1,K),5)
12820    End(17,5)=DROUND(Hz(6,K)/Hz(1,K),5)
12830    IF Hz(6,K)=0 THEN GOTO 12870
12840    End(18,5)=DROUND(Hz(5,K)/Hz(6,K),5)
12850    End(19,5)=DROUND(Hz(5,K)/Hz(1,K),5)
12860    IF Hh=6 THEN GOTO 14120
12870    IF Hz(7,K)=0 THEN GOTO 12900
12880    End(20,5)=DROUND(Hz(8,K)/Hz(7,K),5)
12890    End(21,5)=DROUND(Hz(9,K)/Hz(7,K),4)
12900    IF Hz(10,K)=0 THEN GOTO 14120
12910    End(22,5)=DROUND(Hz(11,K)/Hz(10,K),5)
12920    End(23,5)=DROUND(Hz(12,K)/Hz(10,K),4)
12930    IF P_oro(1,K,0)=0 THEN GOTO 12980
12940    End(14,6)=DROUND(P_oro(2,K,0)/P_oro(1,K,0),5)
12950    End(15,6)=DROUND(P_oro(3,K,0)/P_oro(1,K,0),5)
12960    End(16,6)=DROUND(P_oro(4,K,0)/P_oro(1,K,0),5)
12970    End(17,6)=DROUND(P_oro(6,K,0)/P_oro(1,K,0),5)
12980    IF P_oro(6,K,0)=0 THEN GOTO 13020
12990    End(18,6)=DROUND(P_oro(5,K,0)/P_oro(6,K,0),5)
13000    End(19,6)=DROUND(P_oro(5,K,0)/P_oro(1,K,0),5)
13010    IF Hh=6 THEN GOTO 14280
13020    IF P_oro(7,K,0)=0 THEN GOTO 13050
13030    End(20,6)=DROUND(P_oro(8,K,0)/P_oro(7,K,0),5)
13040    End(21,6)=DROUND(P_oro(9,K,0)/P_oro(7,K,0),4)
13050    IF P_oro(10,K,0)=0 THEN GOTO 14450
13060    End(22,6)=DROUND(P_oro(11,K,0)/P_oro(10,K,0),5)
13070    End(23,6)=DROUND(P_oro(12,K,0)/P_oro(10,K,0),4)
13080    IF D_oro(1,K,0)=0 THEN GOTO 13130
13090    End(14,7)=DROUND(D_oro(2,K,0)/D_oro(1,K,0),5)
13100    End(15,7)=DROUND(D_oro(3,K,0)/D_oro(1,K,0),5)
13110    End(16,7)=DROUND(D_oro(4,K,0)/D_oro(1,K,0),5)
13120    End(17,7)=DROUND(D_oro(6,K,0)/D_oro(1,K,0),5)
13130    IF D_oro(6,K,0)=0 THEN GOTO 13170
```

```
13140    End(18,7)=DROUND(D_oro(5,K,0)/D_oro(6,K,0),5)
13150    End(19,7)=DROUND(D_oro(5,K,0)/D_oro(1,K,0),5)
13160    IF Hh=6 THEN GOTO 14440
13170    IF D_oro(7,K,0)=0 THEN GOTO 13200
13180    End(20,7)=DROUND(D_oro(8,K,0)/D_oro(7,K,0),5)
13190    End(21,7)=DROUND(D_oro(9,K,0)/D_oro(7,K,0),4)
13200    IF D_oro(10,K,0)=0 THEN GOTO 14440
13210    End(22,7)=DROUND(D_oro(11,K,0)/D_oro(10,K,0),5)
13220    End(23,7)=DROUND(D_oro(12,K,0)/D_oro(10,K,0),4)
13230    IF P_oro(1,K,1)=0 THEN GOTO 13280
13240    End(14,8)=DROUND(P_oro(2,K,1)/P_oro(1,K,1),5)
13250    End(15,8)=DROUND(P_oro(3,K,1)/P_oro(1,K,1),5)
13260    End(16,8)=DROUND(P_oro(4,K,1)/P_oro(1,K,1),5)
13270    End(17,8)=DROUND(P_oro(6,K,1)/P_oro(1,K,1),5)
13280    IF P_oro(6,K,1)=0 THEN GOTO 13320
13290    End(18,8)=DROUND(P_oro(5,K,1)/P_oro(6,K,1),5)
13300    End(19,8)=DROUND(P_oro(5,K,1)/P_oro(1,K,1),5)
13310    IF Hh=6 THEN GOTO 14610
13320    IF P_oro(7,K,1)=0 THEN GOTO 13350
13330    End(20,8)=DROUND(P_oro(8,K,1)/P_oro(7,K,1),5)
13340    End(21,8)=DROUND(P_oro(9,K,1)/P_oro(7,K,1),4)
13350    IF P_oro(10,K,1)=0 THEN GOTO 14610
13360    End(22,8)=DROUND(P_oro(11,K,1)/P_oro(10,K,1),5)
13370    End(23,8)=DROUND(P_oro(12,K,1)/P_oro(10,K,1),4)
13380    IF D_oro(1,K,2)=0 THEN GOTO 13430
13390    End(14,9)=DROUND(D_oro(2,K,2)/D_oro(1,K,2),5)
13400    End(15,9)=DROUND(D_oro(3,K,2)/D_oro(1,K,2),5)
13410    End(16,9)=DROUND(D_oro(4,K,2)/D_oro(1,K,2),5)
13420    End(17,9)=DROUND(D_oro(6,K,2)/D_oro(1,K,2),5)
13430    IF D_oro(6,K,2)=0 THEN GOTO 13470
13440    End(18,9)=DROUND(D_oro(5,K,2)/D_oro(6,K,2),5)
13450    End(19,9)=DROUND(D_oro(5,K,2)/D_oro(1,K,2),5)
13460    IF Hh=6 THEN GOTO 14770
13470    IF D_oro(7,K,2)=0 THEN GOTO 13500
13480    End(20,9)=DROUND(D_oro(8,K,2)/D_oro(7,K,2),5)
13490    End(21,9)=DROUND(D_oro(9,K,2)/D_oro(7,K,2),4)
13500    IF D_oro(10,K,2)=0 THEN GOTO 14770
13510    End(22,9)=DROUND(D_oro(11,K,2)/D_oro(10,K,2),5)
13520    End(23,9)=DROUND(D_oro(12,K,2)/D_oro(10,K,2),4)
13530    IF Total_oro(1,K,0)=0 THEN GOTO 13580
13540    End(14,11)=DROUND(Total_oro(2,K,1)/Total_oro(1,K,1),5)
13550    End(15,11)=DROUND(Total_oro(3,K,1)/Total_oro(1,K,1),5)
13560    End(16,11)=DROUND(Total_oro(4,K,1)/Total_oro(1,K,1),5)
13570    End(17,11)=DROUND(Total_oro(6,K,1)/Total_oro(1,K,1),5)
13580    IF Total_oro(6,K,1)=0 THEN GOTO 13620
13590    End(18,11)=DROUND(Total_oro(5,K,1)/Total_oro(6,K,1),5)
13600    End(19,11)=DROUND(Total_oro(5,K,1)/Total_oro(1,K,1),5)
13610    IF Hh=6 THEN GOTO 14930
13620    IF Total_oro(7,K,1)=0 THEN GOTO 13650
13630    End(20,11)=DROUND(Total_oro(8,K,1)/Total_oro(7,K,1),5)
13640    End(21,11)=DROUND(Total_oro(9,K,1)/Total_oro(7,K,1),4)
13650    IF Total_oro(10,K,1)=0 THEN GOTO 14930
13660    End(22,11)=DROUND(Total_oro(11,K,1)/Total_oro(10,K,1),5)
13670    End(23,11)=DROUND(Total_oro(12,K,1)/Total_oro(10,K,1),4)
13680    IF W_oro(1,K,2)=0 THEN GOTO 13730
```

```
13690    End(14,10)=DROUND(W_oro(2,K,2)/W_oro(1,K,2),5)
13700    End(15,10)=DROUND(W_oro(3,K,2)/W_oro(1,K,2),5)
13710    End(16,10)=DROUND(W_oro(4,K,2)/W_oro(1,K,2),5)
13720    End(17,10)=DROUND(W_oro(6,K,2)/W_oro(1,K,2),5)
13730    IF W_oro(6,K,2)=0 THEN GOTO 13770
13740    End(18,10)=DROUND(W_oro(5,K,2)/W_oro(6,K,2),5)
13750    End(19,10)=DROUND(W_oro(5,K,2)/W_oro(1,K,2),5)
13760    IF Hh=6 THEN GOTO 15090
13770    IF W_oro(7,K,2)=0 THEN GOTO 13800
13780    End(20,10)=DROUND(W_oro(8,K,2)/W_oro(7,K,2),5)
13790    End(21,10)=DROUND(W_oro(9,K,2)/W_oro(7,K,2),4)
13800    IF W_oro(10,K,2)=0 THEN GOTO 15090
13810    End(22,10)=DROUND(W_oro(11,K,2)/W_oro(10,K,2),5)
13820    End(23,10)=DROUND(W_oro(12,K,2)/W_oro(10,K,2),4)
13830    IF D_mant(1,K)=0 THEN GOTO 13880
13840    End(14,12)=DROUND(D_mant(2,K)/D_mant(1,K),5)
13850    End(15,12)=DROUND(D_mant(3,K)/D_mant(1,K),5)
13860    End(16,12)=DROUND(D_mant(4,K)/D_mant(1,K),5)
13870    End(17,12)=DROUND(D_mant(6,K)/D_mant(1,K),5)
13880    IF D_mant(6,K)=0 THEN GOTO 13920
13890    End(18,12)=DROUND(D_mant(5,K)/D_mant(6,K),5)
13900    End(19,12)=DROUND(D_mant(5,K)/D_mant(1,K),5)
13910    IF Hh=6 THEN GOTO 15260
13920    IF D_mant(7,K)=0 THEN GOTO 13950
13930    End(20,12)=DROUND(D_mant(8,K)/D_mant(7,K),5)
13940    End(21,12)=DROUND(D_mant(9,K)/D_mant(7,K),4)
13950    IF D_mant(10,K)=0 THEN GOTO 15260
13960    End(22,12)=DROUND(D_mant(11,K)/D_mant(10,K),5)
13970    End(23,12)=DROUND(D_mant(12,K)/D_mant(10,K),4)
13980    IF Mor(1,K)=0 THEN GOTO 14030
13990    End(14,13)=DROUND(Mor(2,K)/Mor(1,K),5)
14000    End(15,13)=DROUND(Mor(3,K)/Mor(1,K),5)
14010    End(16,13)=DROUND(Mor(4,K)/Mor(1,K),5)
14020    End(17,13)=DROUND(Mor(6,K)/Mor(1,K),5)
14030    IF Mor(6,K)=0 THEN GOTO 14070
14040    End(18,13)=DROUND(Mor(5,K)/Mor(6,K),5)
14050    End(19,13)=DROUND(Mor(5,K)/Mor(1,K),5)
14060    IF Hh=6 THEN GOTO 15430
14070    IF Mor(7,K)=0 THEN GOTO 14100
14080    End(20,13)=DROUND(Mor(8,K)/Mor(7,K),5)
14090    End(21,13)=DROUND(Mor(9,K)/Mor(7,K),4)
14100    IF Mor(10,K)=0 THEN GOTO 15430
14110    End(22,13)=DROUND(Mor(11,K)/Mor(10,K),5)
14120    End(23,13)=DROUND(Mor(12,K)/Mor(10,K),4)
14130    IF Sum_up(1,K)=0 THEN GOTO 14180
14140    End(14,14)=DROUND(Sum_up(2,K)/Sum_up(1,K),5)
14150    End(15,14)=DROUND(Sum_up(3,K)/Sum_up(1,K),5)
14160    End(16,14)=DROUND(Sum_up(4,K)/Sum_up(1,K),5)
14170    End(17,14)=DROUND(Sum_up(6,K)/Sum_up(1,K),5)
14180    IF Sum_up(6,K)=0 THEN GOTO 14220
14190    End(18,14)=DROUND(Sum_up(5,K)/Sum_up(6,K),5)
14200    End(19,14)=DROUND(Sum_up(5,K)/Sum_up(1,K),5)
14210    IF Hh=6 THEN GOTO 15600
14220    IF Sum_up(7,K)=0 THEN GOTO 14250
14230    End(20,14)=DROUND(Sum_up(8,K)/Sum_up(7,K),5)
```

```
14240     End(21,14)=DROUND(Sum_up(9,K)/Sum_up(7,K),4)
14250     IF Sum_up(10,K)=0 THEN GOTO 15600
14260     End(22,14)=DROUND(Sum_up(11,K)/Sum_up(10,K),5)
14270     End(23,14)=DROUND(Sum_up(12,K)/Sum_up(10,K),4)
14280     IF Sum_low(1,K)=0 THEN GOTO 14330
14290     End(14,15)=DROUND(Sum_low(2,K)/Sum_low(1,K),5)
14300     End(15,15)=DROUND(Sum_low(3,K)/Sum_low(1,K),5)
14310     End(16,15)=DROUND(Sum_low(4,K)/Sum_low(1,K),5)
14320     End(17,15)=DROUND(Sum_low(6,K)/Sum_low(1,K),5)
14330     IF Sum_low(6,K)=0 THEN GOTO 14370
14340     End(18,15)=DROUND(Sum_low(5,K)/Sum_low(6,K),5)
14350     End(19,15)=DROUND(Sum_low(5,K)/Sum_low(1,K),5)
14360     IF Hh=6 THEN GOTO 15770
14370     IF Sum_low(7,K)=0 THEN GOTO 14400
14380     End(20,15)=DROUND(Sum_low(8,K)/Sum_low(7,K),5)
14390     End(21,15)=DROUND(Sum_low(9,K)/Sum_low(7,K),4)
14400     IF Sum_low(10,K)=0 THEN GOTO 15770
14410     End(22,15)=DROUND(Sum_low(11,K)/Sum_low(10,K),5)
14420     End(23,15)=DROUND(Sum_low(12,K)/Sum_low(10,K),4)
14430     IF Sum_sub(1,K)=0 THEN GOTO 14480
14440     End(14,16)=DROUND(Sum_sub(2,K)/Sum_sub(1,K),5)
14450     End(15,16)=DROUND(Sum_sub(3,K)/Sum_sub(1,K),5)
14460     End(16,16)=DROUND(Sum_sub(4,K)/Sum_sub(1,K),5)
14470     End(17,16)=DROUND(Sum_sub(6,K)/Sum_sub(1,K),5)
14480     IF Sum_sub(6,K)=0 THEN GOTO 14520
14490     End(18,16)=DROUND(Sum_sub(5,K)/Sum_sub(6,K),5)
14500     End(19,16)=DROUND(Sum_sub(5,K)/Sum_sub(1,K),5)
14510     IF Hh=6 THEN GOTO 15940
14520     IF Sum_sub(7,K)=0 THEN GOTO 14550
14530     End(20,16)=DROUND(Sum_sub(8,K)/Sum_sub(7,K),5)
14540     End(21,16)=DROUND(Sum_sub(9,K)/Sum_sub(7,K),4)
14550     IF Sum_sub(10,K)=0 THEN GOTO 15940
14560     End(22,16)=DROUND(Sum_sub(11,K)/Sum_sub(10,K),5)
14570     End(23,16)=DROUND(Sum_sub(12,K)/Sum_sub(10,K),4)
14580     DATA "MASS","204 Pb","206 Pb","207 Pb","208 Pb","232 Th","238 U","86 Sr
","87 Sr","87 Rb","144 Nd","143 Nd","147 Sm"
14590     RESTORE 14580
14600     FOR I=1 TO 13
14610       READ End_col$(I)
14620     NEXT I
14630     DATA "206Pb/204Pb","207Pb/204Pb","208Pb/204Pb","238U/204Pb","232Th/238U
" ,"232Th/204Pb","87Sr/86Sr","87Rb/86Sr","143Nd/144Nd","147Sm/144Nd"
14640     RESTORE 14630
14650     IF Gg=1 AND Hh=6 THEN
14660       Yyyy=14
14670       Zzzz=19
14680     END IF
14690     IF Gg=1 AND Hh=12 THEN
14700       Yyyy=14
14710       Zzzz=23
14720     END IF
14730     IF Gg=7 AND Hh=12 THEN
14740       Yyyy=20
14750       Zzzz=23
14760     END IF
```

```
14770    FOR I=14 TO 23
14780      READ End_col$(I)
14790    NEXT I
14800    DATA "Pb","Th","U","Sr","Rb","Nd","Sm"
14810    RESTORE 14800
14820     FOR I=1 TO 7
14830      READ Elem$(I)
14840     NEXT I
14850 Clear
14860    PRINT TABXY(1,10);FNH$("Printing average ending values on printer")
14870!Use_prt:PRINTER IS 701;WIDTH 132
14880    PRINT " "
14890 Print_values:!
14900              PRINT
14910              PRINT
14920    PRINT "
                                          PAGE 1"
14930    PRINT
14940    PRINT "                                                * * * ENDING MASS AN
D RESERVOIR ABUNDANCES * * *"
14950    PRINT "                                                      (AVERAGES,
in E+15 moles)"
14960    PRINT
14970    PRINT "          A2(Fraction of MORidge to create Dorogene)= ";A2,"
      Bs(Baselevel of vertical erosion)= ";Bs;" Km"
14980    PRINT "          Dp(Fraction of vertically-eroded Ucrust to go
      A3(Vertical erosion)= ";A3
14990    PRINT "            to Dorogene)= ";Dp,"                                F
inal Continental Area = ";F_area;" E+06  (Km^2)"
15000    F_area=FNArea(Sum_low(0,46),35.0,3.0)
15010    PRINT "          B1(Fraction of Porogene fill from Dorogene)= ";B1,"
       FOR ";Model$;" MODEL  ";Model_num$;" times upper mantle"
15020    PRINT "          B2(Fraction of Worogene filled from Dorogene)= ";B2
15030    PRINT "          B3(Fraction of Porogene fill from Worogene) = ";B3
15040    PRINT
15050    PRINT USING "20X,18A,2X,6(11A,3X)";"ISOTOPES","MANTLE","TOTAL ORO","UPP
ER","LOWER","  SUB","TOTAL"
15060    PRINT "                        ==========================================
==========================================="
15070    FOR I=1 TO Hh+1
15080      PRINT USING "20X,18A,2X,DDDDD.DDD,9X,2A,4X,4(DDDDD.DDD,5X)";End_col$(
I),End(I,3),"--",End(I,4),End(I,5),End(I,6),End(I,7)
15090    NEXT I
15100    PRINT
15110    PRINT
15120    PRINT
15130    PRINT
15140    PRINT "                                                * * * ENDING RESERVOIR
CONCENTRATIONS * * *"
15150    PRINT "                                                      (AVERAGES, i
n ppm)"
15160    PRINT
15170    PRINT USING "20X,18A,2X,5(11A,4X)";"ELEMENT","MANTLE","TOTAL ORO"," UPP
ER","  LOWER","   SUB"
15180    PRINT "                      ==========================================
```

```
==========================================="
15190       FOR I=Yy TO Zz
15200       PRINT USING "20X,18A,2X,DDDDD.DDDD,9X,2A,6X,4(DDDDD.DDDD,5X)";Elem$(I
),Ppm(I,3),"--",Ppm(I,4),Ppm(I,5),Ppm(I,6)
15210       NEXT I
15220     PRINT CHR$(12)     !FORM FEED
15230     PRINT "
                                                  PAGE 2"
15240     PRINT
15250     PRINT
15260     PRINT "                                      * * * ENDING OROGENIC ISOTOPIC
RATIOS * * *"
15270     PRINT
15280     PRINT USING "20X,2(16A,2X),2(20A,7X)";"MORB","EROSION","BEFORE TRUCKING
","AFTER TRUCKING"
15290     PRINT "                          ========    ====================    ========
============    ========================================="
15300     PRINT USING "18A,2X,9(10A,2X)";"RATIOS","     ","Vert","Horiz","Porogene
","Dorogene","Porogene","Dorogene","Worogene","Total_or"
15310     PRINT "=============================================================
============================================================="
15320     FOR I=Yyyy TO Zzzz
15330       PRINT USING "18A,2X,9(DD.DDDDD,4X)";End_col$(I),End(I,13),End(I,4),En
d(I,5),End(I,6),End(I,7),End(I,8),End(I,9),End(I,10),End(I,11)
15340     NEXT I
15350       PRINT
15360       PRINT
15370       PRINT
15380       PRINT
15390       PRINT
15400       PRINT "                                  * * * * ENDING ISOTOPIC RATIOS FOR
 THE MAJOR RESERVOIRS * * * *"
15410       PRINT
15420       PRINT USING "20X,18A,2X,4(10A,2X)";"RATIOS","Mantle","Upper","Lower",
"Subcrust"
15430       PRINT "                        ========================================
======================="
15440       FOR I=Yyyy TO Zzzz
15450       PRINT USING "20X,18A,2X,9(DD.DDDDD,4X)";End_col$(I),End(I,12),End(I,1
4),End(I,15),End(I,16)
15460     NEXT I
15470     PRINT CHR$(12)
15480     PRINTER IS CRT
15490     GOTO Options
15500 !
15510 Segments:!
15520           Clear
15530           PRINT TABXY(1,15);FNH$("I'm thinking")
15540     MAT End_seg= (0)
15550    FOR K=1 TO Cycles
15560      Time(K)=T_start-((K-1)*Td)
15570    NEXT K
15580 Lo=1
15590    FOR J=1 TO Cycles
15600    FOR I=1 TO 3
```

```
15610     End_seg(Lo,I,1)=DROUND(Time(Lo),2)
15620     End_seg(Lo,I,3)=M(0,1,J,I)
15630    IF M(1,1,J,I)=0 THEN 15680
15640     End_seg(Lo,I,4)=M(2,1,J,I)/M(1,1,J,I)
15650     End_seg(Lo,I,5)=M(3,1,J,I)/M(1,1,J,I)
15660     End_seg(Lo,I,6)=M(4,1,J,I)/M(1,1,J,I)
15670     End_seg(Lo,I,7)=M(6,1,J,I)/M(1,1,J,I)
15680    IF M(6,1,J,I)=0 THEN 15700
15690     End_seg(Lo,I,8)=M(5,1,J,I)/M(6,1,J,I)
15700    IF M(7,1,J,I)=0 THEN 15730
15710     End_seg(Lo,I,9)=M(8,1,J,I)/M(7,1,J,I)
15720     End_seg(Lo,I,10)=M(9,1,J,I)/M(7,1,J,I)
15730    IF M(10,1,J,I)=0 THEN 15760
15740     End_seg(Lo,I,11)=M(11,1,J,I)/M(10,1,J,I)
15750     End_seg(Lo,I,12)=M(12,1,J,I)/M(10,1,J,I)
15760    IF M(0,1,J,I)=0 THEN 15860
15770     End_seg(Lo,I,13)=FNPpm_pb(M(0,1,J,I),M(1,1,J,I),M(2,1,J,I),M(3,1,J,I),
M(4,1,J,I))
15780     End_seg(Lo,I,14)=FNPpm_others(M(0,1,J,I),M(5,1,J,I),Iso_abund(2),El_mw
(2))
15790     End_seg(Lo,I,15)=FNPpm_others(M(0,1,J,I),M(6,1,J,I),Iso_abund(3),El_mw
(3))
15800     End_seg(Lo,I,16)=FNPpm_others(M(0,1,J,I),M(7,1,J,I),Iso_abund(4),El_mw
(4))
15810     End_seg(Lo,I,17)=FNPpm_others(M(0,1,J,I),M(9,1,J,I),Iso_abund(5),El_mw
(5))
15820     End_seg(Lo,I,18)=FNPpm_others(M(0,1,J,I),M(10,1,J,I),Iso_abund(6),El_m
w(6))
15830     End_seg(Lo,I,19)=FNPpm_others(M(0,1,J,I),M(12,1,J,I),Iso_abund(7),El_m
w(7))
15840   IF I<2 OR I>2 THEN 15860
15850     End_seg(Lo,I,20)=FNArea(M(0,1,J,I),35.0-E19,3.0)   !Area of each lower
 crustal segment at 0.0 b.y.
15860  NEXT I
15870     Lo=Lo+1
15880  NEXT J
15890  Go=Lo
15900    Cname1$[1,198]=""
15910    Cname2$[1,198]=""
15920    DATA " "," ","AGE"," ","Mass","206/204","207/204","208/204","238U/204",
"232/238U"
15930    DATA "87Sr/86", "87Rb/86","143/144","147/144","  Pb  ","  U  ","  Th  "
,"  Sr  ","  Rb  ","  Nd  ","  Sm  ","Seg_area"
15940    RESTORE 15920
15950    FOR I=1 TO 22
15960      S1=9*I-8
15970      S2=S1+8
15980      READ C2$
15990      Cname2$[S1,S2]=C2$
16000    NEXT I
16010    Seg$(1)="upseg"
16020    Seg$(2)="loseg"
16030    Seg$(3)="suseg"
16040    FOR I=1 TO 3
16050      Rowlength=198
```

```
16060       File$=File1$&Seg$(I)&"A"&Drive_string$
16070       Heading$[1,Rowlength]=""
16080       Heading$[1,Rowlength]="Segment Mass (E+24 grams), Area (Km^2), Ratios
 and Concentrations (ppm) "&File1$&Seg$(I)
16090       Clear
16100       PRINT TABXY(1,10)
16110       PRINT TAB(Center);"Writing file for segements for "&FNH$(Seg$(I))
16120 Writefile2(End_seg(*),Heading$,File$,Cname1$,Cname2$,50,Go,198,22,I)
16130     NEXT I
16140     GOTO Options
16150!*************************************************************************
16160 Illegit_neg:  !
16170               Clunk
16180               Warn_neg=0
16190               RETURN
16200 !
16210 Quit: Clear
16220       OFF KEY
16230       OFF KBD
16240 PRINT TABXY(10,10);"Place "&FNH$("ISOPLOT")&" disk in left-hand drive and
press "&FNCt$
16250 PRINT
16260 PRINT TAB(Center);"Press "&FNH$("k9")&" to return to the Output Options  s
creen."
16270 ON KBD GOTO 16310
16280 ON KEY 9 LABEL "  RETURN" GOTO Options
16290 DISP "?"
16300 GOTO 16300
16310 K=FNGet_key
16320 OFF KBD
16330 SELECT K
16340 CASE -67
16350   GOTO Load_isoplot
16360 CASE ELSE
16370   GOTO Quit
16380 END SELECT
16390 Load_isoplot: Clear
16400               OFF KEY
16410               DISP FNB1$("Accessing ISOPLOT")
16420               LOAD "ISOPLOT:HP9121,700,0"
16430     END
16440 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
16450 Even:DEF FNEven(N)  ! determine whether N is an even integer
16460       IF ABS(N/2-INT(N/2))<1.E-10*N THEN
16470         RETURN 1
16480       ELSE
16490         RETURN 0
16500       END IF
16510     FNEND
16520 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
16530 Emoles:DEF FNEmoles(N,Mass1,Mass2,Bias)   ! This function normalizes the n
umber of moles to be enriched
16540  ! N is the number of moles in the reservoir to begin with
16550  ! Mass1 & Mass2 are the masses involved in the normalization
16560  ! Bias is the elemental bias (usually an enrichment factor) for each
```

```
16570  !    isotope.  It goes with Mass1.
16580        Bang=Bias/(1+Bias)
16590        Denom=Mass1*Bang+Mass2*(1-Bang)
16600        IF Denom=0 THEN RETURN 0
16610        Moles=N*Mass1*Bang/Denom
16620        RETURN Moles
16630      FNEND
16640 Read_col:DEF FNRead_col(V(*),H,I)
16650        Col(H)=V(H,I)
16660        RETURN Col(J)
16670      FNEND
16680 Convert_vc:SUB Convert_vc(Vc(*),Ms$(*),Drive_string$,F$,Rev_date$)
16690 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
16700 !     This subprogram is adapted from Ken Ludwig's subprogram to         !
16710 ! ISOPLOT200 A PLOTTING AND REGRESSION PROGRAM FOR ISOTOPE GEOCHEMISTS !
16720 ! FOR THE USE WITH HP SERIES 200 COMPUTERS (Revision May 2, 1986)       !
16730 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
16740 !
16750 ! Subprogram to access VISICALC /PF files.  The VISICALC files must have
16760 ! the following attributes:
16770 !
16780 !   The file title, if any, must occupy the first row of the file, and
16790 !   will be printed out by this program above the data.
16800 !
16810 !   The file must have been set for a column-width of 9, and stored
16820 !   with a printer-width of 80 (not 132!).
16830 !
16840 !   The column-names or headings can occupy 2 rows, but must appear dir-
16850 !   ectly above a line of "equals" signs (=============) which serves as
16860 !   both an indicator of the column-heading rows and as a visual double-
16870 !   underscore sign.
16880 !
16890 !   The sample or row names are assumed to be in the first two columns
16900 !   (A and B) of the file.  Numeric data in columns A and B will not be
16910 !   accessed by this subprogram.
16920 !
16930 !   The file can contain up to 500 rows of sample-data and up to 50 !@@
16940 !   columns.
16950 !
16960 !
16970         OPTION BASE 1
16980         OFF KEY
16990         OFF KBD
17000         File0$=File$
17010 !
17020         DIM V$[500],Temp$[80],R$[160],Str$(50)[16],Crud$[24],Cnum(50),C$(50)[
6],Cat$(80)[80],Row$[500],Vfile$(80)[10],File0$[25],Tc(5),Cpl$(5)[9]
17030         DIM Title$[80],L1$[500],L2$[500],Tfile$[25],Top_cname$(50)[9],Bottom_
cname$(50)[9],Cname$(50)[18],Bh$(5)[9],Th$(5)[9],T$[18]
17040         DIM Name$(100)[18]
17050 !
17060         Print=0
17070         Q$=CHR$(34)
17080  !
17090 Start:Clear
```

```
17100       OFF KEY
17110       PRINTER IS CRT
17120       Printfile=0
17130   !
17140       PRINT TABXY(10,2);"<<<<<<<<<<<< ";FNH$("PLUMBOTECTONICS MODELLING PRO
GRAM");" >>>>>>>>>>>>"
17150       PRINT TABXY(4,4);"Robert Zartman and Sara Haines"
17160       PRINT TABXY(65,4);"(Rev. ";Rev_date$;")"
17170       PRINT TABXY(5,10);"Press "&FNH$(" k0 ")&" for a catalog of the disk i
n the "&FNUn$("left")&"-hand drive."
17180       PRINT TABXY(5,12);"Press "&FNH$(" k1 ")&" for a catalog of the disk i
n the "&FNUn$("right")&"-hand drive."
17190       PRINT TABXY(5,16);"Press "&FNH$(" k4 ")&" to load a Visicalc datafile
 into memory."
17200   !
17210       ON KEY 4 LABEL "LOAD DATAFILE" GOTO Whichfile
17220       ON KEY 0 LABEL "CATALOG Drv #0" GOTO Drive0
17230       ON KEY 1 LABEL "CATALOG Drv #1" GOTO Drive1
17240   !
17250       GOTO 17250
17260 Use_crt:Print=0
17270       BEEP
17280       GOTO Start
17290   !
17300 Drive0:Drive_string$=Ms$(0)
17310       Drive_number=0
17320       GOTO 17360
17330   !
17340 Drive1:Drive_string$=Ms$(1)
17350       Drive_number=1
17360       OFF KEY
17370       OFF KBD
17380       Clear
17390       DISP "Getting disk catalog..."
17400       ON ERROR GOTO Bad_cat
17410       S=8! CAT strings containing filenames start on 8th CAT string
17420       Nfiles=0
17430       CAT Drive_string$ TO Cat$(*)
17440       DISP
17450       OFF ERROR
17460       PRINT USING "K,19X,K,2/,14X,K,6X,K,/";"DATAFILES ON DRIVE #"&Drive_st
ring$[13],TRIM$(Cat$(2)),FNUn$("FILE#"),FNUn$("FILE NAME")
17470       LOOP! Determine if file is a VISICALC /PF file (must  end in A and be
17480          ! padded with underscore characters.
17490         Tac$=TRIM$(REV$(Cat$(S)[1,10]))
17500       EXIT IF LEN(Tac$)=0
17510         WHILE Tac$[1,1]="_" AND LEN(Tac$)>1
17520           Tac$=Tac$[2]
17530         END WHILE
17540         IF Tac$[1,1]="A" AND Nfiles<=80 THEN
17550           Nfiles=1+Nfiles
17560           Vfile$(Nfiles)=REV$(Tac$[2])
17570           IF INT(Nfiles/15)=Nfiles/15 THEN
17580             DISP "Press"&FNCt$&"to see more files..."
17590             BEEP 330,.05
```

```
17600             PAUSE
17610           END IF
17620           PRINT TAB(16);Nfiles;TAB(21);"---  ";Vfile$(Nfiles)
17630         END IF
17640         S=S+1
17650       EXIT IF LEN(TRIM$(Cat$(S)))=0 OR S=80
17660       END LOOP
17670       OFF ERROR
17680       IF Nfiles=0 THEN
17690         PRINT USING "/,5X,K";FNH$("THERE AREN'T ANY /PF VISICALC FILES ON T
HE DISK IN DRIVE# "&Drive_string$[13])
17700         DISP "Press"&FNCt$&"when ready..."
17710         PAUSE
17720         GOTO Start
17730       ELSE
17740         DISP "ENTER THE FILE-"&FNUn$("NUMBER")&" THAT YOU WANT TO LOAD";
17750         Krlinput(Nv(*),Ninputs,R$,"",0)
17760         IF NUM(R$)=9 THEN Start
17770         Filenum=Nv(1)
17780         IF Filenum<1 OR Filenum>Nfiles THEN
17790           Bad_input("OUT OF RANGE OF KNOWN FILE NUMBERS")
17800           GOTO Start
17810         END IF
17820         Temp$=Vfile$(Filenum)
17830         GOTO Catfile
17840       END IF
17850   !
17860 Bad_cat:OFF ERROR
17870       DISP FNH$(ERRM$)
17880       Clunk
17890       WAIT 2
17900       GOTO Start
17910   !
17920 Whichfile:Clear
17930       OFF KBD
17940       OFF KEY
17950       PRINT TABXY(1,10);"WHICH VISICALC ASCII-FILE [:DRIVE] DO YOU WANT TO
BRING INTO MEMORY?"
17960       PRINT TABXY(1,12);"(example: Trialfile:1)"
17970       IF LEN(File$) THEN
17980         PRINT TABXY(1,15);"Press"&FNCt$&"for more data from "&Q$&File0$&Q$
17990         Temp$=File$
18000       END IF
18010       ON KBD GOTO 18070
18020       ON KEY 0 LABEL "CATALOG Drv #0" GOTO Drive0
18030       ON KEY 1 LABEL "CATALOG Drv #1" GOTO Drive1
18040       ON KEY 9 LABEL "    ESCAPE" GOTO Start
18050       DISP "?"
18060       GOTO 18060
18070   !
18080   K=FNGet_key
18090    OFF KBD
18100    OFF KEY
18110    SELECT K
18120    CASE 48 TO 57,65 TO 90,97,122
```

```
18130     OUTPUT KBD;CHR$(K);
18140     PRINT TABXY(1,7);RPT$(" ",80)
18150     DISP FNAst$;
18160     INPUT Temp$
18170     IF Temp$="*" THEN Start
18180   CASE -67   !CONTINUE
18190   CASE ELSE
18200     GOTO 18010
18210   END SELECT
18220   !
18230 Catfile:Clear
18240      L=LEN(Temp$)
18250      IF L=0 THEN Whichfile! ENTER pressed with no input
18260      IF L<3 THEN
18270        N1=NUM(Temp$[1,1])
18280        N2=NUM(Temp$[L,L])
18290        IF N1>48 AND N1<58 AND N2>47 AND N2<58 THEN Temp$=Vfile$(VAL(Temp$)
)
18300      END IF
18310      LOOP
18320        Colon=POS(Temp$,":")! Use colon to separate file-name from drive#
18330      EXIT IF Colon
18340        Temp$=TRIM$(Temp$)&":"&VAL$(Drive_number)
18350      END LOOP
18360      ON ERROR GOTO Badspec
18370      File$=TRIM$(Temp$[1,Colon-1])
18380      File0$=File$
18390      L=LEN(File$)
18400      IF L<10 THEN File$=File$&"A"
18410      Drive_number=VAL(Temp$[1+Colon])
18420      Drive_string$=Ms$(Drive_number)
18430      OFF ERROR
18440 !
18450      L=LEN(File$)
18460      IF L<10 THEN File$[L+1,10]=RPT$("_",10-L)! Add underscore characters
as fill to be compatible with the Visicalc format.
18470   !
18480      Badcount=0
18490 Recoup:ON ERROR GOTO Badfile
18500      DISP "Accessing file "&Q$&File0$&Q$&"..."
18510      F$=File$
18520      Tfile$=File$&Drive_string$
18530      ASSIGN @P1 TO Tfile$
18540      ENTER @P1;V$
18550      L=LEN(V$)
18560      L=MIN(80,L)
18570      Title$=V$[1,L]! First line in the file assumed to be a title-line
18580      File$=Temp$
18590      OFF ERROR
18600 !
18610      Clear
18620 !
18630      Control1=POS(Title$,CHR$(27))  ! Escape-character location -
18640      IF Control1 THEN               ! trim printer-control code if present
18650        Control2=POS(Title$[Control1]," ")! end of  control-sequence
```

```
18660         IF Control2 THEN Title$=Title$[1,Control1-1]&Title$[Control2]
18670       END IF
18680       PRINT Title$
18690       OFF KBD
18700       GOSUB Search_colnames
18710 !
18720       ON KEY 9 LABEL "    ESCAPE" GOTO Done
18730       N_colspecs=0
18740       Clear
18750       FOR I=1 TO Ncols
18760         IF LEN(C$(I)) THEN N_colspecs=1+N_colspecs
18770       NEXT I
18780       FOR I=1 TO Ncols  ! Figure out where in the row-strings to find
18790                         ! the numeric data for the specified columns.
18800         L=LEN(C$(I))
18810         IF L THEN
18820           C$(I)=UPC$(C$(I))! Convert column-letters to uppercase
18830           IF L=1 THEN Cnum(I)=NUM(C$(I))-64
18840     ! Cnum(I) is the column-number for column-letter C$(I).
18850           IF L=2 THEN Cnum(I)=26*(NUM(C$(I)[1,1])-64)+NUM(C$(I)[2,2])-64
18860           IF L>2 OR Cnum(I)<1 THEN
18870             Bad_input("INVALID COLUMN-SPECIFIER INPUT")
18880             GOTO Entercol1
18890           END IF
18900           IF ((I=1 OR I=3) AND Cnum(I)=0) OR Cnum(I)<3 OR Cnum(I)-2>Num_dat
acolumns THEN
18910             Bad_input("INVALID COLUMN-SPECIFIER RESPONSE")
18920             GOTO Enter_columns
18930           END IF
18940         END IF
18950       NEXT I
18960   !
18970       MAT Vc= (0)
18980       Row=0
18990       Npoints=0
19000       DISP CHR$(130)&"Getting data..."&CHR$(128)
19010   !
19020       REPEAT              !  Get sample names and numeric data from file
19030         ON ERROR GOTO Done! Generally will reflect end-of-file.
19040         Row$=""           ! This row-string is the entire row of the file.
19050         FOR I=1 TO Nsux ! Build row-strig from "Nsux" calls from file.
19060           ENTER @P1;V$
19070           Row$=Row$&V$[1+3*(I>1)]
19080         NEXT I
19090         OFF ERROR
19100         Row=1+Row
19110         Npoints=Row
19120         Name$(Row)=TRIM$(Row$[1,18])! Row-names in 1st 2 columns (A and B)
19130         FOR Col=1 TO Ncols
19140           C=Col
19150           ON ERROR GOTO 19180!**
19160           Temp$[1,9]=TRIM$(Row$[9*C-8,9*C])
19170           Vc(Row,Col)=VAL(Temp$)  ! Extract numeric value
19180           OFF ERROR         !**
19190         NEXT Col
```

```
19200       UNTIL Row=100
19210   !
19220 Done:OFF ERROR  ! Got all data,  so printout or display values
19230       ASSIGN @P1 TO *
19240       Clear
19250       ON KEY 9 LABEL "    ESCAPE" GOTO Exit_sub
19260   !
19270 Exit_sub:File$=File0$
19280       IF Dfile THEN Temp_dfile=0
19290       GOTO Subexit
19300   !
19310 Search_colnames:  ! Search for column-name strings in the datafile.
19320       Nsux=1
19330       Row$=V$
19340       GOTO 19430
19350       IF LEN(V$)>75 THEN ! Evidently not stored with a printer-width of 80
19360         PRINT USING "18/,K,3/";FNCenter$(FNH$(File0$&" IS NOT A COMPATIBLE
VISICALC /PF FILE"))
19370         PRINT USING "K,/";"PLEASE, "&FNUn$("PLEASE")&", "&FNH$("PLEASE")&",
 "&FNB1$("PLEASE")&" TRY TO REMEMBER NOT TO STORE YOUR /PF FILES"
19380         PRINT USING "K,4/";"WHILE THE PRINTER-WIDTH IS DEFINED AS ANYTHING
BESIDES 80 CHARACTERS."
19390         Clunk
19400         WAIT 7
19410         GOTO Convert_vc
19420       END IF
19430       IF LEN(V$)=72 THEN ! Length of 1st string is 72 only if the real row
19440         LOOP             ! is longer than 72 characters.
19450           ENTER @P1;V$
19460           IF LEN(V$)<>72 THEN ! Sorry, really only N*8 rows if still =72
19470             Nsux=1+Nsux
19480             Row$=Row$&V$[4]! The operating system adds 3 space-characters a
fter
19490           END IF         !   each 72 characters in the row.
19500         EXIT IF LEN(V$)<75
19510         END LOOP
19520       END IF
19530       Ncols=INT(LEN(Row$)/9)! Assume a column-width of 9 characters.
19540   !
19550       ON ERROR GOTO No_colnames! Look for a row-string of repeating-equals.
19560       ASSIGN @P1 TO Tfile$  ! Necessary only for files with #s of columns
19570                             !   evenly divisible by 8, I think.
19580       L1$=""                !  signs extending across the whole file-width.
19590       L2$=""                ! This row separates the column-names from the
19600       LOOP                  !   actual data.
19610         Row$=""
19620         FOR I=1 TO Nsux
19630           ENTER @P1;V$
19640           Row$=Row$&V$[1+3*(I>1)]
19650         NEXT I
19660       EXIT IF POS(Row$,RPT$("=",9*Ncols))
19670         L2$=L1$
19680         L1$=Row$
19690       END LOOP
19700       OFF ERROR
```

```
19710  !
19720  ! Extract column-names from the 2 rows above the "======..." row.
19730      Num_datacolumns=Ncols-2
19740      IF Num_datacolumns<2 THEN
19750        Bad_input("FILE CONTAINS LESS THAN TWO DATA-COLUMNS")
19760        GOTO Convert_vc
19770      END IF
19780      FOR I=1 TO Num_datacolumns
19790        ON ERROR GOTO 19810
19800        Top_cname$(I)=TRIM$(L2$[9*(I+1)+1,9*(I+2)])
19810        ON ERROR GOTO 19880
19820        Bottom_cname$(I)=TRIM$(L1$[9*(I+1)+1,9*(I+2)])
19830        IF LEN(Top_cname$(I))+LEN(Bottom_cname$(I))<18 THEN
19840          Cname$(I)=Top_cname$(I)&" "&Bottom_cname$(I)
19850        ELSE
19860          Cname$(I)=Top_cname$(I)&Bottom_cname$(I)
19870        END IF
19880        OFF ERROR
19890      NEXT I
19900  !
19910      RETURN
19920  !
19930  !
19940 Badfile:Badcount=1+Badcount ! Assume that either the case or the drive-
19950      IF Badcount<7 THEN     ! specifier is wrong, & keep trying until all
19960                             ! permutations of wrong drives/case are tried.
19970        IF Badcount=3 THEN File$=UPC$(File$)
19980        IF Badcount=5 THEN File$=LWC$(File$)
19990        Drive_string$[13,13]=VAL$(NOT (VAL(Drive_string$[13,13])))
20000        GOTO Recoup
20010      END IF
20020      DISP USING "12X,K";CHR$(131)&"  **** CAN'T ACCESS FILE "&Q$&File0$&Q$
&" ****  "&CHR$(128)
20030      File$=""
20040      Clunk
20050      WAIT 3
20060      GOTO Convert_vc
20070  !
20080 Badspec:Bad_input("INVALID FILE OR MASS-STORAGE SPECIFIER")
20090      File$=""
20100      GOTO Whichfile
20110  !
20120 No_colnames:Clear
20130      OFF ERROR
20140      PRINT USING "6/,K,/,K,/";FNH$("CAN'T FIND REPEATING-EQUALS LINE (====
=====...) THAT MUST UNDERLIE THE "),FNCenter$(FNH$(" COLUMN-NAMES "))
20150      Clunk
20160      WAIT 5
20170      Clear
20180      GOTO Convert_vc
20190    !
20200 !
20210 Subexit:PRINTER IS CRT
20220    SUBEND! ------------------------------------------------------------
--
```

```
20230 Clear:SUB Clear   !Clear screen
20240       OUTPUT KBD;CHR$(255);CHR$(75);
20250       OUTPUT KBD;CHR$(255);CHR$(84);
20260     SUBEND!-------------------------------------------------------------
---
20270 H: DEF FNH$(String$)     !Highlight string
20280       RETURN CHR$(129)&" "&String$&" "&CHR$(128)
20290     FNEND!--------------------------------------------------------------
---
20300 Un:DEF FNUn$(String$)    !Underline string
20310       RETURN CHR$(132)&String$&CHR$(128)
20320     FNEND!--------------------------------------------------------------
---
20330 Clunk:SUB Clunk  ! Clunky sound to indicate error
20340! BEEP 250,.1     ! for quiet environments
20350! BEEP 100,.2
20360       FOR J=0 TO 2000 STEP 200
20370         FOR I=1 TO 1! Better for a noisy environment
20380           BEEP 3500-J,.001
20390         NEXT I
20400       NEXT J
20410     SUBEND! -------------------------------------------------------------
--
20420 !
20430 !
20440 Krlinput:SUB Krlinput(Nval(*),Ninputs,S$,OPTIONAL Softkeys$,Crt_y,Nlines)
20450 ! Allow user to respond to a query with (a) numeric value(s) or to
20460 !  escape using and ESCAPE softkey.  If the optional SOFTKEY$ param-
20470 !  eter is present, it contains the numbers and labels of softkeys to
20480 !  be active, and the subprogam will return the number of the softkey
20490 !  pressed in the form of CHR$(Softkey number).  If the optional CRT_Y and
20500 !  NLINES parameters are present, then the CRT will be cleared from line
20510 !  CRT_Y for NLINES down.  If CRT_Y=0, then the asterisk-escape message
20520 !  will be suppressed.  CHR$(10) is returned if the CONTINUE key was
20530 !  pressed.
20540 !
20550       DIM Klabel$(0:9)[14],Skey(0:9),K$[14]
20560       IF NPAR<4 THEN No_softkeys=1
20570       IF NPAR>4 THEN
20580         IF Crt_y=0 THEN No_aster=1
20590         IF Crt_y THEN Clear_crt=1
20600       END IF
20610       OFF KEY
20620       Skey(9)=1
20630       IF NOT No_softkeys THEN
20640         IF LEN(Softkeys$) THEN
20650           REPEAT
20660             Key=VAL(Softkeys$)
20670             Skey(Key)=1
20680             Nextpos=0
20690             FOR J=2 TO LEN(Softkeys$)
20700               N=NUM(Softkeys$[J,J])
20710               IF N>47 AND N<58 THEN
20720                 Nextpos=J
20730                 GOTO 20760
```

```
20740             END IF
20750           NEXT J
20760           IF Nextpos THEN
20770             K$=Softkeys$[2,Nextpos-1]
20780             Softkeys$=Softkeys$[J]
20790           ELSE
20800             K$=Softkeys$[2,MIN(16,LEN(Softkeys$))]
20810           END IF
20820           Klabel$(Key)=RPT$(" ",7-LEN(K$)/2)&K$! Center the softkey label
20830         UNTIL Nextpos=0
20840       END IF
20850     END IF
20860     OFF KEY
20870     ON KEY 9 LABEL "    ESCAPE" GOTO 20870
20880     FOR Key=0 TO 8
20890       IF Skey(Key) THEN ON KEY Key LABEL Klabel$(Key) GOTO 20890
20900     NEXT Key
20910     IF NOT No_aster THEN DISP "?"
20920 Again:ON KBD ALL GOTO 20940
20930     GOTO 20930
20940     K=FNGet_key
20950     SELECT K
20960     CASE -80                    ! PAUSE key
20970       PAUSE
20980       GOTO Again
20990     CASE -57 TO -48             ! Softkeys 0 to 9
21000       Key=-K-48
21010       IF NOT Skey(Key) THEN Again
21020       S$=CHR$(Key)
21030       SUBEXIT
21040     CASE 48 TO 57,43,45,46,-63 ! Start of numeric input
21050       IF Clear_crt THEN PRINT TABXY(1,Crt_y);RPT$(" ",80*Nlines)
21060       IF NOT No_aster THEN DISP FNAst$;
21070       IF K=-63 THEN             ! RECALL key
21080         OUTPUT KBD;CHR$(255)&CHR$(63);
21090       ELSE                      ! Numeric key or + - .
21100         OUTPUT KBD;CHR$(K);
21110       END IF
21120       LINPUT S$
21130       IF S$="*" THEN            ! Escape if input is a single asterisk
21140         S$=CHR$(9)
21150         SUBEXIT
21160       END IF
21170       Parse((S$),Nval(*),Ninputs,0)
21180       IF Ninputs=0 THEN Again  ! Input must be numeric
21190       SUBEXIT
21200     CASE -67                    ! CONTINUE key pressed
21210       S$=CHR$(10)
21220       SUBEXIT
21230     CASE -94,-60                ! UP-arrow (KNOB)
21240       OUTPUT KBD;CHR$(255)&CHR$(94)
21250     CASE -86,-62                ! DOWN-arrow (KNOB)
21260       OUTPUT KBD;CHR$(255)&CHR$(86)
21270     CASE ELSE                   ! Invalid keystroke
21280       GOTO Again
```

```
21290       END SELECT
21300     SUBEND! ------------------------------------------------------------
--
21310 Get_key:DEF FNGet_key
21320       DIM Key$[3]
21330       ON ERROR GOTO Error
21340       Key$=KBD$
21350       L=LEN(Key$)
21360       SELECT L
21370       CASE 1
21380         Keycode=NUM(Key$)
21390       CASE 2
21400         Keycode=-NUM(Key$[2,2])
21410       CASE 3
21420         Keycode=-NUM(Key$[3,3])-128
21430       END SELECT
21440       IF Keycode=-77 THEN CALL Alpha
21450       IF Keycode=-79 THEN DUMP ALPHA
21460       IF Keycode=-76 THEN CALL Graph
21470       RETURN Keycode
21480   !
21490 Error:RETURN 0
21500     FNEND! -----------------------------------------------------------
---
21510 !
21520 Parse:SUB Parse(Input_string$,Numeric_value(*),Ninputs,OPTIONAL Enterstrin
g)
21530!From a string input, extract 1 or more numeric values, separated by commas
21540! Ninputs is the # of numeric values in the string
21550       IF NPAR=3 THEN
21560         Estring=1
21570       ELSE
21580         Estring=Enterstring
21590       END IF
21600       IF Estring THEN LINPUT Input_string$! Otherwise string supplied by ca
lling context
21610       MAT Numeric_value= (0)
21620       Comma=1
21630       Ninputs=0
21640       ON ERROR GOTO Done
21650       WHILE Comma>0
21660         Numeric_value(1+Ninputs)=VAL(Input_string$)! Extract number
21670         Ninputs=1+Ninputs! Increment #-of-values counter
21680         Comma=POS(Input_string$,",")! Position of next comma
21690         Input_string$=Input_string$[1+Comma]! Strip segment of string alrea
dy extracted
21700       END WHILE
21710 Done:SUBEND! -----------------------------------------------------------
21720 !
21730 Pmoles:DEF FNPmoles(N,Mass1,Mass2,Bias)
21740       ! THIS function normalizes when using partitioning coefficients
21750         Denom=Mass1*Bias+Mass2*(1-Bias)
21760         IF Denom=0 THEN RETURN 0
21770         Moles=N*Mass1*Bias/Denom
21780         RETURN Moles
```

```
21790       FNEND!-----------------------------------------------------------------
21800 Ppm_pb:DEF FNPpm_pb(Mass_res,Moles4,Moles6,Moles7,Moles8)
21810   X=(Moles4*204.037)+(Moles6*206.0402)+(Moles7*207.0419)+(Moles8*208.0468)
21820   IF Mass_res=0 THEN
21830    Y=0
21840    GOTO 21870
21850   END IF
21860   Y=(X/Mass_res)*(.001)
21870   RETURN Y
21880   FNEND!-----------------------------------------------------------------
21890 Ppm_others:DEF FNPpm_others(Mass_res,Moles,Iso_abund,Mw)
21900   X=(Moles/Iso_abund)*Mw
21910   IF Mass_res=0 THEN
21920    Y=0
21930     GOTO 21960
21940   END IF
21950   Y=(X/Mass_res)*(.001)
21960   RETURN Y
21970   FNEND!-----------------------------------------------------------------
21980 Bl:  DEF FNBl$(String$)      !Blinking string
21990      RETURN CHR$(130)&String$&CHR$(128)
22000      FNEND !----------------------------------------------------------------
22010 Hbl: DEF FNHbl$(String$)  ! Blinking highlighted
22020      RETURN CHR$(132)&CHR$(130)&String$&CHR$(128)&CHR$(128)
22030      FNEND !----------------------------------------------------------------
22040 Area: DEF FNArea(Mass,Height,Density)     !Area for a specific mass knowin
g height and density
22050              ! Density must be given in gm/cm^3
22060              ! Height in km*(1.0+E05 cm/km) = cm
22070              ! Mass in gm*(1.0+E24)
22080       Area=(Mass)/(Height*Density)          !in cm^2
22090       Area=DROUND(Area*1000,4)                 !in Km^2
22100       RETURN Area
22110       FNEND !----------------------------------------------------------------
22120 Writefile1: SUB Writefile1(Matrix1(*),Heading$,File$,Cname1$,Cname2$,INTEG
ER Filesize,Go,Rowlength,Ncolumn)
22130           DIM Rowstring$[600]
22140             CREATE ASCII File$,Filesize
22150             ASSIGN @Path TO File$
22160             OUTPUT @Path;Heading$[1,Rowlength]
22170             OUTPUT @Path;Cname1$[1,Rowlength]
22180             OUTPUT @Path;Cname2$[1,Rowlength]
22190             OUTPUT @Path;RPT$("=",Rowlength)
22200   !
22210             FOR H=1 TO Go
22220               Rowstring$[1,Rowlength]=""
22230               FOR J=1 TO Ncolumn-2
22240               S1=9*(J+2)-8
22250               S2=S1+8
22260               Value$=VAL$(DROUND(Matrix1(H,J),9))
22270               Rowstring$[S1,S2]=Value$
22280             NEXT J
22290   !
22300             OUTPUT @Path;Rowstring$
22310           NEXT H
```

```
22320           ASSIGN @Path TO *
22330   SUBEND !-------------------------------------------------------------
22340 Writefile2: SUB Writefile2(Matrix2(*),Heading$,File$,Cname1$,Cname2$,INTEG
ER Filesize,Go,Rowlength,Ncolumn,H)
22350           DIM Rowstring$[600]
22360             CREATE ASCII File$,Filesize
22370             ASSIGN @Path TO File$
22380             OUTPUT @Path;Heading$[1,Rowlength]
22390             OUTPUT @Path;Cname1$[1,Rowlength]
22400             OUTPUT @Path;Cname2$[1,Rowlength]
22410             OUTPUT @Path;RPT$("=",Rowlength)
22420   !
22430           FOR Lo=1 TO Go
22440               Rowstring$[1,Rowlength]=""
22450             FOR J=1 TO Ncolumn-2
22460               S1=9*(J+2)-8
22470               S2=S1+8
22480               Value$=VAL$(DROUND(Matrix2(Lo,H,J),9))
22490               Rowstring$[S1,S2]=Value$
22500             NEXT J
22510   !
22520             OUTPUT @Path;Rowstring$
22530           NEXT Lo
22540           ASSIGN @Path TO *
22550   SUBEND !-------------------------------------------------------------
22560 Ct:     DEF FNCt$
22570           RETURN " "&FNH$("CONTINUE")&" "
22580         FNEND  !-------------------------------------------------------
22590 Ast:   DEF FNAst$
22600          RETURN "(enter"&FNH$("*")&"to escape)"
22610          FNEND  !-------------------------------------------------------
```